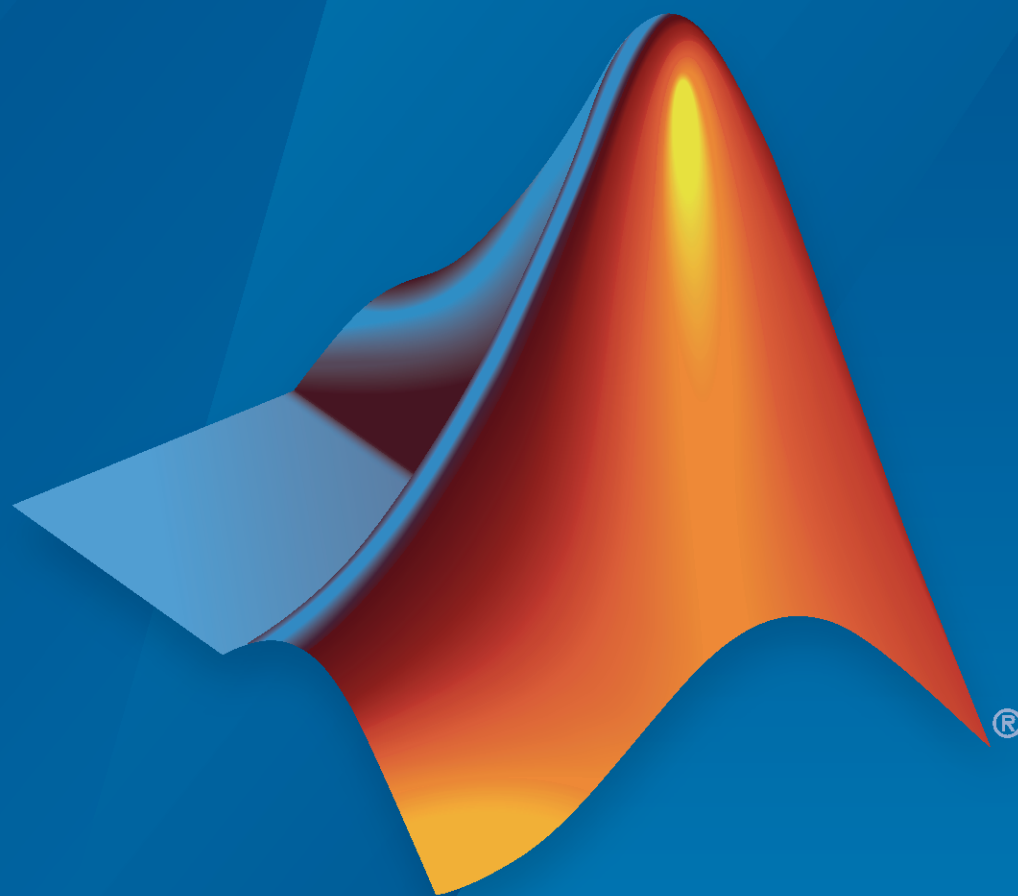


Simscape™ Electrical™

Reference (Specialized Power Systems)

Hydro-Québec



MATLAB® & SIMULINK®

R2024b



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.
1 Apple Hill Drive
Natick, MA 01760-2098

Simscape™ Electrical™ Reference (Specialized Power Systems)

© COPYRIGHT 1998-2024 by Hydro-Québec and The MathWorks, Inc.

The software described in this document is furnished under a license agreement. The software may be used or copied only under the terms of the license agreement. No part of this manual may be photocopied or reproduced in any form without prior written consent from The MathWorks, Inc.

FEDERAL ACQUISITION: This provision applies to all acquisitions of the Program and Documentation by, for, or through the federal government of the United States. By accepting delivery of the Program or Documentation, the government hereby agrees that this software or documentation qualifies as commercial computer software or commercial computer software documentation as such terms are used or defined in FAR 12.212, DFARS Part 227.72, and DFARS 252.227-7014. Accordingly, the terms and conditions of this Agreement and only those rights specified in this Agreement, shall pertain to and govern the use, modification, reproduction, release, performance, display, and disclosure of the Program and Documentation by the federal government (or other entity acquiring for or through the federal government) and shall supersede any conflicting contractual terms or conditions. If this License fails to meet the government's needs or is inconsistent in any respect with federal procurement law, the government agrees to return the Program and Documentation, unused, to The MathWorks, Inc.

Trademarks

MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.

Patents

MathWorks products are protected by one or more U.S. patents. Please see www.mathworks.com/patents for more information.

Revision History

January 1998	First printing	Version 1.0 (Release 10)
September 2000	Second printing	Revised for Version 2.1 (Release 12)
June 2001	Online only	Revised for Version 2.2 (Release 12.1)
July 2002	Online only	Revised for Version 2.3 (Release 13) (Renamed from <i>Power System Blockset User's Guide</i>)
February 2003	Third printing	Revised for Version 3.0 (Release 13SP1)
June 2004	Online only	Revised for Version 3.1 (Release 14)
October 2004	Fourth printing	Revised for Version 4.0 (Release 14SP1)
March 2005	Online only	Revised for Version 4.0.1 (Release 14SP2)
May 2005	Online only	Revised for Version 4.1 (Release 14SP2+)
September 2005	Online only	Revised for Version 4.1.1 (Release 14SP3)
March 2006	Online only	Revised for Version 4.2 (Release 2006a)
September 2006	Online only	Revised for Version 4.3 (Release 2006b)
March 2007	Online only	Revised for Version 4.4 (Release 2007a)
September 2007	Online only	Revised for Version 4.5 (Release 2007b)
March 2008	Online only	Revised for Version 4.6 (Release 2008a)
October 2008	Online only	Revised for Version 5.0 (Release 2008b)
March 2009	Online only	Revised for Version 5.1 (Release 2009a)
September 2009	Online only	Revised for Version 5.2 (Release 2009b)
March 2010	Online only	Revised for Version 5.2.1 (Release 2010a)
September 2010	Online only	Revised for Version 5.3 (Release 2010b)
April 2011	Online only	Revised for Version 5.4 (Release 2011a)
September 2011	Online only	Revised for Version 5.5 (Release 2011b)
March 2012	Online only	Revised for Version 5.6 (Release 2012a)
September 2012	Online only	Revised for Version 5.7 (Release 2012b)
March 2013	Online only	Revised for Version 5.8 (Release 2013a)
September 2013	Online only	Revised for Version 6.0 (Release 2013b) (Renamed from <i>SimPowerSystems™ Reference</i>)
March 2014	Online only	Revised for Version 6.1 (Release 2014a) (Renamed from <i>SimPowerSystems™ Reference (Second Generation)</i>)
October 2014	Online only	Revised for Version 6.2 (Release 2014b)
March 2015	Online only	Revised for Version 6.3 (Release 2015a)
September 2015	Online only	Revised for Version 6.4 (Release 2015b)
March 2016	Online only	Revised for Version 6.5 (Release 2016a) (Renamed from <i>SimPowerSystems™ Reference (Specialized Technology)</i>)
September 2016	Online only	Revised for Version 6.6 (Release 2016b)
March 2017	Online only	Revised for Version 6.7 (Release 2017a)
September 2017	Online only	Revised for Version 6.8 (Release 2017b)
March 2018	Online only	Revised for Version 6.9 (Release 2018a)
September 2018	Online only	Revised for Version 7.0 (Release 2018b) (Renamed from <i>Simscape™ Power Systems™ Reference (Specialized Technology)</i>)
March 2019	Online only	Revised for Version 7.1 (Release 2019a)
September 2019	Online only	Revised for Version 7.2 (Release 2019b)
March 2020	Online only	Revised for Version 7.3 (Release 2020a)
September 2020	Online only	Revised for Version 7.4 (Release 2020b)
March 2021	Online only	Revised for Version 7.5 (Release 2021a)
September 2021	Online only	Revised for Version 7.6 (Release 2021b)
March 2022	Online only	Revised for Version 7.7 (Release 2022a)
September 2022	Online only	Revised for Version 7.8 (Release 2022b)
March 2023	Online only	Revised for Version 7.9 (Release 2023a)
September 2023	Online only	Revised for Version 23.2 (R2023b)
March 2024	Online only	Revised for Version 24.1 (R2024a)
September 2024	Online only	Revised for Version 24.2 (R2024b)

1	Blocks

2	Functions

3	Apps

A	Technical Conventions

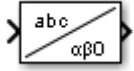
Per-Unit System of Units	A-2
International System of Units	A-2
What Is the Per-Unit System?	A-2
Example 1: Three-Phase Transformer	A-4
Example 2: Asynchronous Machine	A-5
Base Values for Instantaneous Voltage and Current Waveforms	A-6
Why Use the Per-Unit System Instead of the Standard SI Units?	A-6
Park Transformation	A-7
References	A-7

B	Bibliography

Blocks

abc to Alpha-Beta-Zero, Alpha-Beta-Zero to abc

Perform transformation from three-phase (abc) signal to $\alpha\beta 0$ stationary reference frame or the inverse



Libraries:

Simscape / Electrical / Specialized Power Systems / Control

Description

The abc to Alpha-Beta-Zero block performs a Clarke transform on a three-phase abc signal. The Alpha-Beta-Zero to abc block performs an inverse Clarke transform on the $\alpha\beta 0$ components.

$$\begin{bmatrix} u_\alpha \\ u_\beta \\ u_0 \end{bmatrix} = \begin{bmatrix} \frac{2}{3} & -\frac{1}{3} & -\frac{1}{3} \\ 0 & \frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{3}} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{bmatrix} \begin{bmatrix} u_a \\ u_b \\ u_c \end{bmatrix}$$

The inverse transformation is given by

$$\begin{bmatrix} u_a \\ u_b \\ u_c \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} & 1 \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} & 1 \end{bmatrix} \begin{bmatrix} u_\alpha \\ u_\beta \\ u_0 \end{bmatrix}$$

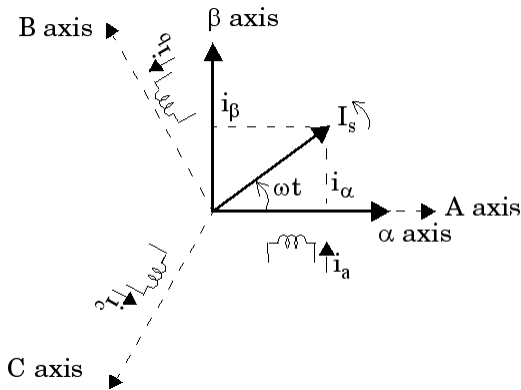
Assume that u_a , u_b , u_c quantities represent three sinusoidal balanced currents:

$$i_a = I \sin(\omega t)$$

$$i_b = I \sin\left(\omega t - \frac{2\pi}{3}\right)$$

$$i_c = I \sin\left(\omega t + \frac{2\pi}{3}\right)$$

These currents are flowing respectively into windings A, B, C of a three-phase winding, as the figure shows.



In this case, the i_α and i_β components represent the coordinates of the rotating space vector I_s in a fixed reference frame whose α axis is aligned with phase A axis. I_s amplitude is proportional to the rotating magnetomotive force produced by the three currents. It is computed as follows:

$$I_s = i_a + j \cdot i_\beta = \frac{2}{3} \left(i_a + i_b \cdot e^{j\frac{2\pi}{3}} + i_c \cdot e^{-j\frac{2\pi}{3}} \right)$$

Examples

The power_Transformations example shows various uses of blocks performing Clarke and Park transformations.

Ports

Input

abc — Vectorized abc signal
vector

Vectorized abc signal.

Output

$\alpha\beta 0$ — Vectorized $\alpha\beta 0$ signal
vector

Vectorized $\alpha\beta 0$ signal.

Version History

Introduced in R2013a

Extended Capabilities

C/C++ Code Generation

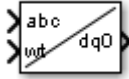
Generate C and C++ code using Simulink® Coder™.

See Also

Alpha-Beta-Zero to dq0, dq0 to Alpha-Beta-Zero

abc to dq0, dq0 to abc

Perform transformation from three-phase (abc) signal to dq0 rotating reference frame or the inverse



Libraries:

Simscape / Electrical / Specialized Power Systems / Control

Description

The abc to dq0 block uses a Park transformation to transform a three-phase (abc) signal to a dq0 rotating reference frame. The angular position of the rotating frame is given by the input ωt , in rad.

The dq0 to abc block uses an inverse Park transformation to transform a dq0 rotating reference frame to a three-phase (abc) signal. The angular position of the rotating frame is given by the input ωt , in rad.

When the rotating frame alignment at $\omega t=0$ is 90 degrees behind the phase A axis, a positive-sequence signal with $\text{Mag}=1$ and $\text{Phase}=0$ degrees yields the following dq values: $d=1$, $q=0$.

$$V_d = \frac{2}{3}(V_a \sin(\omega t) + V_b \sin(\omega t - 2\pi/3) + V_c \sin(\omega t + 2\pi/3))$$

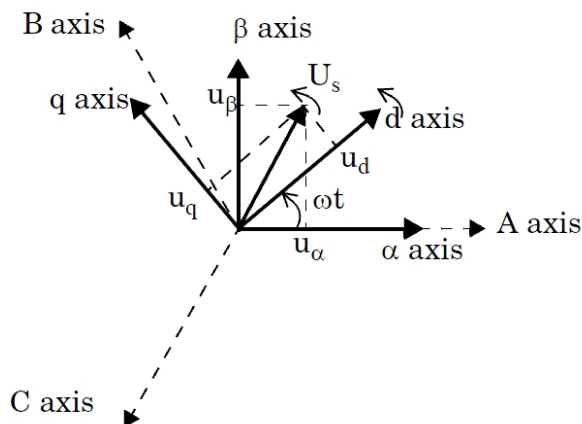
$$V_q = \frac{2}{3}(V_a \cos(\omega t) + V_b \cos(\omega t - 2\pi/3) + V_c \cos(\omega t + 2\pi/3))$$

$$V_0 = \frac{1}{3}(V_a + V_b + V_c)$$

$$V_a = V_d \sin(\omega t) + V_q \cos(\omega t) + V_0$$

$$V_b = V_d \sin(\omega t - 2\pi/3) + V_q \cos(\omega t - 2\pi/3) + V_0$$

$$V_c = V_d \sin(\omega t + 2\pi/3) + V_q \cos(\omega t + 2\pi/3) + V_0$$



The block supports the two conventions used for the Park transformation:

- When the rotating frame is aligned with the phase A axis at $t = 0$, that is, at $t = 0$, the d -axis is aligned with the a -axis. This type of Park transformation is also known as the cosine-based Park transformation.
- When the rotating frame is aligned 90 degrees behind the phase A axis, that is, at $t = 0$, the q -axis is aligned with the a -axis. This type of Park transformation is also known as the sine-based Park transformation. Use this transformation in Simscape Electrical Specialized Power Systems models with three-phase synchronous and asynchronous machines.

Deduce the dq0 components from the abc signals by performing an abc to $\alpha\beta 0$ Clarke transformation in a fixed reference frame. Then perform an $\alpha\beta 0$ to dq0 transformation in a rotating reference frame, that is, by performing a $-(\omega \cdot t)$ rotation on the space vector $U_s = u_\alpha + j \cdot u_\beta$.

The abc-to-dq0 transformation depends on the dq frame alignment at $t = 0$. The position of the rotating frame is given by $\omega \cdot t$, where ω represents the dq frame rotation speed.

When the rotating frame is aligned with the phase A axis, the following relations are obtained:

$$U_s = u_d + j \cdot u_q = (u_a + j \cdot u_\beta) \cdot e^{-j\omega t} = \frac{2}{3} \cdot \left(u_a + u_b \cdot e^{-\frac{j2\pi}{3}} + u_c \cdot e^{\frac{j2\pi}{3}} \right) \cdot e^{-j\omega t}$$

$$u_0 = \frac{1}{3}(u_a + u_b + u_c)$$

$$\begin{bmatrix} u_d \\ u_q \\ u_0 \end{bmatrix} = \frac{2}{3} \begin{bmatrix} \cos(\omega t) & \cos\left(\omega t - \frac{2\pi}{3}\right) & \cos\left(\omega t + \frac{2\pi}{3}\right) \\ -\sin(\omega t) & -\sin\left(\omega t - \frac{2\pi}{3}\right) & -\sin\left(\omega t + \frac{2\pi}{3}\right) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} u_a \\ u_b \\ u_c \end{bmatrix}$$

The inverse transformation is given by:

$$\begin{bmatrix} u_a \\ u_b \\ u_c \end{bmatrix} = \begin{bmatrix} \cos(\omega t) & -\sin(\omega t) & 1 \\ \cos\left(\omega t - \frac{2\pi}{3}\right) & -\sin\left(\omega t - \frac{2\pi}{3}\right) & 1 \\ \cos\left(\omega t + \frac{2\pi}{3}\right) & -\sin\left(\omega t + \frac{2\pi}{3}\right) & 1 \end{bmatrix} \begin{bmatrix} u_d \\ u_q \\ u_0 \end{bmatrix}$$

When the rotating frame is aligned 90 degrees behind the phase A axis, the following relations are obtained:

$$U_s = u_d + j \cdot u_q = (u_\alpha + j \cdot u_\beta) \cdot e^{-j\left(\omega t - \frac{\pi}{2}\right)}$$

$$\begin{bmatrix} u_d \\ u_q \\ u_0 \end{bmatrix} = \frac{2}{3} \begin{bmatrix} \sin(\omega t) & \sin\left(\omega t - \frac{2\pi}{3}\right) & \sin\left(\omega t + \frac{2\pi}{3}\right) \\ \cos(\omega t) & \cos\left(\omega t - \frac{2\pi}{3}\right) & \cos\left(\omega t + \frac{2\pi}{3}\right) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} u_a \\ u_b \\ u_c \end{bmatrix}$$

The inverse transformation is given by:

$$\begin{bmatrix} u_a \\ u_b \\ u_c \end{bmatrix} = \begin{bmatrix} \sin(\omega t) & \cos(\omega t) & 1 \\ \sin\left(\omega t - \frac{2\pi}{3}\right) & \cos\left(\omega t - \frac{2\pi}{3}\right) & 1 \\ \sin\left(\omega t + \frac{2\pi}{3}\right) & \cos\left(\omega t + \frac{2\pi}{3}\right) & 1 \end{bmatrix} \begin{bmatrix} u_d \\ u_q \\ u_0 \end{bmatrix}$$

Examples

Perform Clarke and Park Transformations

The `power_Transformations` example shows various ways to use blocks to perform Clarke and Park transformations.

Ports

Input

abc — abc signal
vector

abc signal, specified as a vector.

dq0 — dq0 signal
vector

dq0 signal, specified as a vector.

wt — Angular position of dq rotating frame
positive scalar

Angular position of the dq rotating frame, in radians, specified as a positive scalar.

Output

dq0 — dq0 signal
vector

dq0 signal, returned as a vector.

abc — abc signal
vector

abc signal, returned as a vector.

Parameters

Rotating frame alignment at wt=0 — Alignment of rotating frame
90 degrees behind phase A axis (default) | Aligned with phase A axis

Alignment of the rotating frame at $t = 0$ of the dq0 components of a three-phase balanced signal:

$$u_a = \sin(\omega t); u_b = \sin\left(\omega t - \frac{2\pi}{3}\right); u_c = \sin\left(\omega t + \frac{2\pi}{3}\right)$$

The positive-sequence magnitude equals 1.0 pu, and the phase angle equals 0 degrees.

When you select **Aligned with phase A axis**, the dq0 components are $d = 0$, $q = -1$, and $zero = 0$.

When you select **90 degrees behind phase A axis**, the dq0 components are $d = 1$, $q = 0$, and $zero = 0$.

Version History

Introduced in R2013a

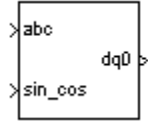
Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

abc_to_dq0 Transformation

Perform Park transformation from three-phase (abc) reference frame to dq0 reference frame



Library

powerlib_extras/Measurements, powerlib_extras/Discrete Measurements

Note The Transformations section of the Control and Measurements library contains the abc to dq0 block. This is an improved version of the abc_to_dq0 Transformation block. The new block features a mechanism that eliminates duplicate continuous and discrete versions of the same block by basing the block configuration on the simulation mode. If your legacy models contain the abc_to_dq0 Transformation block, they continue to work. However, for best performance, use the abc to dq0 block in your new models.

Description

The abc_to_dq0 Transformation block computes the direct axis, quadratic axis, and zero sequence quantities in a two-axis rotating reference frame for a three-phase sinusoidal signal. The following transformation is used:

$$V_d = \frac{2}{3}(V_a \sin(\omega t) + V_b \sin(\omega t - 2\pi/3) + V_c \sin(\omega t + 2\pi/3))$$

$$V_q = \frac{2}{3}(V_a \cos(\omega t) + V_b \cos(\omega t - 2\pi/3) + V_c \cos(\omega t + 2\pi/3))$$

$$V_0 = \frac{1}{3}(V_a + V_b + V_c),$$

where ω = rotation speed (rad/s) of the rotating frame.

The transformation is the same for the case of a three-phase current; you simply replace the V_a , V_b , V_c , V_d , V_q , and V_0 variables with the I_a , I_b , I_c , I_d , I_q , and I_0 variables.

This transformation is commonly used in three-phase electric machine models, where it is known as a Park transformation [1 on page 1-10]. It allows you to eliminate time-varying inductances by referring the stator and rotor quantities to a fixed or rotating reference frame. In the case of a synchronous machine, the stator quantities are referred to the rotor. I_d and I_q represent the two DC currents flowing in the two equivalent rotor windings (d winding directly on the same axis as the field winding, and q winding on the quadratic axis), producing the same flux as the stator I_a , I_b , and I_c currents.

You can use this block in a control system to measure the positive-sequence component V_1 of a set of three-phase voltages or currents. The V_d and V_q (or I_d and I_q) then represent the rectangular coordinates of the positive-sequence component.

You can use the Math Function block and the Trigonometric Function block to obtain the modulus and angle of V_1 :

$$|V_1| = \sqrt{V_q^2 + V_d^2}$$
$$\angle V_1 = \text{atan2}(V_q/V_d).$$

This measurement system does not introduce any delay, but, unlike the Fourier analysis done in the Sequence Analyzer block, it is sensitive to harmonics and imbalances.

Inputs and Outputs

abc

Connect to the first input the vectorized sinusoidal phase signal to be converted [phase A phase B phase C].

sin_cos

Connect to the second input a vectorized signal containing the $[\sin(\omega t) \cos(\omega t)]$ values, where ω is the rotation speed of the reference frame.

dq0

The output is a vectorized signal containing the three sequence components $[d \ q \ 0]$, in the same units as the **abc** input signal.

References

[1] Krause, P. C. *Analysis of Electric Machinery*. New York: McGraw-Hill, 1994, p.135.

Version History

Introduced in R2013a

See Also

abc to dq0, dq0 to abc | dq0_to_abc Transformation

AC Current Source

Implement sinusoidal current source



Library

Simscape / Electrical / Specialized Power Systems / Sources

Description

The AC Current Source block implements an ideal AC current source. The positive current direction is indicated by the arrow in the block icon. The generated current I is described by the following relationship:

$$I = A \sin(\omega t + \phi), \quad \omega = 2\pi f, \quad \phi = \text{Phase in radians.}$$

Negative values are allowed for amplitude and phase. A zero frequency and a 90-degree phase specify a DC current source. You cannot enter a negative frequency; the software returns an error in that case, and the block displays a question mark in the block icon. You can modify the first three block parameters at any time during the simulation.

Parameters

Peak amplitude

The peak amplitude of the generated current, in amperes (A). Default is 10.

Phase

The phase in degrees (deg). Specify a frequency of 0 and a phase of 90 degrees to implement a DC current source. Default is 0.

Frequency

The source frequency in hertz (Hz). Default is 60. Specify a frequency of 0 and a phase of 90 degrees to implement a DC current source.

Sample time

The sample period in seconds (s). The default is 0, corresponding to a continuous source.

Measurements

Default is None. Select Current to measure the current flowing through the AC Current Source block.

Place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurements** list box of the Multimeter block, the measurement is identified by a label followed by the block name.

Measurement	Label
Current	Isrc:

Examples

The `power_accurent` example uses two AC Current Source blocks in parallel to sum two sinusoidal currents in a resistor.

See Also

Controlled Current Source, Multimeter

Version History

Introduced before R2006a

AC Voltage Source

Implement sinusoidal voltage source



Libraries:

Simscape / Electrical / Specialized Power Systems / Sources

Description

The AC Voltage Source block implements an ideal AC voltage source. The generated voltage U is described by the following relationship:

$$U = A \sin(\omega t + \phi), \quad \omega = 2\pi f, \quad \phi = \text{Phase in radians.}$$

Negative values are allowed for amplitude and phase. A frequency of 0 and phase equal to 90 degrees specify a DC voltage source. Negative frequency is not allowed; otherwise the software signals an error, and the block displays a question mark in the block icon.

Examples

The `power_acvoltage` example uses two AC Voltage Source blocks at different frequencies connected in series across a resistor. The sum of the two voltages is read by a Voltage Measurement block.

Ports

Conserving

+ — Positive terminal
specialized electrical

Specialized electrical conserving port associated with the positive terminal.

- — Negative terminal
specialized electrical

Specialized electrical conserving port associated with the negative terminal.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink® Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Parameters

Peak amplitude — Peak amplitude of generated voltage
100 (default) | scalar

Peak amplitude of the generated voltage, in volts (V).

Phase — Phase

0 (default) | scalar

Phase in degrees (deg).

Frequency — Source frequency

60 (default) | scalar

Source frequency in hertz (Hz).

Sample time — Sample period

0 (default) | nonnegative scalar

Sample period in seconds (s). The default is 0, corresponding to a continuous source.

Measurements — Measurements option

None (default) | Voltage

Select **Voltage** to measure the voltage across the terminals of the AC Voltage Source block.

Place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurements** list box of the Multimeter block, the measurement is identified by a label followed by the block name:

Measurement	Label
Voltage	Usrc:

Load Flow

The load flow parameters define block parameters for use with the Load Flow tool of the Powergui block. These load flow parameters are for model initialization only. They have no impact on the block model or on the simulation performance.

The configuration of the **Load Flow** tab depends on the option selected for the **Generator type** parameter.

Generator type — Generator type of voltage source

swing (default) | PV | PQ

Specify the generator type of the voltage source.

Select **swing** to implement a generator controlling the magnitude and phase angle of its terminal voltage. Specify the reference voltage magnitude and angle in the **Swing bus or PV bus voltage** and **Swing bus voltage angle** parameters of the Load Flow Bus block connected to the voltage source terminals.

Select **PV** to implement a generator controlling its output active power **P** and voltage magnitude **V**. Specify **P** in the **Active power generation P** parameter of the block. Specify **V** in the **Swing bus or PV bus voltage** parameter of the Load Flow Bus block connected to the voltage source terminals. You can control the minimum and maximum reactive power generated by the block by using the **Minimum reactive power Qmin** and **Maximum reactive power Qmax** parameters.

Select **PQ** to implement a generator controlling its output active power **P** and reactive power **Q**. Specify **P** and **Q** in the **Active power generation P** and **Reactive power generation Q** parameters of the block, respectively.

Active power generation P — Active power generated by source

10e3 | scalar

Specify the active power that you want generated by the source, in watts. When the machine operates in motor mode, you specify a negative value.

Dependencies

To enable this parameter, set **Generator type** to **PV** or **PQ**.

Reactive power generation Q — Reactive power generated by source

0 (default) | scalar

Specify the reactive power that you want generated by the source, in vars. A negative value indicates that the reactive power is absorbed by the machine.

Dependencies

To enable this parameter, set **Generator type** to **PQ**.

Minimum reactive power Qmin — Minimum reactive power generated by the source

-inf (default) | scalar

Minimum reactive power that can be generated by the source while keeping the terminal voltage at its reference value. This reference voltage is specified in the **Swing bus or PV bus voltage** parameter of the Load Flow Bus block connected to the source terminals. The default value is `-inf`, which means that there is no lower limit on the reactive power output.

Dependencies

To enable this parameter, set **Generator type** to **PV**.

Maximum reactive power Qmax — Maximum reactive power generated by the source

inf (default) | scalar

Maximum reactive power that can be generated by the source while keeping the terminal voltage at its reference value. This reference voltage is specified in the **Swing bus or PV bus voltage** parameter of the Load Flow Bus block connected to the source terminals. The default value is `inf`, which means that there is no upper limit on the reactive power output.

Dependencies

To enable this parameter, set **Generator type** to **PV**.

Version History

Introduced before R2006a

Extended Capabilities

C/C++ Code Generation

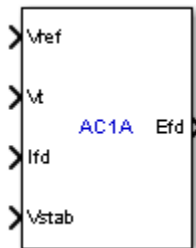
Generate C and C++ code using Simulink® Coder™.

See Also

[Controlled Voltage Source](#) | [DC Voltage Source](#) | [Multimeter](#)

AC1A Excitation System

Implements IEEE type AC1A excitation system model



Library

Simscape / Electrical / Specialized Power Systems / Electrical Machines / Synchronous Machine Control

Description

This block models an ac alternator driving a diode rectifier to produce the field voltage V_f required by the Synchronous Machine block. A non-controlled voltage regulator provides a voltage in p.u. with a lower limit of zero imposed by the diode rectifier.

This block is an adaptation of the AC1A excitation system of the IEEE® 421 standard, copyright IEEE 2005, all rights reserved.

Parameters

- “Controllers Tab” on page 1-17
- “Exciter and Rectifier Tab” on page 1-18
- “Initial Values Tab” on page 1-18

Controllers Tab

Low-pass filter time constant

The time constant T_r of the first-order system representing the stator terminal voltage transducer. Default is $20e-3$.

Voltage regulator gain and time constant

The gain K_a and time constant T_a of the first-order system representing the main regulator. Default is $[400 \ 0.02]$.

Voltage regulator internal limits

The voltage regulator internal limits V_{Amin} and V_{Amax} , in p.u. Default is $[-14.5 \ 14.5]$.

Voltage regulator output limits

The voltage regulator output limits V_{Rmin} and V_{Rmax} , in p.u. Default is $[-5.43 \ 6.03]$.

Damping filter gain and time constant

The gain K_f and time constant T_f of the first-order system representing the derivative feedback. Default is [0.03 1.0].

Transient gain reduction lead and lag time constants

The time constants T_b and T_c of the first-order system representing the lead-lag compensator. Default is [0 0].

Exciter and Rectifier Tab**Exciter gain and time constant**

The gain K_e and time constant T_e of the first-order system representing the exciter. Default is [1.0 0.80].

Exciter alternator voltage values

The exciter saturation function is defined as a multiplier of exciter alternator output voltage to represent the increase in exciter excitation requirements due to saturation [1]. The saturation function is determined by specifying two voltage points, V_{e1} and V_{e2} in p.u., on the air-gap line and no-load saturation curve and providing the corresponding two saturation multipliers Se_{Ve1} and Se_{Ve2} . Default is [4.18 3.14].

Typically, the voltage V_{e1} is a value near the expected exciter maximum output voltage, V_{e2} value is about 75% of V_{e1} [1].

Exciter saturation function values

The exciter saturation function is defined as a multiplier of exciter alternator output voltage to represent the increase in exciter excitation requirements due to saturation [1]. The saturation function is determined by specifying two voltage points, V_{e1} and V_{e2} in p.u., on the air-gap line and no-load saturation curve and providing the corresponding two saturation multipliers Se_{Ve1} and Se_{Ve2} . Default is [0.10 0.03].

Se_{Ve1} and Se_{Ve2} multipliers are equal to $C-B/B$, C is the value of exciter field current on the no-load saturation curve corresponding to the specified V_e voltage, and B is the value of exciter field current on the air-gap line corresponding to the selected V_e voltage [1].

If you do not want to model the saturation effect, set Se_{Ve1} and Se_{Ve2} values to zero.

Demagnetizing factor K_d

The gain K_d represents the demagnetizing factor, a function of exciter alternator reactances. Default is 0.38.

Rectifier loading factor K_c

The gain K_c represents the rectifier loading factor proportional to the commutating reactance. Default is 0.20.

Initial Values Tab**Initial values of terminal voltage and field voltage**

The initial values of terminal voltage V_{t0} and field voltage E_{fd0} , both in p.u. Initial terminal voltage is normally set to 1 pu. The V_{t0} and E_{fd0} values can be determined using the Powergui Load Flow tool. Default is [1 1].

Sample time

Specify a value greater than zero to discretize the block at the given sample time. Set to -1 to inherit the simulation type and sample time parameters of the Powergui block. Default is 0.

Ports

Vref

The reference value of the stator terminal voltage, in p.u.

Vt

The measured value in p.u. of the stator terminal voltage of the controlled Synchronous Machine block.

Ifd

The measured value in p.u. of the stator field current of the controlled Synchronous Machine block.

Vstab

Connect this input to a power system stabilizer to provide additional stabilization of power system oscillations. When you do not use this option, connect to a Simulink ground block. The input is in p.u.

Efd

The field voltage to apply to the Vf input of the controlled Synchronous Machine block. The output is in p.u.

References

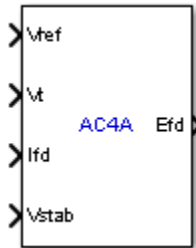
- [1] "IEEE Recommended Practice for Excitation System Models for Power System Stability Studies." *IEEE Standard*, Vol. 421, No. 5, 2005 (Revision of IEEE 521.5-1992).

Version History

Introduced in R2012b

AC4A Excitation System

Implements IEEE type AC4A excitation system model



Library

Simscape / Electrical / Specialized Power Systems / Electrical Machines / Synchronous Machine Control

Description

This block models an ac alternator connected to a thyristor rectifier to produce the field voltage V_f required by the Synchronous Machine block. A voltage regulator controls the thyristors and the excitation system stabilization is accomplished by a series lead-lag compensator.

This block is an adaptation of the AC4A excitation system of the IEEE 421 standard, copyright IEEE 2005, all rights reserved.

Parameters

- “Controllers Tab” on page 1-20
- “Exciter and Rectifier Tab” on page 1-21
- “Initial Values Tab” on page 1-21

Controllers Tab

Low-pass filter time constant

The time constant T_r of the first-order system representing the stator terminal voltage transducer. Default is $20e-3$.

Voltage regulator gain and time constant

The gain K_a and time constant T_a of the first-order system representing the main regulator. Default is $[200 \ 0.015]$.

Voltage regulator input limits

The voltage regulator input limits V_{lmin} and V_{lmax} , in p.u. Default is $[-10 \ 10]$.

Voltage regulator output limits

The voltage regulator output limits V_{Rmin} and V_{Rmax} , in p.u. Default is $[-4.53 \ 5.64]$.

Transient gain reduction lead and lag time constants

The time constants T_b and T_c of the first-order system representing the lead-lag compensator. Default is [10 1.0].

Exciter and Rectifier Tab**Rectifier loading factor K_c**

The gain K_c represents the rectifier loading factor proportional to the commutating reactance. Default is 0.

Initial Values Tab**Initial values of terminal voltage and field voltage**

The initial values of terminal voltage V_{t0} and field voltage E_{fd0} , both in p.u. Initial terminal voltage is normally set to 1 pu. The V_{t0} and E_{fd0} values can be determined using the Powergui Load Flow tool. Default is [1 1].

Sample time

Specify a value greater than zero to discretize the block at the given sample time. Set to -1 to inherit the simulation type and sample time parameters of the Powergui block. Default is 0.

Ports

V_{ref}

The reference value of the stator terminal voltage, in p.u.

V_t

The measured value in p.u. of the stator terminal voltage of the controlled Synchronous Machine block.

I_{fd}

The measured value in p.u. of the stator field current of the controlled Synchronous Machine block.

V_{stab}

Connect this input to a power system stabilizer to provide additional stabilization of power system oscillations. When you do not use this option, connect to a Simulink ground block. The input is in p.u.

E_{fd}

The field voltage to apply to the V_f input of the controlled Synchronous Machine block. The output is in p.u.

References

- [1] "IEEE Recommended Practice for Excitation System Models for Power System Stability Studies." *IEEE Standard*, Vol. 421, No. 5, 2005 (Revision of IEEE 521.5-1992).

Version History

Introduced in R2012b

AC5A Excitation System

Implements IEEE type AC5A excitation system model



Library

Simscape / Electrical / Specialized Power Systems / Electrical Machines / Synchronous Machine Control

Description

This block models a simplified brushless excitation system where the regulator is supplied from a permanent magnet generator. This model is good to represent simplified systems with rotating rectifiers.

This block is an adaptation of the AC5A excitation system of the IEEE 421 standard, copyright IEEE 2005, all rights reserved.

Parameters

- “Controllers Tab” on page 1-22
- “Exciter and Rectifier Tab” on page 1-23
- “Initial Values Tab” on page 1-23

Controllers Tab

Low-pass filter time constant

The time constant T_r of the first-order system representing the stator terminal voltage transducer. Default is $20e-3$.

Voltage regulator gain and time constant

The gain K_a and time constant T_a of the first-order system representing the main regulator. Default is $[100 \ 0.02]$.

Voltage regulator output limits

The voltage regulator output limits V_{Rmin} and V_{Rmax} , in p.u. Default is $[-7.3 \ 7.3]$.

Damping filter gain and time constants

The gain K_f and time constants T_{f1} , T_f , and T_{f3} of the second-order system representing the derivative feedback. Default is $[0.03 \ 1 \ 0 \ 0]$.

Exciter and Rectifier Tab

Exciter gain and time constant

The gain K_e and time constant T_e of the first-order system representing the exciter. Default is [1.0 0.02].

Field voltage values

The exciter saturation function is defined as a multiplier of exciter alternator output voltage to represent the increase in exciter excitation requirements due to saturation [1]. The saturation function is determined by specifying two voltage points, E_{fd1} and E_{fd2} in p.u., on the air-gap line and constant resistance load saturation curve and providing the corresponding two saturation multipliers Se_{Efd1} and Se_{Efd2} . Default is [5.6 0.75*5.6].

Typically, the voltage E_{fd1} is a value near the exciter expected maximum output voltage, E_{fd2} value is about 75% of E_{fd1} .

Exciter saturation function values

The exciter saturation function is defined as a multiplier of exciter alternator output voltage to represent the increase in exciter excitation requirements due to saturation [1]. The saturation function is determined by specifying two voltage points, E_{fd1} and E_{fd2} in p.u., on the air-gap line and Constant Resistance Load saturation curve, and providing the corresponding two saturation multipliers Se_{Efd1} and Se_{Efd2} . Default is [0.86 0.5].

Se_{Efd1} and Se_{Efd2} multipliers are equal to $A-B / B$, A is the value of exciter field current on the Constant Resistance Load saturation curve corresponding to the selected E_{fd} voltage, and B is the value of exciter field current on the air-gap line corresponding to the selected E_{fd} voltage.

If you do not want to model the saturation effect, set Se_{Efd1} and Se_{Efd2} values to zero.

Initial Values Tab

Initial values of terminal voltage and field voltage

The initial values of terminal voltage V_{t0} and field voltage E_{fd0} , both in p.u. Initial terminal voltage is normally set to 1 pu. The V_{t0} and E_{fd0} values can be determined using the Powergui Load Flow tool. Default is [1 1].

Sample time

Specify a value greater than zero to discretize the block at the given sample time. Set to -1 to inherit the simulation type and sample time parameters of the Powergui block. Default is 0.

Ports

V_{ref}

The reference value of the stator terminal voltage, in p.u.

V_t

The measured value in p.u. of the stator terminal voltage of the controlled Synchronous Machine block.

V_{stab}

Connect this input to a power system stabilizer to provide additional stabilization of power system oscillations. When you do not use this option, connect to a Simulink ground block. The input is in p.u.

Efd

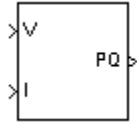
The field voltage to apply to the Vf input of the controlled Synchronous Machine block. The output is in p.u.

References

- [1] "IEEE Recommended Practice for Excitation System Models for Power System Stability Studies." *IEEE Standard*, Vol. 421, No. 5, 2005 (Revision of IEEE 521.5-1992).

Active & Reactive Power

Measure active and reactive powers of voltage-current pair



Library

powerlib_extras/Measurements

A discrete version of this block is available in the powerlib_extras/Discrete Measurements library.

A phasor version of this block is available in the powerlib_extras/Phasor Library.

Note The Sensors and Measurements library contains the Power block. This is an improved version of the Active & Reactive Power block. The new block features a mechanism that eliminates duplicate continuous and discrete versions of the same block by basing the block configuration on the simulation mode. If your legacy models contain the Active & Reactive Power block, they continue to work. However, for best performance, use the Power block in your new models.

Description

The Active & Reactive Power block measures the active power P and reactive power Q associated with a periodic voltage-current pair that can contain harmonics. P and Q are calculated by averaging the $V I$ product with a running average window over one cycle of the fundamental frequency, so that the powers are evaluated at fundamental frequency.

$$P = \frac{1}{T} \int_{t-T}^t V(\omega t) \cdot I(\omega t) dt$$

$$Q = \frac{1}{T} \int_{t-T}^t V(\omega t) \cdot I(\omega t - \pi/2) dt,$$

where $T = 1/(\text{fundamental frequency})$.

A current flowing into an RL branch, for example, produces positive active and reactive powers.

As this block uses a running window, one cycle of simulation has to be completed before the output gives the correct active and reactive powers.

The discrete version of this block, available in the powerlib_extras/Discrete Measurements library, allows you to specify the initial input voltage and current (magnitude and phase). For the first cycle of simulation, the outputs are held constant using the values specified by the initial input parameters.

Parameters

Fundamental frequency (Hz)

The fundamental frequency, in hertz, of the instantaneous voltage and current.

Inputs and Outputs

V

The first input is the instantaneous voltage, in volts.

I

The second input is the instantaneous current, in amperes.

PQ

The output is a vector [P Q] of the active and reactive powers, in watts and vars, respectively.

Examples

The “Three-Winding Distribution Transformer” example simulates a three-winding distribution transformer rated at 75 kVA:14400/120/120 V.

Version History

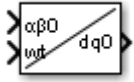
Introduced before R2006a

See Also

Power

Alpha-Beta-Zero to dq0, dq0 to Alpha-Beta-Zero

Perform transformation from $\alpha\beta 0$ stationary reference frame to dq0 rotating reference frame or the inverse



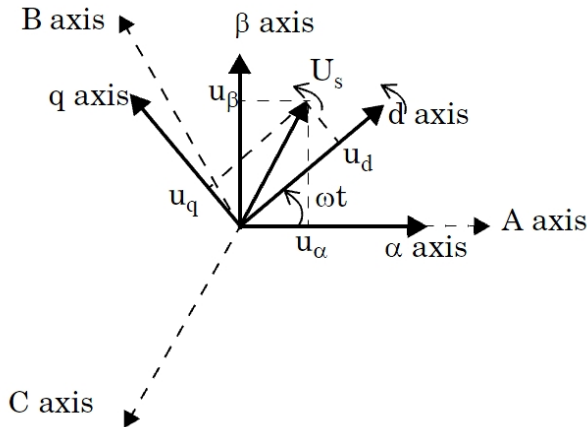
Libraries:

Simscape / Electrical / Specialized Power Systems / Control

Description

The Alpha-Beta-Zero to dq0 block performs a transformation of $\alpha\beta 0$ Clarke components in a fixed reference frame to dq0 Park components in a rotating reference frame.

The dq0 to Alpha-Beta-Zero block performs a transformation of dq0 Park components in a rotating reference frame to $\alpha\beta 0$ Clarke components in a fixed reference frame.



The block supports the two conventions used in the literature for Park transformation:

- Rotating frame aligned with A axis at $t = 0$. This type of Park transformation is also known as the cosine-based Park transformation.
- Rotating frame aligned 90 degrees behind A axis. This type of Park transformation is also known as the sine-based Park transformation. Use it in Simscape Electrical Specialized Power Systems models of three-phase synchronous and asynchronous machines.

Knowing that the position of the rotating frame is given by $\omega.t$ (where ω represents the frame rotation speed), the $\alpha\beta 0$ to dq0 transformation performs a $-(\omega.t)$ rotation on the space vector $U_s = u_\alpha + j \cdot u_\beta$. The homopolar or zero-sequence component remains unchanged.

Depending on the frame alignment at $t = 0$, the dq0 components are deduced from $\alpha\beta 0$ components as follows:

When the rotating frame is aligned with A axis, the following relations are obtained:

$$U_s = u_d + j \cdot u_q = (u_\alpha + j \cdot u_\beta) \cdot e^{-j\omega t}$$

$$\begin{bmatrix} u_d \\ u_q \\ u_0 \end{bmatrix} = \begin{bmatrix} \cos(\omega t) & \sin(\omega t) & 0 \\ -\sin(\omega t) & \cos(\omega t) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_\alpha \\ u_\beta \\ u_0 \end{bmatrix}$$

The inverse transformation is given by

$$u_\alpha + j \cdot u_\beta = (u_d + j \cdot u_q) \cdot e^{j\omega t}$$

$$\begin{bmatrix} u_\alpha \\ u_\beta \\ u_0 \end{bmatrix} = \begin{bmatrix} \cos(\omega t) & -\sin(\omega t) & 0 \\ \sin(\omega t) & \cos(\omega t) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_d \\ u_q \\ u_0 \end{bmatrix}$$

When the rotating frame is aligned 90 degrees behind A axis, the following relations are obtained:

$$U_s = u_d + j \cdot u_q = (u_\alpha + j \cdot u_\beta) \cdot e^{-j(\omega t - \frac{\pi}{2})}$$

$$\begin{bmatrix} u_d \\ u_q \\ u_0 \end{bmatrix} = \frac{2}{3} \begin{bmatrix} \sin(\omega t) & \sin(\omega t - \frac{2\pi}{3}) & \sin(\omega t + \frac{2\pi}{3}) \\ \cos(\omega t) & \cos(\omega t - \frac{2\pi}{3}) & \cos(\omega t + \frac{2\pi}{3}) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} u_a \\ u_b \\ u_c \end{bmatrix}$$

The inverse transformation is given by

$$u_\alpha + j \cdot u_\beta = (u_d + j \cdot u_q) \cdot e^{j(\omega t - \frac{\pi}{2})}$$

The abc-to-Alpha-Beta-Zero transformation applied to a set of balanced three-phase sinusoidal quantities u_a , u_b , u_c produces a space vector U_s whose u_α and u_β coordinates in a fixed reference frame vary sinusoidally with time. In contrast, the abc-to-dq0 transformation (Park transformation) applied to a set of balanced three-phase sinusoidal quantities u_a , u_b , u_c produces a space vector U_s whose u_d and u_q coordinates in a dq rotating reference frame stay constant.

Examples

The `power_Transformations` example shows various uses of blocks performing Clarke and Park transformations.

Ports

Input

$\alpha\beta 0$ — Vectorized $\alpha\beta 0$ signal
vector

Vectorized $\alpha\beta 0$ signal.

$dq 0$ — Vectorized dq0 signal
vector

Vectorized dq0 signal.

Output

wt — Angular position of dq rotating frame
scalar

The angular position, in radians, of the dq rotating frame relative to the stationary frame.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Rotating frame alignment (at wt=0) — Alignment of rotating frame
90 degrees behind phase A axis (default) | Aligned with phase A axis

Alignment of the rotating frame, when $wt = 0$, of the dq0 components of a three-phase balanced signal:

$$u_a = \sin(\omega t); \quad u_b = \sin\left(\omega t - \frac{2\pi}{3}\right); \quad u_c = \sin\left(\omega t + \frac{2\pi}{3}\right)$$

(positive-sequence magnitude = 1.0 pu; phase angle = 0 degree)

When you select **Aligned with phase A axis**, the dq0 components are $d = 0$, $q = -1$, and zero = 0.

When you select **90 degrees behind phase A axis**, the default option, the dq0 components are $d = 1$, $q = 0$, and zero = 0.

Version History

Introduced in R2013a

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

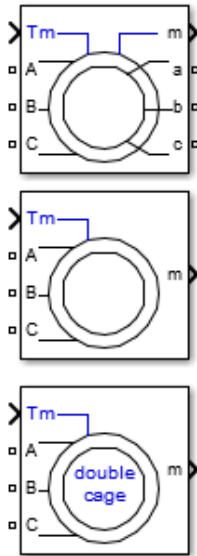
abc to Alpha-Beta-Zero, Alpha-Beta-Zero to abc

Asynchronous Machine

Model dynamics of three-phase asynchronous machine, also known as induction machine, in SI or pu units

Libraries:

Simscape / Electrical / Specialized Power Systems / Electrical Machines



Description

The Asynchronous Machine SI Units and Asynchronous Machine pu Units blocks implement a three-phase asynchronous machine (wound rotor, squirrel cage, or double squirrel cage) modeled in a selectable dq reference frame (rotor, stator, or synchronous). Stator and rotor windings are connected in wye to an internal neutral point. The block operates in either generator or motor mode. The mode of operation is dictated by the sign of the mechanical torque:

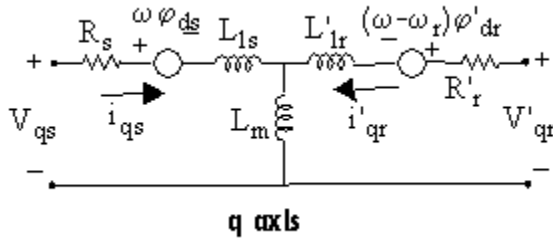
- If **Tm** is positive, the machine acts as a motor.
- If **Tm** is negative, the machine acts as a generator.

The electrical part of the machine is represented by a fourth-order (or sixth-order for the double squirrel-cage machine) state-space model, and the mechanical part by a second-order system. All electrical variables and parameters are referred to the stator, which is indicated by the prime signs in the following machine equations. All stator and rotor quantities are in the arbitrary two-axis reference frame (dq frame). The subscripts used are defined in this table.

Subscript	Definition
d	d -axis quantity
q	q -axis quantity
r	Rotor quantity (wound-rotor or single-cage)
$r1$	Cage 1 rotor quantity (double-cage)
$r2$	Cage 2 rotor quantity (double-cage)

Subscript	Definition
s	Stator quantity
l	Leakage inductance
m	Magnetizing inductance

Electrical System of the Wound-Rotor or Squirrel-Cage Machine



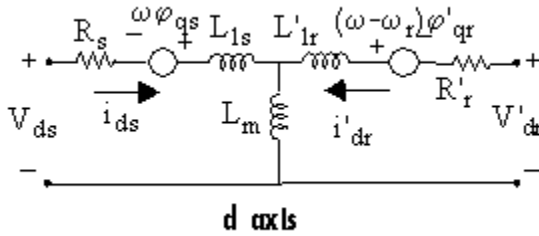
$$V_{qs} = R_s i_{qs} + d\phi_{qs}/dt + \omega \phi_{ds}$$

$$V_{ds} = R_s i_{ds} + d\phi_{ds}/dt - \omega \phi_{qs}$$

$$V'_{qr} = R'_r i'_{qr} + d\phi'_{qr}/dt + (\omega - \omega_r) \phi'_{dr}$$

$$V'_{dr} = R'_r i'_{dr} + d\phi'_{dr}/dt - (\omega - \omega_r) \phi'_{qr}$$

$$T_e = 1.5p(\phi_{ds} i_{qs} - \phi_{qs} i_{ds})$$



ω — Reference frame angular velocity

ω_r — Electrical angular velocity

$$\phi_{qs} = L_s i_{qs} + L_m i'_{qr}$$

$$\phi_{ds} = L_s i_{ds} + L_m i'_{dr}$$

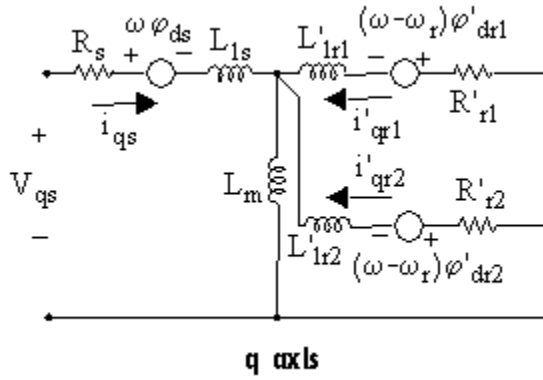
$$\phi'_{qr} = L'_r i'_{qr} + L_m i_{qs}$$

$$\phi'_{dr} = L'_r i'_{dr} + L_m i_{ds}$$

$$L_s = L_{ls} + L_m$$

$$L'_r = L'_{lr} + L_m$$

Electrical System of the Double Squirrel-Cage Machine



$$V_{qs} = R_s i_{qs} + d\phi_{qs}/dt + \omega \phi_{ds}$$

$$V_{ds} = R_s i_{ds} + d\phi_{ds}/dt - \omega \phi_{qs}$$

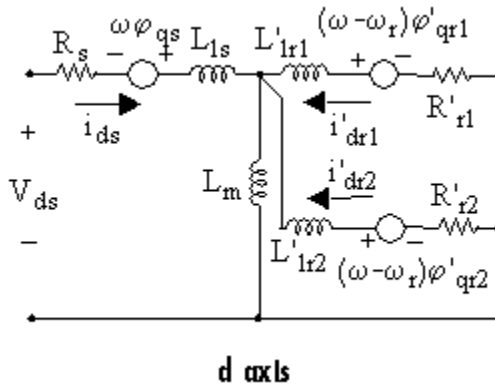
$$0 = R'_{r1} i'_{qr1} + d\phi'_{qr1}/dt + (\omega - \omega_r) \phi'_{dr1}$$

$$0 = R'_{r1} i'_{dr1} + d\phi'_{dr1}/dt - (\omega - \omega_r) \phi'_{qr1}$$

$$0 = R'_{r2} i'_{qr2} + d\phi'_{qr2}/dt + (\omega - \omega_r) \phi'_{dr2}$$

$$0 = R'_{r2} i'_{dr2} + d\phi'_{dr2}/dt - (\omega - \omega_r) \phi'_{qr2}$$

$$T_e = 1.5p(\phi_{ds} i_{qs} - \phi_{qs} i_{ds})$$



$$\phi_{qs} = L_s i_{qs} + L_m (i'_{qr1} + i'_{qr2})$$

$$\phi_{ds} = L_s i_{ds} + L_m (i'_{dr1} + i'_{dr2})$$

$$\phi'_{qr1} = L'_{r1} i'_{qr1} + L_m i_{qs}$$

$$\phi'_{dr1} = L'_{r1} i'_{dr1} + L_m i_{ds}$$

$$\phi'_{qr2} = L'_{r2} i'_{qr2} + L_m i_{qs}$$

$$\varphi'_{dr2} = L'_{r2}i'_{dr2} + L_m i_{ds}$$

$$L_s = L_{ls} + L_m$$

$$L'_{r1} = L'_{lr1} + L_m$$

$$L'_{r2} = L'_{lr2} + L_m$$

Mechanical System

$$\frac{d}{dt}\omega_m = \frac{1}{2H}(T_e - F\omega_m - T_m)$$

$$\frac{d}{dt}\theta_m = \omega_m$$

The Asynchronous Machine block parameters are defined in the table. All quantities are referred to the stator.

Parameters Common to All Models	Definition
R_s, L_{ls}	Stator resistance and leakage inductance
L_m	Magnetizing inductance
L_s	Total stator inductance
V_{qs}, i_{qs}	q -axis stator voltage and current
V_{ds}, i_{ds}	d -axis stator voltage and current
ϕ_{qs}, ϕ_{ds}	Stator q -axis and d -axis fluxes
ω_m	Angular velocity of the rotor
θ_m	Rotor angular position
p	Number of pole pairs
ω_r	Electrical angular velocity ($\omega_m \times p$)
θ_r	Electrical rotor angular position ($\theta_m \times p$)
T_e	Electromagnetic torque
T_m	Shaft mechanical torque
J	Combined rotor and load inertia coefficient. Set to infinite to simulate locked rotor.
H	Combined rotor and load inertia constant. Set to infinite to simulate locked rotor.
F	Combined rotor and load viscous friction coefficient
Parameters Specific to Single-Cage or Wound Rotor	Definition
L'_r	Total rotor inductance
R'_r, L'_{lr}	Rotor resistance and leakage inductance
V'_{qr}, i'_{qr}	q -axis rotor voltage and current
V'_{dr}, i'_{dr}	d -axis rotor voltage and current

Parameters Specific to Single-Cage or Wound Rotor	Definition
ϕ'_{qr}, ϕ'_{dr}	Rotor q -axis and d axis fluxes
Parameters Specific to Double-Cage Rotor	Definition
R'_{r1}, L'_{lr1}	Rotor resistance and leakage inductance of cage 1
R'_{r2}, L'_{lr2}	Rotor resistance and leakage inductance of cage 2
L'_{r1}, L'_{r2}	Total rotor inductances of cage 1 and 2
i'_{qr1}, i'_{qr2}	q -axis rotor current of cage 1 and 2
i'_{dr1}, i'_{dr2}	d -axis rotor current of cage 1 and 2
ϕ'_{qr1}, ϕ'_{dr1}	q -axis and d -axis rotor fluxes of cage 1
ϕ'_{qr2}, ϕ'_{dr2}	q -axis and d -axis rotor fluxes of cage 2

Examples

Use of the Asynchronous Machine Block in Motor Mode

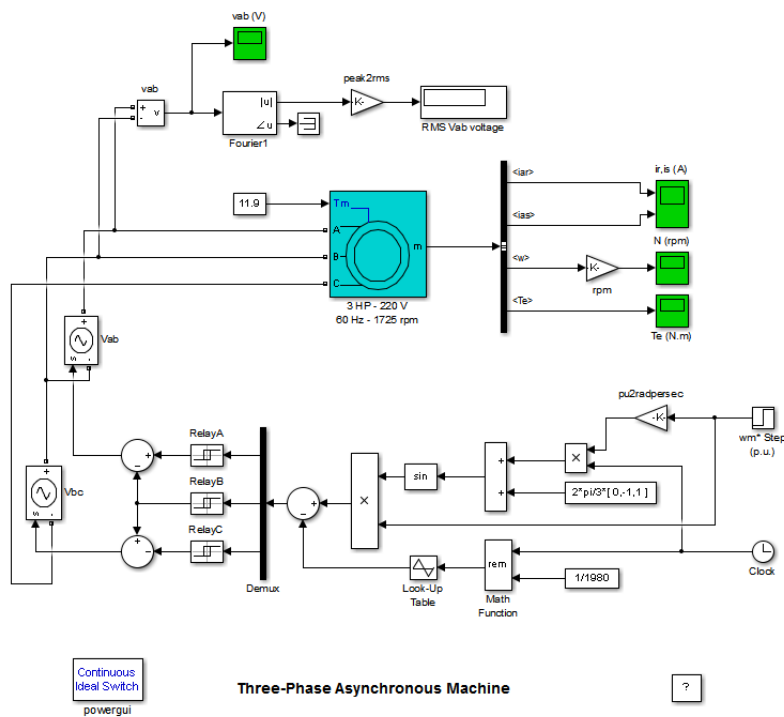
The “Three-Phase Asynchronous Machine” example uses a Asynchronous Machine block in motor mode. The example consists of an asynchronous machine in an open-loop speed control system.

The machine rotor is short-circuited, and the stator is fed by a PWM inverter built with Simulink blocks and interfaced to the Asynchronous Machine block through the Controlled Voltage Source block. The inverter uses sinusoidal pulse-width modulation. The base frequency of the sinusoidal reference wave is set at 60 Hz and the triangular carrier wave frequency is set at 1980 Hz. This frequency corresponds to a frequency modulation factor m_f of 33 (60 Hz x 33 = 1980).

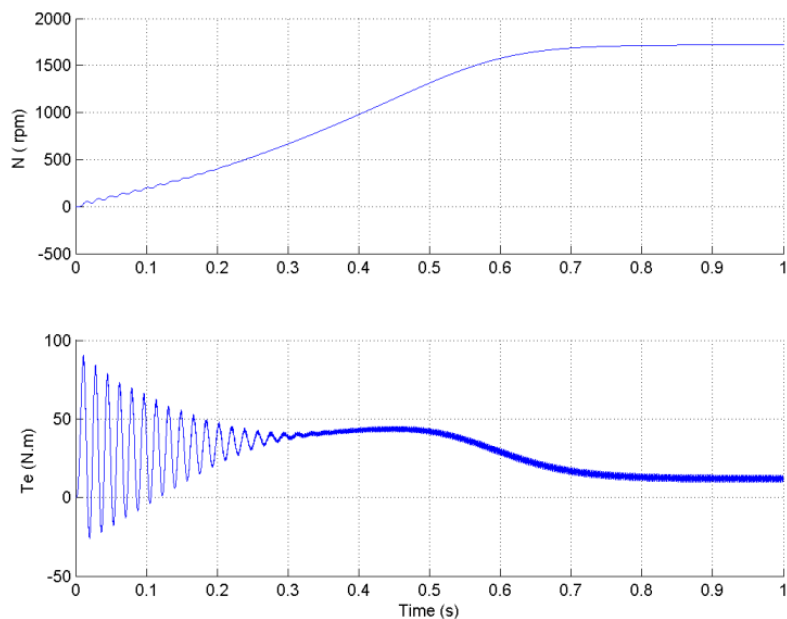
The 3 HP machine is connected to a constant load of nominal value (11.9 N.m). It is started and reaches the set point speed of 1.0 pu at $t = 0.9$ seconds.

The parameters of the machine are the same as the Asynchronous Machine SI Units block, except for the stator leakage inductance, which is set to twice the normal value to simulate a smoothing inductor placed between the inverter and the machine. Also, the stationary reference frame was used to obtain the results shown.

Open the “Three-Phase Asynchronous Machine” example. In the simulation parameters, a small relative tolerance is required because of the high switching rate of the inverter.

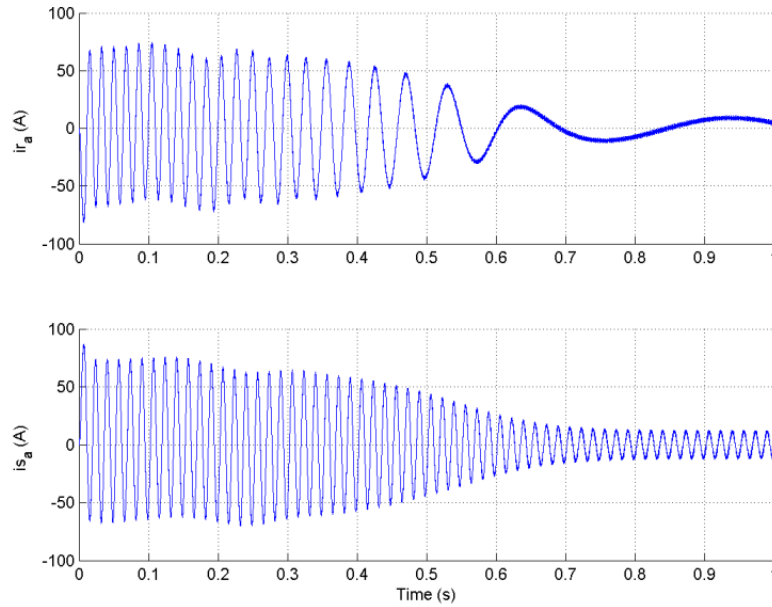


Run the simulation and observe the machine's speed and torque.

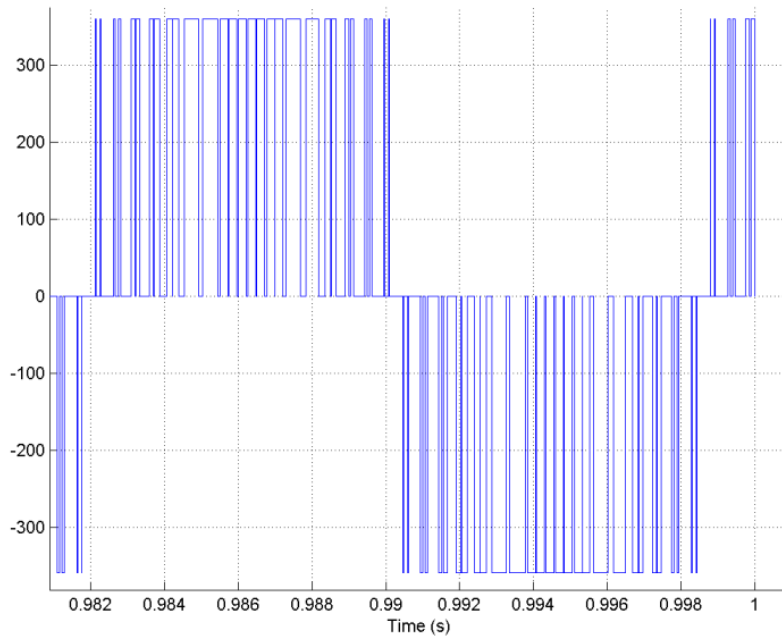


The first graph shows the machine's speed going from 0 to 1725 rpm (1.0 pu). The second graph shows the electromagnetic torque developed by the machine. Because the stator is fed by a PWM inverter, a noisy torque is observed.

However, this noise is not visible in the speed because it is filtered out by the machine's inertia, but it can be seen in the stator and rotor currents.

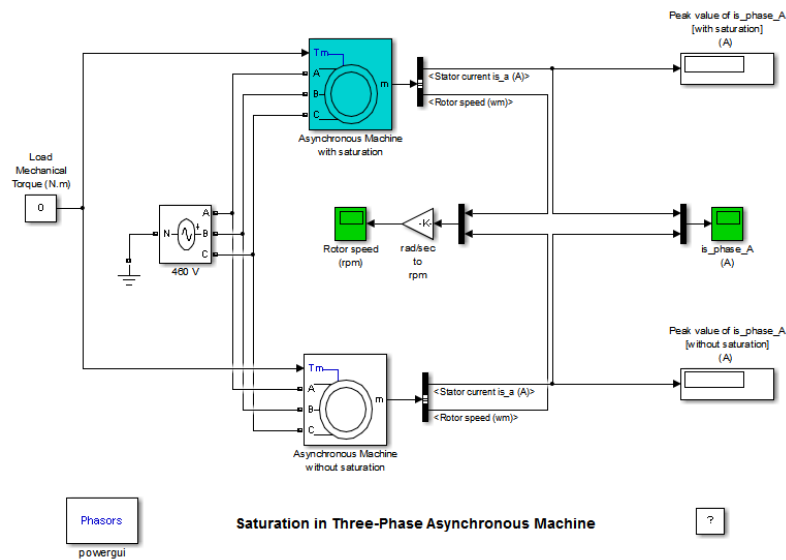


Look at the output of the PWM inverter. Because nothing of interest can be seen at the simulation time scale, the graph concentrates on the last moments of the simulation.



Effect of Saturation of the Asynchronous Machine Block

The “Saturation in Three-Phase Asynchronous Machine” example illustrates the effect of saturation of the Asynchronous Machine block.



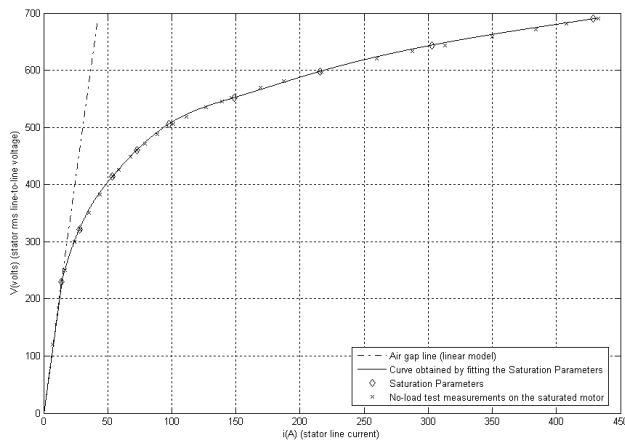
Two identical three-phase motors (50 HP, 460 V, and 1800 rpm) are simulated, with and without saturation, to observe the saturation effects on the stator currents. Two different simulations are realized in the example.

The first simulation is the no-load steady-state test. This table contains the values of the saturation parameters and the measurements obtained by simulating different operating points on the saturated motor (no-load and in steady-state).

Saturation Parameters		Measurements	
Vsat (Vrms L-L)	Isat (peak A)	Vrms L-L	Is_A (peak A)
-	-	120	7.322
230	14.04	230	14.03
-	-	250	16.86
-	-	300	24.04
322	27.81	322	28.39
-	-	351	35.22
-	-	382	43.83
414	53.79	414	54.21
-	-	426	58.58
-	-	449	67.94
460	72.69	460	73.01
-	-	472	79.12
-	-	488	88.43
506	97.98	506	100.9
-	-	519	111.6
-	-	535	126.9
-	-	546	139.1

Saturation Parameters		Measurements	
Vsat (Vrms L-L)	Isat (peak A)	Vrms L-L	Is_A (peak A)
552	148.68	552	146.3
-	-	569	169.1
-	-	581	187.4
598	215.74	598	216.5
-	-	620	259.6
-	-	633	287.8
644	302.98	644	313.2
-	-	659	350
-	-	672	383.7
-	-	681	407.9
690	428.78	690	432.9

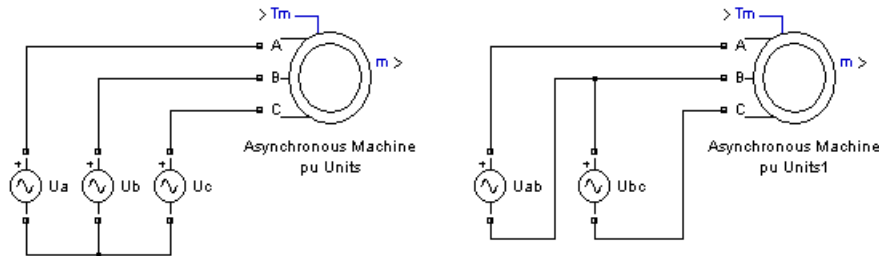
The next graph illustrates these results and shows the accuracy of the saturation model. The measured operating points fit well the curve that is plotted from the saturation parameters data.



You can observe the other effects of saturation on the stator currents by running the simulation with a blocked rotor or with many different values of load torque.

Assumptions and Limitations

- The Asynchronous Machine blocks do not include a representation of the saturation of leakage fluxes. Be careful when you connect ideal sources to the stator of the machine. If you choose to supply the stator via a three-phase, Y-connected infinite voltage source, you must use three sources connected in Y. However, if you choose to simulate a delta source connection, you must use only two sources connected in series.



- When you use Asynchronous Machine blocks in discrete systems, you might have to connect a small parasitic resistive load at the machine terminals to avoid numerical oscillations. Large sample times require larger loads. The optimum resistive load is proportional to the sample time. With a 25 μ s time step on a 60 Hz system, the minimum load is approximately 2.5% of the machine nominal power. For example, a 200 MVA asynchronous machine in a power system discretized with a 50 μ s sample time requires approximately 5% of resistive load or 10 MW. If the sample time is reduced to 20 μ s, a resistive load of 4 MW is sufficient.

Ports

The stator terminals of the Asynchronous Machine blocks are identified by the letters **A**, **B**, and **C**. The rotor terminals are identified by the letters **a**, **b**, and **c**. The neutral connections of the stator and rotor windings are not available. Three-wire Y connections are assumed.

Input

Tm — Mechanical torque
scalar

Mechanical torque at the machine's shaft, specified as a scalar. When the input is positive, the asynchronous machine behaves as a motor. When the input is negative, the asynchronous machine behaves as a generator.

When you use the Asynchronous Machine SI Units block, the input is a signal in N.m. When you use the Asynchronous Machine pu Units block, the input is a signal in pu.

Dependencies

To enable this port, in the **Configuration** tab, set the **Mechanical input** parameter to Torque **Tm**.

w — Machine speed
scalar

Machine speed in rad/s (for the Asynchronous Machine SI Units block) or in pu (for the Asynchronous Machine pu Units block), specified as a scalar.

Dependencies

To enable this port, in the **Configuration** tab, set the **Mechanical input** parameter to Speed **w**.

Output

m — Measurement signal
28-element vector

Measurement signals, returned as a vector. You can demultiplex these signals by using the Bus Selector block. The units are in SI or pu, depending on whether you use the Asynchronous Machine SI Units or Asynchronous Machine pu Units block. The cage 2 rotor signals return a null signal when the **Rotor type** parameter on the **Configuration** tab is set to Wound or Squirrel-cage.

Name	Definition	Units
iar	Rotor current ir_a	A or pu
ibr	Rotor current ir_b	A or pu
icr	Rotor current ir_c	A or pu
iqr	Rotor current iq	A or pu
idr	Rotor current id	A or pu
phiqr	Rotor flux phir_q	V.s or pu
phidr	Rotor flux phir_d	V.s or pu
vqr	Rotor voltage Vr_q	V or pu
vdr	Rotor voltage Vr_d	V or pu
iar2	Cage 2 rotor current ir_a	A or pu
ibr2	Cage 2 rotor current ir_b	A or pu
icr2	Cage 2 rotor current ir_c	A or pu
iqr2	Cage 2 rotor current iq	A or pu
idr2	Cage 2 rotor current id	A or pu
phiqr2	Cage 2 rotor flux phir_q	V.s or pu
phidr2	Cage 2 rotor flux phir_d	V.s or pu
ias	Stator current is_a	A or pu
ibs	Stator current is_b	A or pu
ics	Stator current is_c	A or pu
iqs	Stator current is_q	A or pu
ids	Stator current is_d	A or pu
phiqs	Stator flux phis_q	V.s or pu
phids	Stator flux phis_d	V.s or pu
vqs	Stator voltage vs_q	V or pu
vds	Stator voltage vs_d	V or pu
w	Rotor speed	rad/s
Te	Electromagnetic torque Te	N.m or pu
theta	Rotor angle thetam	rad

Conserving

A — Phase A stator terminal
specialized electrical

Specialized electrical conserving port associated with the phase A stator terminal.

B — Phase B stator terminal
specialized electrical

Specialized electrical conserving port associated with the phase B stator terminal.

C — Phase C stator terminal
specialized electrical

Specialized electrical conserving port associated with the phase C stator terminal.

a — Phase a rotor terminal
specialized electrical

Specialized electrical conserving port associated with the phase a rotor terminal.

Dependencies

This port applies only to the Asynchronous Machine SI Units block.

b — Phase b rotor terminal
specialized electrical

Specialized electrical conserving port associated with the phase b rotor terminal.

Dependencies

This port applies only to the Asynchronous Machine SI Units block.

c — Phase c rotor terminal
specialized electrical

Specialized electrical conserving port associated with the phase c rotor terminal.

Dependencies

This port applies only to the Asynchronous Machine SI Units block.

S — Machine rotor shaft
mechanical

Mechanical rotational conserving port associated with the machine rotor.

Dependencies

To enable this port, in the **Configuration** tab, set the **Mechanical input** parameter to Mechanical rotational port.

Parameters

Configuration

Rotor type — Rotor type
Wound | Squirrel-cage | Double squirrel-cage

Type of rotor. For the Asynchronous Machine SI Units block, the default value is Wound. For the Asynchronous Machine pu Units block, the default value is Squirrel-cage.

Squirrel-cage preset model — Electrical and mechanical parameters
No (default) | 01: 5 HP 460 V 60Hz 1750 RPM | 02: 10 HP 460 V 60Hz 1760 RPM | ...

Set of predetermined electrical and mechanical parameters for various asynchronous machine ratings of power (HP), phase-to-phase voltage (V), frequency (Hz), and rated speed (rpm) for single squirrel-cage machines.

Select one of the preset models to load the corresponding electrical and mechanical parameters. The preset models do not include predetermined saturation parameters. Choices are:

- 01: 5 HP 460 V 60Hz 1750 RPM
- 02: 10 HP 460 V 60Hz 1760 RPM
- 03: 20 HP 460 V 60Hz 1760 RPM
- 04: 50 HP 460 V 60Hz 1780 RPM
- 05: 100 HP 460 V 60Hz 1780 RPM
- 06: 150 HP 460 V 60Hz 1785 RPM
- 07: 200 HP 460 V 60Hz 1785 RPM
- 08: 5 HP 575 V 60Hz 1750 RPM
- 09: 10 HP 575 V 60Hz 1760 RPM
- 10: 20 HP 575 V 60Hz 1765 RPM
- 11: 50 HP 575 V 60Hz 1775 RPM
- 12: 100 HP 575 V 60Hz 1780 RPM
- 13: 150 HP 575 V 60Hz 1785 RPM
- 14: 200 HP 575 V 60Hz 1785 RPM
- 15: 5.4 HP (4KW) 400 V 50Hz 1430 RPM
- 16: 10 HP (7.5KW) 400 V 50Hz 1440 RPM
- 17: 20 HP (15KW) 400 V 50Hz 1460 RPM
- 18: 50 HP (37KW) 400 V 50Hz 1480 RPM
- 19: 100 HP (75KW) 400 V 50Hz 1484 RPM
- 20: 150 HP (110KW) 400 V 50Hz 1487 RPM
- 21: 215 HP (160KW) 400 V 50Hz 1487 RPM

Select **No** if you do not want to use a preset model, or if you want to modify some of the parameters of a preset model.

When you select a preset model, the electrical and mechanical parameters in the **Parameters** tab are dimmed. To start from a preset model and then modify machine parameters:

- 1 Select the preset model for which you want to initialize the parameters.
- 2 Change the **Preset model** parameter to **No**. This action does not change the machine parameters, but breaks the connection with the preset model.
- 3 Modify the machine parameters as you want.

Dependencies

To enable this parameter, set **Rotor type** to Squirrel-cage.

Double squirrel-cage preset model — Preset models
button

Click **Open parameter estimator** to open an interface to the `power_AsynchronousMachineParams` function that gives you access to preset models for double-cage asynchronous machines.

Mechanical input — Mechanical input

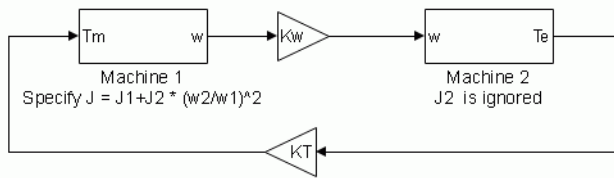
Torque T_m (default) | Speed w | Mechanical rotational port

Whether to represent the torque applied to the shaft or the rotor speed as a Simulink input of the block, or to represent the machine shaft by a Simscape rotational mechanical port.

Select **Torque T_m** to specify a torque input, in N.m or in pu, and expose the **T_m** port. The machine speed is determined by the machine inertia J (for the SI machine) or inertia constant H (for the pu machine) and by the difference between the applied mechanical torque T_m , and the internal electromagnetic torque, T_e . When the speed is positive, a positive torque signal indicates motor mode and a negative signal indicates generator mode.

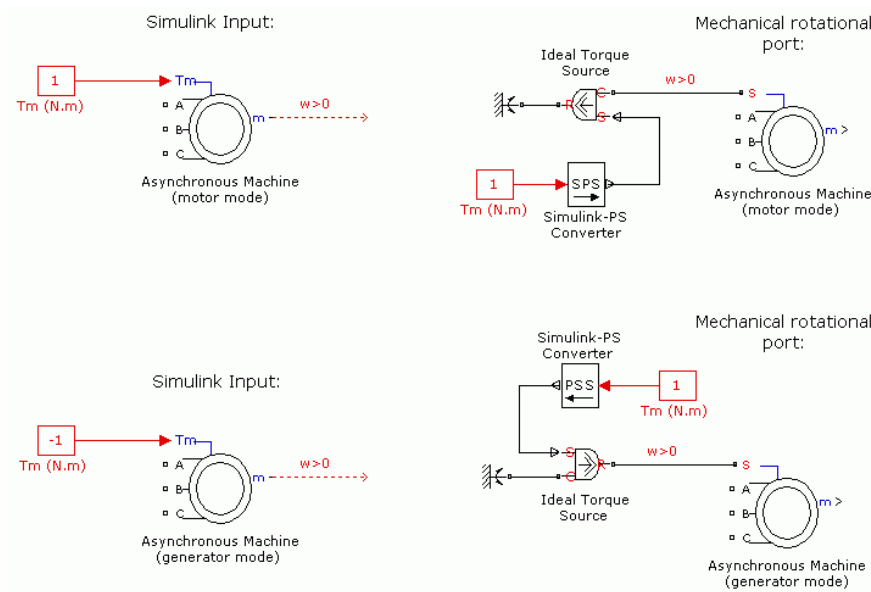
Select **Speed w** to specify a speed input, in rad/s or in pu, and expose the **w** port. The machine speed is imposed and the mechanical part of the model (the machine inertia J) is ignored. Using the speed as the mechanical input allows you to model a mechanical coupling between two machines.

The figure indicates how to model a stiff shaft interconnection in a motor-generator set when friction torque is ignored in machine 2. The speed output of machine 1 (the motor) is connected to the speed input of machine 2 (the generator), while machine 2 electromagnetic torque output T_e is applied to the mechanical torque input T_m of machine 1. The K_w factor takes into account the speed units of both machines (rad/s or pu) and gear box ratio w_2/w_1 . The K_T factor takes into account the torque units of both machines (N.m or pu) and machine ratings. Also, because inertia J_2 is ignored in machine 2, J_2 refers to the speed of machine 1 and must be added to machine 1 inertia J_1 .



Select **Mechanical rotational port** to expose a Simscape mechanical rotational port, **S** , that allows you to connect the machine shaft to other Simscape blocks that have mechanical rotational ports.

The figure indicates how to connect an Ideal Torque Source block from the Simscape library to the machine shaft to represent the machine in motor mode or in generator mode when the rotor speed is positive.



Reference frame — Reference frame for converting Rotor (default) | Stationary | Synchronous

Reference frame that is used to convert input voltages (abc reference frame) to the dq reference frame, and output currents (dq reference frame) to the abc reference frame. Choose from the following reference frame transformations:

- Rotor — Park transformation
- Stationary — Clarke or $\alpha\beta$ transformation
- Synchronous

The following relationships describe the abc-to-dq reference frame transformations applied to the asynchronous machine phase-to-phase voltages.

$$\begin{bmatrix} V_{qs} \\ V_{ds} \end{bmatrix} = \frac{1}{3} \begin{bmatrix} 2\cos\theta & \cos\theta + \sqrt{3}\sin\theta \\ 2\sin\theta & \sin\theta - \sqrt{3}\cos\theta \end{bmatrix} \begin{bmatrix} V_{abs} \\ V_{bcs} \end{bmatrix}$$

$$\begin{bmatrix} V'_{qr} \\ V'_{dr} \end{bmatrix} = \frac{1}{3} \begin{bmatrix} 2\cos\beta & \cos\beta + \sqrt{3}\sin\beta \\ 2\sin\beta & \sin\beta - \sqrt{3}\cos\beta \end{bmatrix} \begin{bmatrix} V'_{abr} \\ V'_{bcr} \end{bmatrix}$$

In the preceding equations, Θ is the angular position of the reference frame, while $\beta = \theta - \theta_r$ is the difference between the position of the reference frame and the position (electrical) of the rotor. Because the machine windings are connected in a three-wire Y configuration, there is no homopolar (0) component. This configuration also justifies that two line-to-line input voltages are used inside the model instead of three line-to-neutral voltages. The following relationships describe the dq-to-abc reference frame transformations applied to the asynchronous machine phase currents.

$$\begin{bmatrix} i_{as} \\ i_{bs} \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ \frac{-\cos\theta + \sqrt{3}\sin\theta}{2} & \frac{-\sqrt{3}\cos\theta - \sin\theta}{2} \end{bmatrix} \begin{bmatrix} i_{qs} \\ i_{ds} \end{bmatrix}$$

$$\begin{bmatrix} i'_{ar} \\ i'_{br} \end{bmatrix} = \begin{bmatrix} \cos\beta & \sin\beta \\ \frac{-\cos\beta + \sqrt{3}\sin\beta}{2} & \frac{-\sqrt{3}\cos\beta - \sin\beta}{2} \end{bmatrix} \begin{bmatrix} i'_{qr} \\ i'_{dr} \end{bmatrix}$$

$$i_{cs} = -i_{as} - i_{bs}$$

$$i'_{cr} = -i'_{ar} - i'_{br}$$

The table shows the values taken by Θ and β in each reference frame (Θ_e is the position of the synchronously rotating reference frame).

Reference Frame	Θ	β
Rotor	Θ_r	0
Stationary	0	$-\Theta_r$
Synchronous	Θ_e	$\Theta_e - \Theta_r$

The choice of reference frame affects the waveforms of all dq variables. It also affects the simulation speed, and in certain cases, the accuracy of the results. The following guidelines are suggested in [1]:

- Use the stationary reference frame if the stator voltages are either unbalanced or discontinuous and the rotor voltages are balanced (or 0).
- Use the rotor reference frame if the rotor voltages are either unbalanced or discontinuous and the stator voltages are balanced.
- Use either the stationary or synchronous reference frames if all voltages are balanced and continuous.

In the following situations, the **Reference frame** parameter is not editable and is set internally:

powergui Block Setting	Asynchronous Machine Block Setting	Reference frame Parameter
Simulation type is set to Phasor or Discrete Phasor	—	Synchronous
Simulation type is set to Discrete and the Automatically handle Discrete solver and Advanced tab solver settings of blocks parameter is selected	—	Rotor
Simulation type is set to Discrete	Discrete solver model is set to Trapezoidal robust or Backward Euler robust	Rotor

Dependencies

To enable this parameter, in the powergui block, set **Simulation type** to Continuous or Discrete and clear **Automatically handle Discrete solver and Advanced tab solver settings of blocks**. Additionally, on the **Advanced** tab, set **Discrete solver model** to Trapezoidal non iterative or Trapezoidal iterative (alg. loop).

Use signal names to identify bus labels — How to identify bus labels

off (default) | on

When this check box is selected, the measurement output uses the signal names to identify the bus labels. Select this option for applications that require bus signal labels to have only alphanumeric characters.

When this check box is cleared, the measurement output uses the signal definition to identify the bus labels. The labels contain nonalphanumeric characters that are incompatible with some Simulink applications.

Parameters for Asynchronous Machine SI Units Block

Tip This tab contains the electrical parameters of the machine. To estimate the electrical parameters of a double-cage asynchronous machine based on standard manufacturer specifications, use the `power_AynchronousMachineParams` function.

Nominal power, voltage (line-line), and frequency [$P_n(VA)$, $V_n(V_{rms})$, $f_n(Hz)$] — Nominal power, voltage, and frequency

[1.845e+04 400 50] (default) | three-element vector

Nominal apparent power P_n (VA), RMS line-to-line voltage V_n (V), and frequency f_n (Hz).

Vrotor/Vstator voltage ratio — Voltage ratio

1 (default) | positive scalar

Vrotor/Vstator voltage ratio of the wound-rotor asynchronous machine when the rotor is at standstill. Specifying this parameter allows you to get the desired rotor voltage without connecting a transformer at the rotor terminals.

Dependencies

To enable this parameter, in the **Configuration** tab, set **Rotor type** to Wound.

Stator resistance and inductance[$R_s(ohm)$ $Lls(H)$] — Stator resistance and leakage inductance

[0.5968 0.0003495] (default) | two-element vector

Stator resistance R_s (Ω) and leakage inductance Lls (H).

Rotor resistance and inductance [$R_r'(ohm)$ $Llr'(H)$] — Rotor resistance and leakage inductance

[0.6258 0.005473] (default) | two-element vector

Rotor resistance R_r' (Ω) and leakage inductance Llr' (H), both referred to the stator.

Dependencies

To enable this parameter, in the **Configuration** tab, set **Rotor type** to Wound or Squirrel-cage.

Cage 1 resistance and inductance [$R_{r1}'(ohm)$ $L_{lr1}'(H)$] — Rotor resistance and leakage inductance

[0.4155 0.002066] (default) | two-element vector

Rotor resistance R_{r1}' (Ω) and leakage inductance L_{lr1}' (H), both referred to the stator.

Dependencies

To enable this parameter, in the **Configuration** tab, set **Rotor type** to Double squirrel-cage.

Cage 2 resistance and inductance [$Rr2'$ (ohm) $Llr2'$ (H)] — Rotor resistance and leakage inductance

[0.4168 0.0003495] (default) | two-element vector

Rotor resistance $Rr2'$ (Ω) and leakage inductance $Llr2'$ (H), both referred to the stator.

Dependencies

To enable this parameter, in the **Configuration** tab, set **Rotor type** to Double squirrel-cage.

Mutual inductance Lm (H) — Magnetizing inductance

0.0354 (default) | scalar

Magnetizing inductance Lm (H).

Inertia, friction factor, pole pairs [J (kg.m²) F (N.m.s) p ()] — Inertia, friction factor, pole pairs

[0.05 0.005879 2] (default) | three-element vector

Combined machine and load inertia coefficient J (kg.m²), combined viscous friction coefficient F (N.m.s), and pole pairs p . The friction torque Tf is proportional to the rotor speed ω , ($Tf = F\omega$). Tf is expressed in N.m, F in N.m.s, and ω in rad/s.

Initial conditions — Initial slip s , electrical angle, stator current magnitude, phase angles, rotor current magnitude, and phase angles

[0 0 0 0 0 0 0 0] (default) | eight-element vector

Initial slip s , electrical angle Θ_e (degrees), stator current magnitude (A), and phase angles (degrees):

[slip, th, i_{as} , i_{bs} , i_{cs} , phase_{as}, phase_{bs}, phase_{cs}]

If the **Rotor type** parameter (in the **Configuration** tab) is set to Wound, you can also specify optional initial values for the rotor current magnitude (A), and phase angles (degrees):

[slip, th, i_{as} , i_{bs} , i_{cs} , phase_{as}, phase_{bs}, phase_{cs}, i_{ar} , i_{br} , i_{cr} , phase_{ar}, phase_{br}, phase_{cr}]

When the **Rotor type** parameter (in the **Configuration** tab) is set to Squirrel-cage, the initial conditions can be computed by the **Load Flow** tool or the **Machine Initialization** tool in the powergui block.

Simulate saturation — Simulate magnetic saturation

off (default) | on

Whether to simulate magnetic saturation of the rotor and stator iron.

Select this check box to provide the matrix of parameters for simulating the saturation.

Clear this check box to not model saturation in your simulation. In this case, the relationship between the stator current and the stator voltage is linear.

[i (Arms) ; v (VLL rms)] — No-load saturation curve parameters

[14.03593122, 27.81365428, 53.79336849, 72.68890987, 97.98006896, 148.6815601, 215.7428561, 302.9841135, 428.7778367 ; 230, 322, 414, 460, 506, 552, 598, 644, 690] (default) | matrix

No-load saturation curve parameters. The magnetic saturation of the stator and rotor iron (the saturation of the mutual flux) is modeled by a piecewise linear relationship specifying points of the no-load saturation curve. The first row of this matrix contains the values of stator currents. The second row contains values of corresponding terminal voltages (stator voltages). The first point (first column of the matrix) must be different from $[0, 0]$. This point corresponds to the point where the effect of saturation begins.

Click **Plot** to view the specified no-load saturation curve.

Dependencies

To enable this parameter, select **Simulate saturation**.

Parameters for Asynchronous Machine pu Units Block

Tip This tab contains the electrical parameters of the machine. To estimate the electrical parameters of a double-cage asynchronous machine based on standard manufacturer specifications, use the `power_AynchronousMachineParams` function.

Nominal power, voltage (line-line), and frequency [$P_n(VA)$, $V_n(V_{rms})$, $f_n(Hz)$] — Nominal power, voltage, and frequency

`[3730 460 60]` (default) | three-element vector

Nominal apparent power P_n (VA), RMS line-to-line voltage V_n (V), and frequency f_n (Hz).

Vrotor/Vstator voltage ratio — Voltage ratio

1 (default) | positive scalar

Vrotor/Vstator voltage ratio of the wound-rotor asynchronous machine when the rotor is at standstill. Specifying this parameter allows you to get the desired rotor voltage without connecting a transformer at the rotor terminals.

Dependencies

To enable this parameter, in the **Configuration** tab, set **Rotor type** to Wound.

Stator resistance and inductance [R_s , L_{ls}] (pu) — Stator resistance and leakage inductance

`[0.01965 0.0397]` (default) | two-element vector

Stator resistance R_s (pu) and leakage inductance L_{ls} (pu).

Rotor resistance and inductance [R_r' , L_{lr}'] (pu) — Rotor resistance and leakage inductance

`[0.01909 0.0397]` (default) | two-element vector

Rotor resistance R_r' (pu) and leakage inductance L_{lr}' (pu), both referred to the stator.

Dependencies

To enable this parameter, in the **Configuration** tab, set **Rotor type** to Wound or Squirrel-cage.

Cage 1 resistance and inductance [R_{r1}' , L_{lr1}'] (pu) — Rotor resistance and leakage inductance

`[0.01909 0.0397]` (default) | two-element vector

Rotor resistance R_{r1}' (pu) and leakage inductance L_{lr1}' (pu), both referred to the stator.

Dependencies

To enable this parameter, in the **Configuration** tab, set **Rotor type** to Double squirrel-cage.

Cage 2 resistance and inductance [$Rr2'$, $Llr2'$] (pu) — Rotor resistance and leakage inductance
[0.01909 0.0397] (default) | two-element vector

Rotor resistance $Rr2'$ (pu) and leakage inductance $Llr2'$ (pu), both referred to the stator.

Dependencies

To enable this parameter, in the **Configuration** tab, set **Rotor type** to Double squirrel-cage.

Mutual inductance L_m (pu) — Magnetizing inductance
1.354 (default) | scalar

Magnetizing inductance L_m (pu).

Inertia constant, friction factor, pole pairs [$H(s)$ $F(pu)$ $p()$] — Inertia constant, friction factor, pole pairs
[0.09526 0.05479 2] (default) | three-element vector

Inertia constant H (s), combined viscous friction coefficient F (pu), and pole pairs p .

Initial conditions — Initial slip s , electrical angle, stator current magnitude, phase angles, rotor current magnitude, and phase angles
[1,0 0,0,0 0,0,0] (default) | eight-element vector

Initial slip s , electrical angle Θ_e (degrees), stator current magnitude (pu), and phase angles (degrees):

[slip, th, i_{as} , i_{bs} , i_{cs} , phase $_{as}$, phase $_{bs}$, phase $_{cs}$]

If the **Rotor type** parameter (in the **Configuration** tab) is set to Wound, you can also specify optional initial values for the rotor current magnitude (pu), and phase angles (degrees):

[slip, th, i_{as} , i_{bs} , i_{cs} , phase $_{as}$, phase $_{bs}$, phase $_{cs}$, i_{ar} , i_{br} , i_{cr} , phase $_{ar}$, phase $_{br}$, phase $_{cr}$]

When the **Rotor type** parameter (in the **Configuration** tab) is set to Squirrel-cage, the initial conditions can be computed by the **Load Flow** tool or the **Machine Initialization** tool in the powergui block.

Simulate saturation — Simulate magnetic saturation
off (default) | on

Whether to simulate the magnetic saturation of the rotor and stator iron.

Select this check box to provide the matrix of parameters for simulating the saturation.

Clear this check box to not model saturation in your simulation. In this case, the relationship between the stator current and the stator voltage is linear.

[i ; v] (pu) — No-load saturation curve parameters
[0.212,0.4201,0.8125,1.0979,1.4799,2.2457,3.2586,4.5763,6.4763 ;
0.5,0.7,0.9,1,1.1,1.2 ,1.3,1.4,1.5] (default) | matrix

No-load saturation curve parameters. The magnetic saturation of the stator and rotor iron (the saturation of the mutual flux) is modeled by a piecewise linear relationship specifying points of the

no-load saturation curve. The first row of this matrix contains the values of stator currents. The second row contains values of corresponding terminal voltages (stator voltages). The first point (first column of the matrix) must be different from $[0, 0]$. This point corresponds to the point where the effect of saturation begins.

Click **Plot** to view the specified no-load saturation curve.

Dependencies

To enable this parameter, select **Simulate saturation**.

Advanced

To enable the **Advanced** tab, in the powergui block, set **Simulation type** to **Discrete** and clear **Automatically handle Discrete solver and Advanced tab solver settings of blocks**.

Sample time (-1 for inherited) — Sample time for block

-1 (default) | scalar

Sample time used by the block. To inherit the sample time specified in the powergui block, set this parameter to **-1**.

Discrete solver model — Integration method

Trapezoidal non iterative (default) | Trapezoidal iterative (alg. loop) |
Trapezoidal robust | Backward Euler robust

Integration method used by the block when the **Solver type** parameter of the powergui block is set to **Discrete**.

The **Discrete solver model** is automatically set to **Trapezoidal robust** when you select the **Automatically handle Discrete solver and Advanced tab solver settings of blocks** parameter of the powergui block.

The **Trapezoidal non iterative** and **Trapezoidal iterative (alg. loop)** methods are no longer recommended for discretizing the Asynchronous Machine blocks. **Trapezoidal non iterative** requires you to add a non-negligible shunt load at the machine terminals to guarantee simulation stability, and **Trapezoidal iterative (alg. loop)** may fail to converge and cause the simulation to stop when the number of machines increases in the model.

The **Trapezoidal robust** and **Backward Euler robust** methods allow you to eliminate the need to use parasitic loads and simulate a machine without loads. To eliminate topological errors of machines connected to an inductive circuit (for example, a circuit breaker connected in series with the machine) the machine models a negligible internal load of 0.01% of nominal power.

The **Trapezoidal robust** method is slightly more accurate than the **Backward Euler robust** method, especially when the model is simulated at larger sample times. The **Trapezoidal robust** method may produce slight damped numerical oscillations on machine voltage in no-load conditions, while the **Backward Euler robust** method prevents oscillations and maintains good accuracy.

For more information on what method you should use in your application, see *Simulating Discretized Electrical Systems*.

Load Flow

The parameter on this tab is used by the **Load Flow** tool of the powergui block. These load flow parameter is used for model initialization only. It has no impact on the block model or on the simulation performance.

Mechanical power (W) — Mechanical power applied to machine shaft
scalar

Mechanical power applied to the machine shaft, in watts. When the machine operates in motor mode, specify a positive value. When the machine operates in generator mode, specify a negative value.

For the Asynchronous Machine SI Units block, the default value is $1.492\text{e}+006$. For the Asynchronous Machine pu Units block, the default value is 0.

Version History

Introduced before R2006a

References

- [1] Krause, P.C., O. Wasynczuk, and S.D. Sudhoff, *Analysis of Electric Machinery*, IEEE Press, 2002.
- [2] Mohan, N., T.M. Undeland, and W.P. Robbins, *Power Electronics: Converters, Applications, and Design*, John Wiley & Sons, Inc., New York, 1995, Section 8.4.1.

Extended Capabilities

C/C++ Code Generation

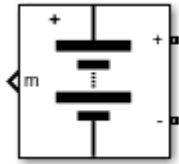
Generate C and C++ code using Simulink® Coder™.

See Also

powergui

Battery

Generic battery model



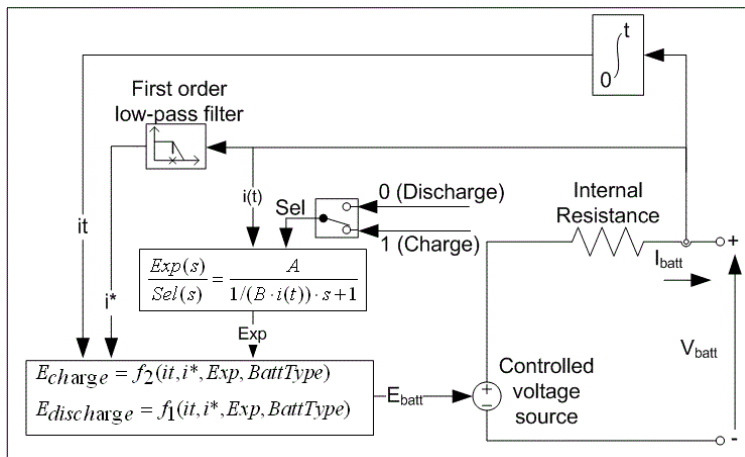
Libraries:

Simscape / Electrical / Specialized Power Systems / Sources

Description

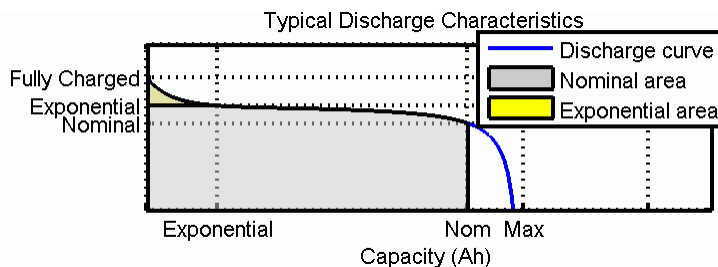
The Battery block implements a generic dynamic model that represents most popular types of rechargeable batteries.

This figure shows the equivalent circuit that the block models.



Charge and Discharge Characteristics

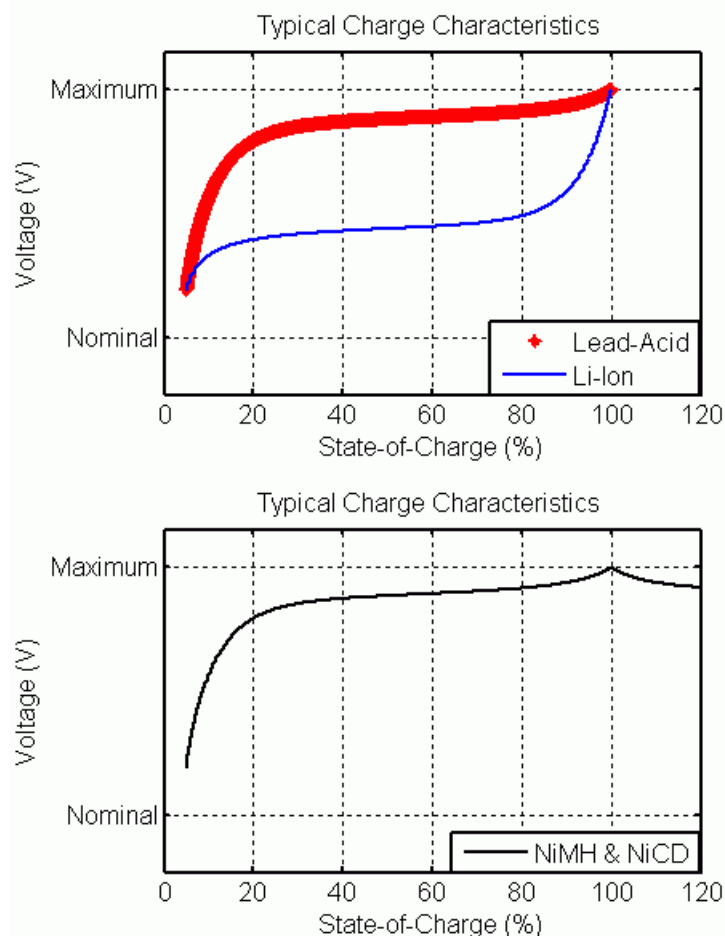
The circuit parameters can be modified to represent a specific battery type and its discharge characteristics. A typical discharge curve consists of three sections.



The first section represents the exponential voltage drop when the battery is charged. The width of the drop depends on the battery type. The second section represents the charge that can be extracted

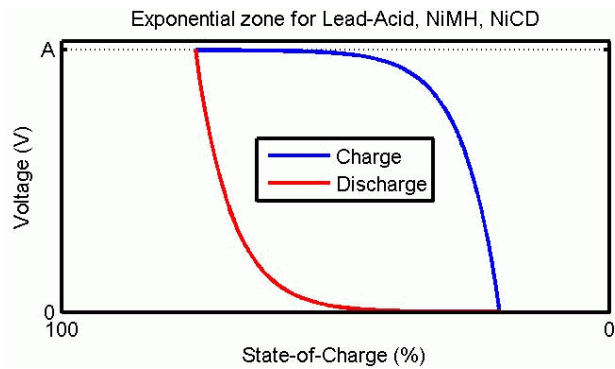
from the battery until the voltage drops below the battery nominal voltage. Finally, the third section represents the total discharge of the battery, when the voltage drops rapidly.

When the battery current is negative, the battery recharges, following a charge characteristic.



The model parameters are derived from the discharge characteristics. The discharging and charging characteristics are assumed to be the same.

The Exp(s) transfer function represents the hysteresis phenomenon for the lead-acid, nickel-cadmium (NiCD), and nickel-metal hydride (NiMH) batteries during the charge and discharge cycles. The exponential voltage increases when a battery is charging, regardless of the battery's state of charge. When the battery is discharging, the exponential voltage decreases immediately.



The state of charge (SOC) for a battery is a measure of battery's charge, expressed as a percent of the full charge. The depth of discharge (DOD) is the numerical complement of the SOC, such that $DOD = 100\% - SOC$.

For example, if the SOC is:

- 100% — The battery is fully charged and the DOD is 0%.
- 75% — The battery is 3/4 charged and the DOD is 25%.
- 50% — The battery is 1/2 charged and the DOD is 50%.
- 0% — The battery has 0 charge and the DOD is 100%.

Model Validation

Experimental validation of the model shows a maximum error of 5% (when SOC is between 10% and 100%) for the charge (when the current is 0 through 2 C) and discharge (when the current is 0 through 5 C) dynamics.

Parameterization

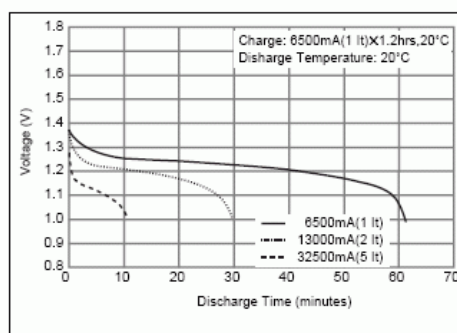
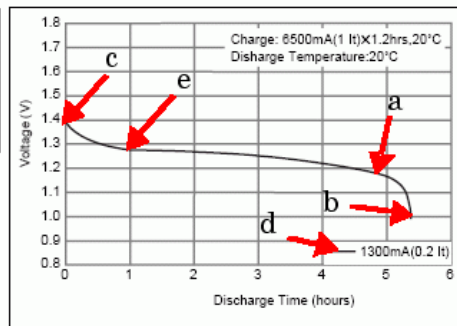
Extract Battery Parameters from Data Sheets

This figure shows detailed parameters extracted from the Panasonic NiMH-HHR650D battery data sheet.

Specifications

		mm	inch	
Diameter		33.0+0/-0.1	1.3+0/-0.04	
Height		61+0/-1.5	2.40+0/-0.08	
Approximate Weight		Grams	Ounces	
		170	6.0	
Nominal Voltage		1.2V		
Discharge Capacity*	Average**	6800 mAh		
	Rated (Min.)	6500 mAh		
Approx. Internal impedance at 1000Hz at charged state.		2mΩ		
Charge		Standard	650mA (0.1It) x 16hrs.	
		Rapid	6500mA (1It) x 1.2 hrs.	
Ambient Temperature	Charge	Standard	°C	°F
			0°C to 45°C	32°F to 113°F
		Rapid	0°C to 40°C	32°F to 104°F
	Storage	Discharge	-10°C to 65°C	14°F to 149°F
		< 2 years	-20°C to 45°C	-4°F to 113°F
		< 6 months	-20°C to 55°C	-4°F to 131°F

Typical Discharge Characteristics

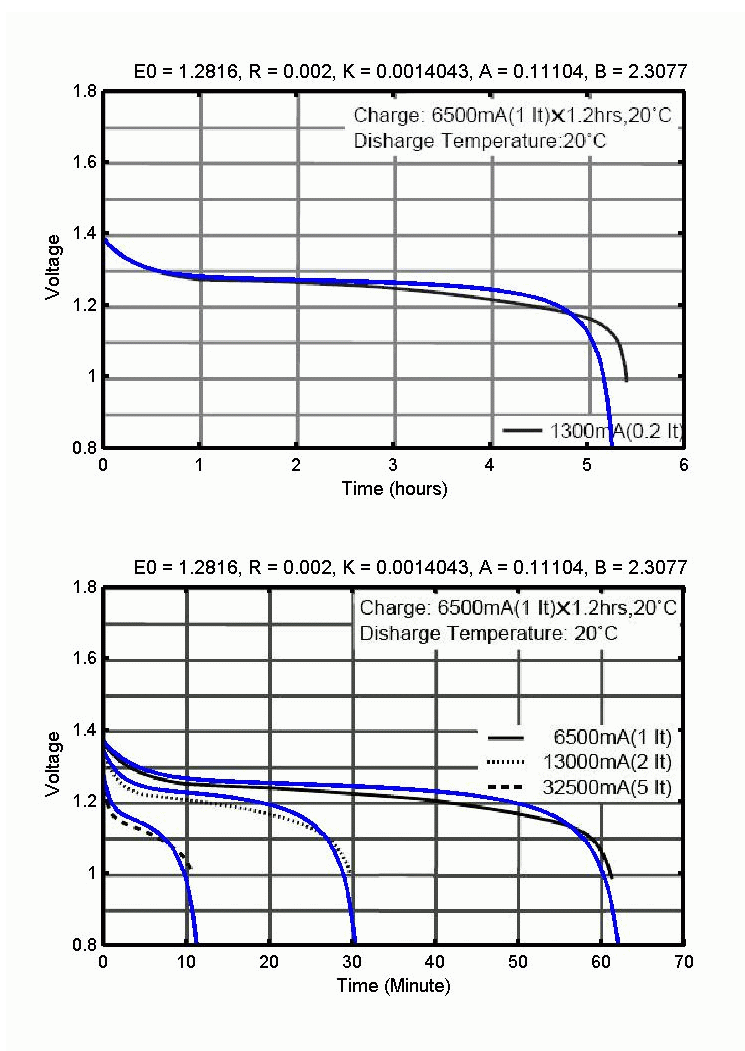


You can obtain the rated capacity and the internal resistance from the specification tables. The other detailed parameters are derived from the Typical Discharge Characteristics plot.

Parameter	Value
Rated Capacity	6.5 Ah
Internal Resistance	2 mΩ
Nominal Voltage (a)	1.18 V
Rated Capacity	6.5 Ah
Maximum Capacity (b)	7 Ah (5.38 h * 1.3 A)
Fully Charged Voltage (c)	1.39 V
Nominal Discharge Current (d)	1.3 A
Capacity @ Nominal Voltage (a)	6.25 Ah
Exponential Voltage (e)	1.28 V
Exponential Capacity (e)	1.3 Ah

These parameters are approximate and depend on the precision of the points obtained from the discharge curve.

The discharge curves you obtain from these parameters, which are marked by dotted lines in the following figures, are similar to the data sheet curves.



Model cells in Series and/or in Parallel

To model a series and/or parallel combination of cells based on the parameters of a single cell, use the parameter transformation shown in the following table can be used. The Nb_ser variable corresponds to the number of cells in series, and Nb_par corresponds to the number of cells in parallel.

Parameter	Value
Nominal voltage	$1.18 * Nb\_ser$
Rated capacity	$6.5 * Nb\_par$
Maximum capacity	$7 * Nb\_par$
Fully charged voltage	$1.39 * Nb\_ser$
Nominal discharge current	$1.3 * Nb\_par$
Internal resistance	$0.002 * Nb\_ser / Nb\_par$
Capacity at nominal voltage	$6.25 * Nb\_par$

Parameter	Value
Exponential zone	$1.28 * Nb\_ser, 1.3 * Nb\_par$

Equations

For the lead-acid battery type, the model uses these equations.

- Discharge Model ($i^* > 0$)

$$f_1(it, i^*, i, Exp) = E_0 - K \cdot \frac{Q}{Q - it} \cdot i^* - K \cdot \frac{Q}{Q - it} \cdot it + \text{Laplace}^{-1} \left(\frac{Exp(s)}{Sel(s)} \cdot 0 \right)$$

- Charge Model ($i^* < 0$)

$$f_2(it, i^*, i, Exp) = E_0 - K \cdot \frac{Q}{it + 0.1 \cdot Q} \cdot i^* - K \cdot \frac{Q}{Q - it} \cdot it + \text{Laplace}^{-1} \left(\frac{Exp(s)}{Sel(s)} \cdot \frac{1}{s} \right)$$

with $it > 0$ during charging.

For the lithium-ion battery type, the model uses these equations.

- Discharge Model ($i^* > 0$)

$$f_1(it, i^*, i) = E_0 - K \cdot \frac{Q}{Q - it} \cdot i^* - K \cdot \frac{Q}{Q - it} \cdot it + A \cdot \exp(-B \cdot it)$$

- Charge Model ($i^* < 0$)

$$f_2(it, i^*, i) = E_0 - K \cdot \frac{Q}{it + 0.1 \cdot Q} \cdot i^* - K \cdot \frac{Q}{Q - it} \cdot it + A \cdot \exp(-B \cdot it)$$

For the nickel-cadmium and nickel-metal-hydride battery types, the model uses these equations.

- Discharge Model ($i^* > 0$)

$$f_1(it, i^*, i, Exp) = E_0 - K \cdot \frac{Q}{Q - it} \cdot i^* - K \cdot \frac{Q}{Q - it} \cdot it + \text{Laplace}^{-1} \left(\frac{Exp(s)}{Sel(s)} \cdot 0 \right)$$

- Charge Model ($i^* < 0$)

$$f_2(it, i^*, i, Exp) = E_0 - K \cdot \frac{Q}{|it| + 0.1 \cdot Q} \cdot i^* - K \cdot \frac{Q}{Q - it} \cdot it + \text{Laplace}^{-1} \left(\frac{Exp(s)}{Sel(s)} \cdot \frac{1}{s} \right).$$

In the equations:

- E_{Batt} is the nonlinear voltage, in V.
- E_0 is the constant voltage, in V.
- $Exp(s)$ is the exponential zone dynamics, in V.
- $Sel(s)$ represents the battery mode. $Sel(s) = 0$ during battery discharge, $Sel(s) = 1$ during battery charging.
- K is the polarization constant, in V/Ah, or polarization resistance, in Ohms.
- i^* is the low-frequency current dynamics, in A.
- i is the battery current, in A.
- it is the extracted capacity, in Ah.

- Q is the maximum battery capacity, in Ah.
- A is the exponential voltage, in V.
- B is the exponential capacity, in Ah⁻¹.

Limitations and Assumptions

Limitations

- The minimum no-load battery voltage is 0 V and the maximum battery voltage is equal to $2 \times E_0$.
- The minimum capacity of the battery is 0 Ah and the maximum capacity is $Q_{\max}$.

Assumptions

- The internal resistance is assumed to be constant during the charge and discharge cycles and does not vary with the amplitude of the current.
- The parameters of the model are derived from the discharge characteristics. The discharging and charging characteristics are assumed to be the same.
- The capacity of the battery does not change with the amplitude of the current (there is no Peukert effect).
- The self-discharge of the battery is not represented. It can be represented by adding a large resistance in parallel with the battery terminals.
- The battery has no memory effect.

Ports

Output

m — Battery temperature, state-of-charge, current, and voltage

Simulink signal | vector

Output vector of signals for the battery temperature, state-of-charge, current and voltage. To demultiplex the signals, you can use a Bus Selector block.

Signal	Definition	Units
Cell Temperature	The cell or internal temperature	°C
SOC	Battery SOC, represented as a percentage (between 0 and 100%). The SOC is 100% for a fully charged battery and 0% for an empty battery. The SOC is calculated as: $SOC = 100 \left(1 - \frac{1}{Q} \int i(t) dt \right).$	%
Current	Battery current	A
Voltage	Battery voltage	V

Conserving

+ — Positive terminal
specialized electrical

Specialized electrical conserving port associated with the battery's positive terminal.

- — Negative terminal
specialized electrical

Specialized electrical conserving port associated with the battery's negative terminal.

Parameters

Parameters

Type — Battery model

Lithium-Ion (default) | Lead-Acid | Nickel-Cadmium | Nickel-Metal-Hydrate

Battery model. The block provides predetermined charge behavior for four battery types. For the Lithium-Ion battery, the block provides models for simulating temperature and aging effects.

Nominal voltage (V) — Nominal voltage

7.2 (default) | positive scalar

Nominal voltage, V_{nom} , of the battery, in V. The nominal voltage represents the end of the linear zone of the discharge characteristics.

Rated capacity (Ah) — Rated capacity

5.4 (default) | positive scalar

Rated capacity, Q_{rated} , of the battery, in Ah. The rated capacity is the minimum effective capacity of the battery.

Initial state-of-charge (%) — Initial SOC

100 (default)

State-of-charge (SOC) of the battery, expressed as a percentage of the maximum potential charge, at the beginning of simulation. An SOC of 100% indicates a fully charged battery and 0% indicates an empty battery.

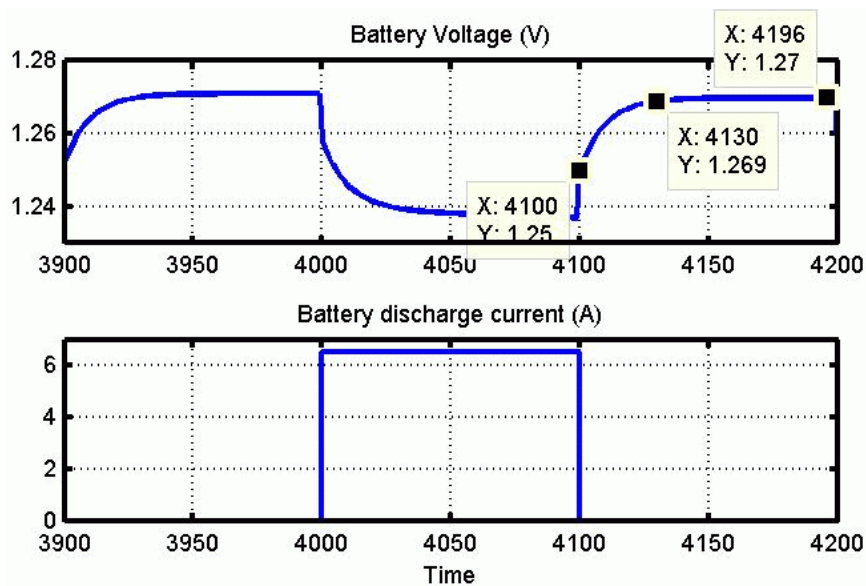
The specified value does not affect the discharge curve that the block generates if, in the **Discharge** settings, you click **Plot**.

Battery response time (s) — Response time

30 (default) | nonnegative scalar

Response time of the battery, in s, at 95% of the final value. This value represents the voltage dynamics and can be observed when a current step is applied.

The plots show the voltage and discharge current for a battery with a response time of 30 s.



Discharge

Determined from the nominal parameters of the battery — Discharge parameter determination on (default) | off

Select to have the block determine the parameters in the **Discharge** settings based on the values specified for the parameters in the **Parameters** settings.

Dependencies

Selecting this parameter disables the parameters in the **Discharge** settings.

Maximum capacity (Ah) — Maximum amp-hour capacity
5.4 (default) | positive scalar

Maximum theoretical capacity, Q , when a discontinuity occurs in the battery voltage, in Ah. This value is generally equal to 105% of the rated capacity.

Cut-off Voltage (V) — Cut-off voltage
5.4 (default) | positive scalar

Minimum allowable battery voltage, in V. This voltage represents the end of the discharge characteristics. At the cut-off voltage, the battery is fully discharged.

Fully charged voltage (V) — Fully charged voltage
8.3807 (default) | positive scalar

Fully charged voltage, V_{full} , for a given discharge current. The fully charged voltage is not the no-load voltage.

Nominal discharge current (A) — Nominal discharge current
2.3478 (default) | positive scalar

Nominal discharge current, in A, for which the discharge curve is measured.

For example, a typical discharge current for a 1.5-Ah NiMH battery is 20% of the rated capacity: $(0.2 * 1.5 \text{ Ah} / 1 \text{ h} = 0.3 \text{ A})$.

Internal resistance (Ohms) — Internal resistance

0.013333 (default) | positive scalar

Internal resistance of the battery, in ohms. When a preset model is used, a generic value is loaded that corresponds to 1% of the nominal power (nominal voltage multiplied by the battery rated capacity). The resistance is constant during the charge and the discharge cycles and does not vary with the amplitude of the current.

Capacity (Ah) at nominal voltage — Amp-hour capacity at nominal voltage

4.8835 (default) | positive scalar

Capacity, Q_{nom} , extracted from the battery until the voltage drops under the nominal voltage. This value should be between Q_{exp} and Q_{max} .

Exponential zone [Voltage (V), Capacity (Ah)] — Exponential zone voltage and capacity

[7.7788 0.2653] (default) | nonnegative vector

Voltage, V_{exp} , and the capacity, Q_{exp} , that correspond to the end of the exponential zone. The voltage should be between V_{nom} and V_{full} . The capacity should be between 0 and Q_{nom} .

Version History

Introduced in R2008a

R2024b: Temperature and aging effects have been removed

The Battery block no longer models the temperature and aging effects.

As a result of this change, this port and these parameters have been removed:

Port

- **Ta**

Parameters

- **Simulate temperature effects**
- **Use a preset battery**
- **Simulating aging effects**
- **Discharge current [i1, i2, i3,...] (A)**
- **Units**
- **Plot**
- **Initial cell temperature (deg, C)**
- **Nominal ambient temperature T1 (deg, C)**
- **Second ambient temperature T2 (deg, C)**
- **Maximum capacity (Ah)**

- **Initial discharge voltage (V)**
- **Voltage at 90% maximum capacity (V)**
- **Exponential zone [Voltage (V), Capacity (Ah)]**
- **Thermal resistance, cell-to-ambient (deg, C/W)**
- **Thermal time constant, cell-to-ambient (s)**
- **Heat loss difference [charge vs. discharge] (W)]**
- **Initial battery age (Equivalent full cycles)**
- **Aging model sampling time (s)**
- **Ambient temperature Ta1 (deg, C)**
- **Capacity at EOL (End Of Life) (Ah)**
- **Internal resistance at EOL (Ohms)**
- **Charge current (nominal, maximum) [Ic (A), Icmx (A)]**
- **Discharge current (nominal, maximum) [Id (A), Idmx (A)]**
- **Cycle life at 100% DOD, Ic and Id (Cycles)**
- **Cycle life at 25% DOD, Ic and Id (Cycles)**
- **Cycle life at 100% DOD, Ic and Idmx (Cycles)**
- **Cycle life at 100% DOD, Icmx and Id (Cycles)**
- **Ambient temperature Ta2 (deg, C)**

References

- [1] Omar N., M. A. Monem, Y. Firouz, J. Salminen, J. Smekens, O. Hegazy, H. Gaulous, G. Mulder, P. Van den Bossche, T. Coosemans, and J. Van Mierlo. "Lithium iron phosphate based battery — Assessment of the aging parameters and development of cycle life model." *Applied Energy*, Vol. 113, January 2014, pp. 1575-1585.
- [2] Saw, L.H., K. Somasundaram, Y. Ye, and A.A.O. Tay, "Electro-thermal analysis of Lithium Iron Phosphate battery for electric vehicles." *Journal of Power Sources*. Vol. 249, pp. 231-238.
- [3] Tremblay, O., L.A. Dessaint, "Experimental Validation of a Battery Dynamic Model for EV Applications." *World Electric Vehicle Journal*. Vol. 3, May 13-16, 2009.
- [4] Zhu, C., X. Li, L. Song, and L. Xiang, "Development of a theoretically based thermal model for lithium ion battery pack." *Journal of Power Sources*. Vol. 223, pp. 155-164.

Extended Capabilities

C/C++ Code Generation

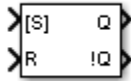
Generate C and C++ code using Simulink® Coder™.

See Also

AC Voltage Source | DC Voltage Source

Bistable

Implement prioritized S-R flip-flop (bistable multivibrator)



Library

Simscape / Electrical / Specialized Power Systems / Control

Description

The Bistable block implements the following logic:

- When S input is true and R input is false, the flip-flop goes to the Set state. This is the first stable state, where Q is true.
- When R is true and S is false, the flip-flop goes to the reset state. This is the second stable state, where Q is false.
- When both S and R are true, the flip-flop goes to the prioritized state defined by the **Select priority** parameter.
- When both S and R are false, the flip-flop stays in its previous state.

Parameters

Select priority

Specify the prioritized state of Q. Choices are **Set** (default) or **Reset**.

Initial condition (state of Q)

Specify the initial value of the output Q. Default is 0.

Sample time (-1 for inherited)

Specify the sample time of the block, in seconds. Set to 0 to implement a continuous block. Default is 0.

Inputs and Outputs

S, [S]

The Set input. Must be Boolean. When the **Select priority** parameter value is **Set**, the input is named [S].

R, [R]

The Reset input. Must be Boolean. When the **Select priority** parameter value is **Reset**, the input is named [R].

Q

Bistable output. The signal is Boolean.

!Q

Bistable complement output. The signal is Boolean.

Characteristics

Direct Feedthrough	Yes
Sample Time	Specified in the Sample Time parameter
Scalar Expansion	Yes, of parameters
Dimensionalized	Yes
Zero-Crossing Detection	No

Examples

The `power_Logic` example shows the operation of the Bistable block.

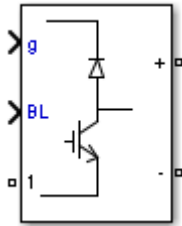
The model sample time is parameterized with the variable `Ts` (default value `Ts = 50e-3`). To simulate a continuous Bistable block, specify `Ts = 0` in the MATLAB® Command Window.

Version History

Introduced in R2013a

Boost Converter

Implement boost power converter



Libraries:

Simscape / Electrical / Specialized Power Systems / Power Electronics

Description

The Boost Converter block implements a boost power converter. You can choose from four model types:

- **Switching devices** — The converter is modeled with IGBT/diode pairs controlled by firing pulses produced by a PWM generator. This model provides the most accurate simulation results.
- **Switching function** — The converter is modeled by a switching-function model. The switches are replaced with two voltage sources and two diodes on the AC side and with two current sources on the DC side.

The converter is controlled by firing pulses produced by a PWM generator (0/1 signals) or by firing pulses averaged over a specified period (PWM averaging: signals between 0 and 1). Both modes of operation produce harmonics normally generated by a PWM-controlled converter and also correctly simulate rectifying operation as well as blanking time. This model type is well-suited for real-time simulation.

- **Average model (D-controlled)** — The converter is modeled using a switching-function model directly controlled by the duty cycle signal ($0 < D < 1$). A PWM generator is not required.
- **Average model (No rectifier mode)** — The block uses the voltage source directly controlled by the reference voltage to model the converter. The model does not require a PWM generator and does not simulate the rectifier mode. This setting provides the fastest simulations.

When doing real-time simulation, use the **Switching function** setting with the firing pulses averaged, or the **Average model (D-controlled)** or **Average model (No rectifier mode)** settings.

Ports

Input

g — Gating signal to control converter
vector of six elements

Vectorized gating signal to control the converter. The gating signal contains six firing pulses. The first two pulses control the Q1 and Q2 switching devices (phase A of the converter), pulses three and four

control the Q3 and Q4 switching devices (phase B of the converter), and the last two pulses control the Q5 and Q6 switching devices (phase C of the converter).

Dependencies

To enable this port, set **Model type** to `Switching devices` or `Switching function`.

D — Signal to control converter
scalar in the range [0,1]

The duty cycle signal (a value between 0 and 1) used to control the converter.

Dependencies

To enable this port, set **Model type** to `Average model (D-controlled)`.

Uref — Signal to control converter
vector of three elements

Vectorized signal to control the converter. The signal contains three reference voltages (one for each phase).

Dependencies

To enable this port, set **Model type** to `Average model (No rectifier mode)`.

BL — Input to block firing pulses
scalar

Input port to block all firing pulses to the converter. To block all firing pulses to the converter, apply a signal value of 1 to this port.

Conserving

+ — Positive terminal
specialized electrical

Specialized electrical conserving port associated with the positive terminal.

- — Negative terminal
specialized electrical

Specialized electrical conserving port associated with the negative terminal.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Model type — Model type
`Switching devices (default)` | `Switching function` | `Average model (D-controlled)` | `Average model (No rectifier mode)`

Select one of these values:

- **Switching devices** — The converter is modeled with IGBT/diode pairs controlled by firing pulses produced by a PWM generator.

- **Switching function** — The converter is modeled by a switching-function model. The switches are replaced with two voltage sources and two diodes on the AC side and with two current sources on the DC side.
- **Average model (D-controlled)** — The converter is modeled using a switching-function model directly controlled by the duty cycle signal ($0 < D < 1$). A PWM generator is not required.
- **Average model (D-controlled, no rectifier mode)** — The block uses the voltage source directly controlled by the reference voltage to model the converter. The model does not require a PWM generator and does not simulate the rectifier mode. This setting provides the fastest simulations.

Device on-state resistance (Ohms) — Internal resistance of switching devices

1e-3 (default) | positive scalar

Internal resistance of the switching devices, in ohms.

Dependencies

To enable this parameter, set **Model type** to **Switching devices**.

Snubber resistance (Ohms) — Snubber resistance

1e6 (default) | positive scalar

Snubber resistance, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to **Switching devices**.

Snubber capacitance (F) — Snubber capacitance

inf (default) | nonnegative scalar

Snubber capacitance, in farads. Set the snubber capacitance to `0` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to **Switching devices**.

Forward voltages [Device Vf (V), Diode Vfd (V)] — Forward voltage of switching device and diode

[0, 0] (default) | vector of two nonnegative values

Forward voltage of the switching device and of the diode, in volts.

Dependencies

To enable this parameter, set **Model type** to **Switching devices**.

Diode on-state resistance (Ohms) — Internal resistance of diodes

1e-3 (default) | positive scalar

Internal resistance of the diodes, in ohms.

Dependencies

To enable this parameter, set **Model type** to **Switching function** or **Average model (D-controlled)**.

Diode snubber resistance (Ohms) — Snubber resistance of diodes`1e6 (default) | positive scalar`

Snubber resistance of the diodes, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to `Switching function` or `Average model (D-controlled)`.

Diode snubber capacitance (F) — Snubber capacitance of diodes`inf (default) | nonnegative scalar`

Snubber capacitance of the diodes, in farads. Set the snubber capacitance to `0` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to `Switching function` or `Average model (D-controlled)`.

Diode forward voltage (V) — Forward voltage across diode`1e-3 (default) | nonnegative scalar`

Forward voltage, in volts, across the diode when it is conducting.

Dependencies

To enable this parameter, set **Model type** to `Switching function` or `Average model (D-controlled)`.

DC side snubber resistance (Ohms) — Snubber resistance across current sources`inf (default) | positive scalar`

Snubber resistance across the two current sources, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to `Switching function`, `Average model (D-controlled)`, or `Average model (No rectifier mode)`.

AC side source resistance (Ohms) — Source resistance on AC side`1e-3 (default) | positive scalar`

Source resistance on the AC side, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to `Average model (No rectifier mode)`.

Version History

Introduced in R2015b

Extended Capabilities

C/C++ Code Generation

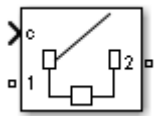
Generate C and C++ code using Simulink® Coder™.

See Also

Buck Converter | Full-Bridge Converter | Two-Level Converter

Breaker

Implement circuit breaker opening at current zero crossing



Libraries:
Simscape / Electrical / Specialized Power Systems / Power Grid Elements

Description

The Breaker block implements a circuit breaker where the opening and closing times can be controlled either from an external Simulink signal (external control mode), or from an internal control timer (internal control mode).

A series Rs-Cs snubber circuit is included in the model. It can be connected to the circuit breaker. If the Breaker block happens to be in series with an inductive circuit, an open circuit or a current source, you must use a snubber.

When the Breaker block is set in external control mode, a Simulink input appears on the block icon. The control signal connected to the Simulink input must be either 0, which opens the breaker, or any positive value, which closes the breaker. For clarity, a 1 signal is commonly used to close the breaker.

When the Breaker block is set in internal control mode, the switching times are specified in the dialog box of the block.

When the breaker is closed, it is represented by a resistance R_{on} . To be negligible compared to external components, you can set the R_{on} value as small as necessary. A typical value is 10 milliohms. When the breaker is open, it has an infinite resistance.

The arc extinction process is simulated by opening the breaker device when its current passes through 0 at the first current zero crossing following the transition of the Simulink control input from 1 to 0.

Operation	Conditions
The Breaker closes when	Control signal goes to 1 (for discrete systems, control signal must stay at 1 for at least three times the sampling period)
The Breaker opens when	Control signal goes to 0 Breaker current passes through 0

Note The Breaker block may not be the appropriate switching device to use for DC circuits. For such applications, it is recommended that you use the Ideal Switch block as a switching device.

Assumptions and Limitations

When the block is connected in series with an inductor or another current source, you must add the snubber circuit. In most applications, you can use a resistive snubber (**Snubber capacitance**

parameter set to `inf`) with a large resistor value (**Snubber resistance** parameter set to `1e6` or so). Because of modeling constraints, the internal breaker inductance `Ron` cannot be set to 0.

Use a stiff integration algorithm to simulate circuits with the Breaker block. `ode23tb` with default parameters usually gives the best simulation speed.

For discretized models, the control signal must stay at 1 for a minimum of three sampling time periods to correctly close the Breaker block, otherwise the device stays open.

Ports

Input

c — Breaker operation control
Simulink signal

Signal to control the breaker operation. The control signal connected to this Simulink input must be either 0, which opens the breaker, or any positive value, which closes the breaker. For clarity, a 1 signal is commonly used to close the breaker.

Dependencies

To enable this port, select the **External** parameter.

Conserving

1 — Terminal 1
specialized electrical

Specialized electrical conserving port associated with the terminal 1 of the breaker.

2 — Terminal 2
specialized electrical

Specialized electrical conserving port associated with the terminal 2 of the breaker.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Initial status — Initial status of breaker
0 (default)

Initial state of the breaker. A closed contact is displayed in the block icon when it is set to 1, and an open contact is displayed when it is set to 0.

If the breaker initial state is set to 1 (closed), the software automatically initializes all the states of the linear circuit and the Breaker block initial current, so that the simulation starts in steady state.

Switching times — Vector of switching times
[1/60 5/60] (default)

Vector of switching times when using the Breaker block in internal control mode. At each switching time the Breaker block opens or closes depending on its initial status. For example, if the **Initial**

status parameter is 0 (open), the breaker closes at the first switching time, opens at the second switching time, and so on.

Dependencies

To enable this parameter, clear the **External** parameter.

External — Option for external control
on (default) | off

If selected, adds a Simulink input to the Breaker block for external control of the switching times of the breaker. The switching times are defined by a logical signal (0 or 1) connected to the Simulink input. Default is selected.

Breaker resistance Ron — Internal breaker resistance
0.01 (default)

Internal breaker resistance, in ohms (Ω). The **Breaker resistance Ron** parameter cannot be set to 0.

Snubber resistance Rs — Snubber resistance
1e6 (default)

Snubber resistance, in ohms (Ω). Set this parameter to `inf` to eliminate the snubber from the model.

Snubber capacitance Cs — Snubber capacitance
`inf` (default)

Snubber capacitance, in farads (F). Set this parameter to 0 to eliminate the snubber, or to `inf` to get a resistive snubber.

Measurements — Measurements
None (default) | Branch voltage | Branch current | Branch voltage and current

Select **Branch voltage** to measure the voltage across the Breaker block terminals.

Select **Branch current** to measure the current flowing through the Breaker block. If the snubber device is connected to the breaker model, the measured current is the one flowing through the breaker contacts only.

Select **Branch voltage and current** to measure the breaker voltage and the breaker current.

Place a Multimeter block in your model to display the selected measurements during the simulation.

In the **Available Measurements** list box of the Multimeter block, the measurement is identified by a label followed by the block name:

Measurement	Label
Branch voltage	Ub:
Branch current	Ib:

Version History

Introduced before R2006a

Extended Capabilities

C/C++ Code Generation

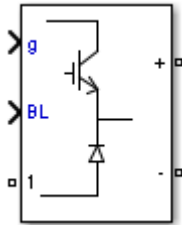
Generate C and C++ code using Simulink® Coder™.

See Also

Three-Phase Breaker | Three-Phase Fault

Buck Converter

Implement buck power converter



Libraries:

Simscape / Electrical / Specialized Power Systems / Power Electronics

Description

The Buck Converter block implements a buck power converter. You can choose from four model types:

- **Switching devices** — The converter is modeled with IGBT/diode pairs controlled by firing pulses produced by a PWM generator. This model provides the most accurate simulation results.
- **Switching function** — The converter is modeled by a switching-function model. The switches are replaced with two voltage sources and two diodes on the AC side and with two current sources on the DC side.

The converter is controlled by firing pulses produced by a PWM generator (0/1 signals) or by firing pulses averaged over a specified period (PWM averaging: signals between 0 and 1). Both modes of operation produce harmonics normally generated by a PWM-controlled converter and also correctly simulate rectifying operation as well as blanking time. This model type is well-suited for real-time simulation.

- **Average model (D-controlled)** — The converter is modeled using a switching-function model directly controlled by the duty cycle signal ($0 < D < 1$). A PWM generator is not required.
- **Average model (D-controlled, no rectifier mode)** — The block uses the voltage source directly controlled by the reference voltage to model the converter. The model does not require a PWM generator and does not simulate the rectifier mode. This setting provides the fastest simulations.

When doing real-time simulation, use the **Switching function** setting with the firing pulses averaged, or the **Average model (D-controlled)** or **Average model (D-controlled, no rectifier mode)** settings.

Ports

Input

BL — Input to block firing pulses
scalar

Input port to block all firing pulses to the converter. To block all firing pulses to the converter, apply a signal value of 1 to this port.

g — Signal to control converter
scalar

The firing pulse signal to control the converter.

Dependencies

To enable this port, set **Model type** to Switching devices or Switching function.

D — Signal to control converter
scalar in the range [0,1]

The duty cycle signal (a value between 0 and 1) used to control the converter.

Dependencies

To enable this port, set **Model type** to Average model (D-controlled).

Conserving

+ — Positive terminal
specialized electrical

Specialized electrical conserving port associated with the converter's positive terminal.

- — Negative terminal
specialized electrical

Specialized electrical conserving port associated with the converter's negative terminal.

1 — Output electrical terminal
specialized electrical

Specialized electrical conserving port associated with the converter's output electrical terminal.

Parameters

Model type — Block model type
Switching devices (default) | Switching function | Average model (D-controlled) |
Average model (D-controlled, no rectifier mode)

Specify the model type for the block to use.

Device on-state resistance (Ohms) — Initial device resistance
1e-3 (default) | positive scalar

Internal resistance of the switching devices, in ohms.

Dependencies

To enable this parameter, set **Model type** to Switching devices.

Snubber resistance (Ohms) — Resistance for snubber
1e6 (default) | positive scalar

The snubber resistance, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to `Switching devices`.

Snubber capacitance (F) — Capacitance for snubber

`inf` (default) | nonnegative scalar

The snubber capacitance, in farads. Set the snubber capacitance to `0` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to `Switching devices`.

Forward voltages [Device Vf (V), Diode Vfd (V)] — Device and diode forward voltage

`[0, 0]` (default) | two-element vector

The forward voltage of the switching device and of the diode, in volts.

Dependencies

To enable this parameter, set **Model type** to `Switching devices`.

Diode on-state resistance (Ohms) — Internal resistance of diodes

`1e-3` (default) | positive scalar

Internal resistance of the diodes, in ohms.

Dependencies

To enable this parameter, set **Model type** to `Switching function` or `Average model (D-controlled)`.

Diode snubber resistance (Ohms) — Snubber resistance of diodes

`1e6` (default) | positive scalar

The snubber resistance, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers. This parameter is available only when you set the **Model type** parameter to `Switching function` or `Average model (D-controlled)`.

Dependencies

To enable this parameter, set **Model type** to `Switching function` or `Average model (D-controlled)`.

Diode snubber capacitance (F) — Snubber capacitance of diodes

`inf` (default) | nonnegative scalar

The snubber capacitance, in farads. Set the snubber capacitance to `0` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to `Switching function` or `Average model (D-controlled)`.

Diode forward voltage (V) — Forward voltage across diode

`1e-3` (default) | nonnegative scalar

Forward voltage, in volts, across the diode when it is conducting.

Dependencies

To enable this parameter, set **Model type** to Switching function or Average model (D-controlled).

Current source snubber resistance (Ohms) — Snubber resistance across two sources

`inf` (default) | positive scalar

The snubber resistance across the two current sources, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to Switching function or Average model (D-controlled).

DC side snubber resistance (Ohms) — Snubber resistance across current sources

`inf` (default) | positive scalar

Snubber resistance across the two current sources, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to Switching function, Average model (D-controlled), or Average model (No rectifier mode).

AC side source resistance (Ohms) — Source resistance on AC side

`1e-3` (default) | positive scalar

Source resistance on the AC side, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to Average model (No rectifier mode).

Version History

Introduced in R2015b

Extended Capabilities

C/C++ Code Generation

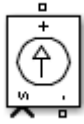
Generate C and C++ code using Simulink® Coder™.

See Also

Full-Bridge Converter | Two-Level Converter

Controlled Current Source

Implement controlled current source

**Libraries:**

Simscape / Electrical / Specialized Power Systems / Sources

Description

The Controlled Current Source block converts the Simulink input signal into an equivalent current source. The generated current is driven by the input signal of the block. The positive current direction is as shown by the arrow in the block icon.

You can initialize the Controlled Current Source block with a specific AC or DC current. If you want to start the simulation in steady state, the block input must be connected to a signal starting as a sinusoidal or DC waveform corresponding to the initial values.

Note The Simulink input signal must be a complex value (phasor) when you are using the Controlled Current Source block in a phasor simulation, because in this case, Simscape Electrical Specialized Power Systems software does not automatically convert the input signal from time-domain into complex (phasor) signal.

Examples

The `power_controlcurr` example uses a Controlled Current Source to generate a 60-Hz current modulated at 5 Hz.

Ports

Input

S — Signal to convert into equivalent current source
Simulink signal

Simulink signal that you want to convert into an equivalent current source.

Conserving

+ — Positive terminal
specialized electrical

Specialized electrical conserving port associated with the positive terminal.

- — Negative terminal
specialized electrical

Specialized electrical conserving port associated with the negative terminal.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Initialize — Whether to initialize Controlled Current Source block
On (default) | Off

Whether to initialize the Controlled Current Source block with the specified **Initial current**, **Initial phase**, and **Initial frequency** parameters.

Source type — Type of current source
AC (default) | DC

Type of current source. Select AC (default) to initialize the Controlled Current Source Block as an AC current source. Select DC to initialize the Controlled Current Source block as a DC current.

Dependencies

To enable this parameter, select the **Initialize** parameter.

Initial amplitude — Initial peak current
0 (default) | nonnegative scalar

Initial peak current for the initialization of the source, in amperes (A).

Dependencies

To enable this parameter, select the **Initialize** parameter.

Initial phase — Initial phase
0 (default) | nonnegative scalar

Initial phase for the initialization of the source, in degrees.

Dependencies

To enable this parameter, select the **Initialize** parameter and set **Source type** to AC.

Initial frequency — Initial frequency
0 (default) | nonnegative scalar

Initial frequency for the initialization of the source, in hertz (Hz).

Dependencies

To enable this parameter, select the **Initialize** parameter and set **Source type** to AC.

Measurements — Option to measure current
None (default) | Current

Select Current to measure the current flowing through the Controlled Current Source block.

Place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurements** list box of the Multimeter block, the measurement is identified by a label followed by the block name:

Measurement	Label
Current	Isrc:

Version History

Introduced before R2006a

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

AC Current Source | Controlled Voltage Source | Multimeter

Controlled Voltage Source

Implement controlled voltage source



Libraries:

Simscape / Electrical / Specialized Power Systems / Sources

Description

The Controlled Voltage Source block converts a Simulink input signal into an equivalent voltage source. The generated voltage is driven by the input signal of the block.

You can initialize the Controlled Voltage Source block with a specific AC or DC voltage. If you want to start the simulation in steady state, the Simulink input must be connected to a signal that begins the simulation as a sinusoidal or DC waveform that corresponds to the initial values.

Note The Simulink input signal must be a complex value (phasor) when you use the Controlled Voltage Source block in a phasor simulation, because the Simscape Electrical Specialized Power Systems software does not automatically convert input signals from time-domain signals to complex (phasor) signals.

Examples

Generate 60 Hz Sinusoidal Voltage with Third Harmonic

The `power_controlvolt` example uses Controlled Voltage Source blocks to generate a 60 Hz sinusoidal voltage containing a third harmonic. One Controlled Voltage Source block is initialized as a 120 V AC voltage source with an initial frequency of 60 Hz and initial phase set to 0. The second Controlled Voltage Source block is not initialized.

Ports

Input

s — Control signal
physical signal

Simulink physical signal input port associated with the control signal.

Conserving

+ — Positive terminal
specialized electrical

Specialized electrical conserving port associated with the positive terminal of the voltage source.

- — Negative terminal
specialized electrical

Specialized electrical conserving port associated with the negative terminal of the voltage source.

Parameters

Initialize — Enable source initialization
on (default) | off

Initializes the Controlled Voltage Source block with the specified values of the **Initial amplitude (V)**, **Initial phase (deg)**, and **Initial frequency (Hz)** parameters.

Source type — AC or DC voltage source
AC (default) | DC

Type of voltage source. Select AC (default) to initialize the Controlled Voltage Source block with an AC voltage source. Select DC to initialize the Controlled Voltage Source block with a DC voltage.

Dependencies

To enable this parameter, select **Initialize**.

Initial amplitude (V) — Amplitude at simulation start
0 (default) | scalar

Initial amplitude of the source, in volts (V).

Dependencies

To enable this parameter, select **Initialize**.

Initial phase (deg) — Phase at simulation start
0 (default) | scalar

Initial phase of the source, in degrees.

Dependencies

To enable this parameter, select **Initialize** and set **Source type** to AC.

Initial frequency (Hz) — Frequency at simulation start
0 (default) | nonnegative scalar

Initial frequency of the source, in hertz (Hz).

Dependencies

To enable this parameter, select **Initialize** and set **Source type** to AC.

Measurements — Source measurements
None (default) | Voltage

Select **Voltage** to measure the voltage across the terminals of the Controlled Voltage Source block.

To measure and voltage, place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurements** pane of the Multimeter block, the measurement is identified by a label followed by the block name:

Measurement	Label
Voltage	Usrc:

Version History

Introduced before R2006a

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

AC Current Source | Controlled Current Source | Multimeter

Current Measurement

Measure current in circuit

**Libraries:**

Simscape / Electrical / Specialized Power Systems / Sensors and Measurements

Description

The Current Measurement block measures the instantaneous current flowing in any electrical block or connection line. The Simulink signal output provides a signal that other Simulink blocks can use.

The `power_currmeasure` example uses four Current Measurement blocks to read currents in different branches of a circuit. To open this example, at the MATLAB Command Window, enter `power_currmeasure`.

Ports

Output

i — Output
scalar

Simulink signal associated with the instantaneous current flowing in any block or connection line.

Conserving

+ — Positive terminal
specialized electrical

Specialized electrical conserving port associated with the positive terminal of the Current Measurement block.

- — Negative terminal
specialized electrical

Specialized electrical conserving port associated with the negative terminal of the Current Measurement block.

Parameters

Output signal — Output signal
Complex (default) | Real-Imag | Magnitude-Angle | Magnitude

Format of the output signal to use when you use the block in a phasor simulation.

Set this parameter to **Complex** to output the measured current as a complex value. The output is a complex signal.

Set this parameter to **Real - Imag** to output the real and imaginary parts of the measured current. The output is a vector of two elements.

Set this parameter to **Magnitude - Angle** to output the magnitude and angle of the measured current. The output is a vector of two elements.

Set this parameter to **Magnitude** to output the magnitude of the measured current. The output is a scalar value.

Dependencies

To enable this parameter, in the powergui block, set the **Simulation type** parameter to **Phasor** or **Discrete Phasor**.

Version History

Introduced before R2006a

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

powergui | Three-Phase V-I Measurement | Voltage Measurement

Current-Voltage Simscape Interface

Ideal coupling between Simscape Electrical Specialized Power Systems and Simscape electrical circuits



Libraries:

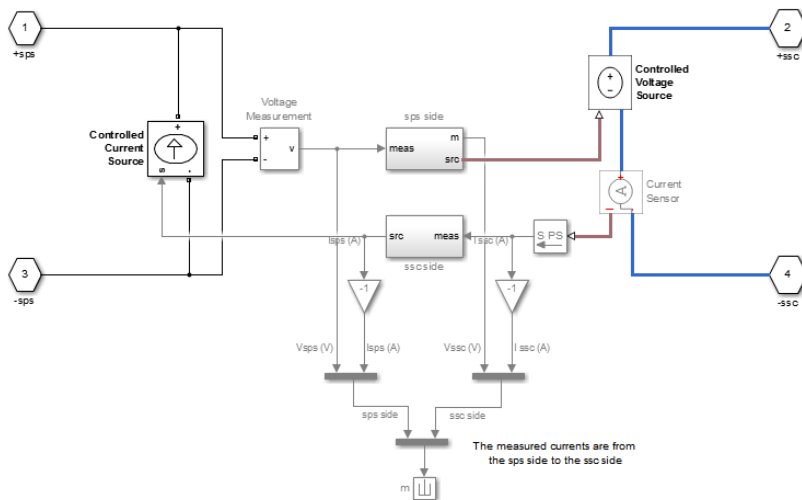
Simscape / Electrical / Specialized Power Systems / Utilities

Description

The Current-Voltage Simscape Interface block connects Simscape Electrical Specialized Power Systems circuits with electrical elements from the Simscape Foundation library. This block preserves current through the block and voltage across the block, conserving electrical power.

- On the Simscape Electrical Specialized Power Systems side, the electrical connector ports **+sps** and **-sps** act like a controlled current source, feeding current from the Simscape side to the connected Simscape Electrical Specialized Power Systems elements, while maintaining the voltage across the block.
- On the Simscape side, the electrical connector ports **+ssc** and **-ssc** act like a controlled voltage source, feeding voltage from the Simscape Electrical Specialized Power Systems side to the connected Simscape elements, while maintaining the current through the block.

The following figure gives an overview of the interface.



The block measures the voltage between the **+sps** and **-sps** ports and propagates it to the **+ssc** and **-ssc** ports, preserving the polarity. The block also measures the current flowing from the **+ssc** to the **-ssc** port and propagates it from the **-sps** to the **+sps** port.

You can specify a filter to break the algebraic loop between Simscape Electrical Specialized Power Systems and Simscape ports. You can specify the filter as a first order continuous transfer function or as a discrete unit delay block. You can apply it on the Simscape Electrical Specialized Power Systems side, to the measured voltage, or on the Simscape side, to the measured current. Depending on the measured signal where the filter is applied, the other signal is always a direct feedthrough.

If you do not specify a filter, direct feedthrough is applied on both measured signals. However, this configuration results in an algebraic loop containing the Voltage Measurement, Controlled Voltage Source, Current Sensor, and Controlled Current Source blocks, which might cause numerical issues.

Limitations

When simulating Simscape Electrical Specialized Power Systems models along with Simscape blocks, select the Simulink solver carefully.

Use `ode23tb` solver for solving Simscape Electrical Specialized Power Systems circuits containing switches or nonlinear element. If you use other stiff solvers, Simscape Electrical Specialized Power Systems software gives you the following warning:

```
Warning: You have required continuous-time simulation of a system containing
switches or nonlinear elements. The ode23tb variable-step stiff solver with
relative tolerance set to 1e-4 is recommended in order to get best accuracy
and simulation performance. For some highly nonlinear models it may be
necessary to set the "Solver reset method" parameter to "Robust".
See "Improving Simulation Performance" chapter in Simscape Electrical
Specialized Power Systems documentation for additional information on how to
select an appropriate integration method.
'To ignore Simscape Electrical Specialized Power Systems warnings, select
"Disable Simscape Electrical SPS warnings" in the Powergui Preferences tab.';
```

The Simscape Electrical Specialized Power Systems warning can be disabled, as described in the warning message. However, if you use the `ode23tb` solver for solving models containing Simscape blocks, Simscape software gives you the following warning:

```
Warning: The solver chosen is not recommended if the model is stiff, which is
typical for models containing Simscape components. These solvers are typically
preferred for Simscape models:ode23t, ode15s or ode14x (fixed-step). To disable
this diagnostic,change the explicit solver diagnostic setting on the Simscape
panel of the model Configuration Parameters.
```

The Simscape warning can also be disabled, as described in the warning message.

Ports

Conserving

+sps — Positive terminal of Specialized Power Systems circuit
specialized electrical

Specialized electrical conserving port associated with the positive terminal of the Specialized Power Systems circuit.

-sps — Negative terminal of Specialized Power Systems circuit
specialized electrical

Specialized electrical conserving port associated with the negative terminal of the Specialized Power Systems circuit.

+ssc — Positive terminal of Simscape circuit
specialized electrical

Specialized electrical conserving port associated with the positive terminal of the Simscape circuit.

-ssc — Negative terminal of Simscape circuit
specialized electrical

Specialized electrical conserving port associated with the negative terminal of the Simscape circuit.

Output

m — Measurement port
Simulink signal

Bus signal that contains the Simscape Electrical Specialized Power Systems current and voltage and the Simscape current and voltages.

Dependencies

To enable this port, select the **Show measurement ports** parameter.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Direct feedthrough (no filter) — Direct feedthrough
off (default) | on

When you select this parameter, the block applies direct feedthrough between Simscape Electrical Specialized Power Systems and Simscape ports on both sides.

Filter type — Type of filter

Low pass filter (continuous) (default) | Unit delay (discrete) | First-order input filtering | Second-order input filtering

Set this parameter to **Low pass filter (continuous) (default)** to implement a first order continuous transfer function filter between Simscape Electrical Specialized Power Systems and Simscape ports.

Set this parameter to **Unit delay (discrete)** to implement a discrete unit delay filter between Simscape Electrical Specialized Power Systems and Simscape ports.

Set this parameter to **First-order input filtering** to implement a first order continuous transfer function filter with first-order filtering of the Simulink-PS Converter block that provides one derivative. You can apply this filter only on the Simscape Electrical Specialized Power Systems side.

Set this parameter to **Second-order input filtering** to implement a first order continuous transfer function filter with second-order filtering of the Simulink-PS Converter block that provides derivatives one and two. You can apply this filter only on the Simscape Electrical Specialized Power Systems side.

Dependencies

To enable this parameter, clear the **Direct feedthrough (no filter)** parameter.

Low pass filter time constant (s) — Time constant of low pass filter
1e-6 (default) | positive scalar

Time constant of low pass filter, in seconds.

Dependencies

To enable this parameter, set **Filter type** to Low pass filter (continuous), First-order input filtering, or Second-order input filtering.

Sample time (s) — Sample time of unit delay filter
50e-6 (default) | positive scalar

Sample time of the unit delay filter, in seconds.

Dependencies

To enable this parameter, set **Filter type** to Unit delay (discrete).

Initial voltage (V) — Initial voltage
0 (default) | nonnegative scalar

Initial voltage value of the measured Simscape Electrical Specialized Power Systems voltage.

Dependencies

To enable this parameter, set **Apply filter on** to Specialized Power Systems side.

Initial current (A) — Initial current
0 (default) | nonnegative scalar

Initial current value of the measured Simscape Electrical current.

Dependencies

To enable this parameter, set **Apply filter on** to Simscape side.

Apply filter on — Side to apply the filter on
inf (default) | nonnegative scalar

Set this parameter to Specialized Power Systems side to apply the filter on the measured Simscape Electrical Specialized Power Systems voltage.

Set this parameter to Simscape side to apply the filter on the measured Simscape current.

Dependencies

To enable this parameter, clear the **Direct feedthrough (no filter)** parameter and set **Filter type** to Low pass filter (continuous) or Unit delay (discrete).

Show measurement port — Option to show measurement port
off (default) | on

If selected, the block exposes a Simulink output that returns the Simscape Electrical Specialized Power Systems current and voltage and to the Simscape current and voltages in a bus signal.

Version History

Introduced in R2011b

Extended Capabilities

C/C++ Code Generation

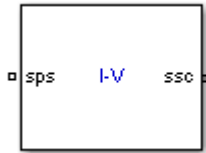
Generate C and C++ code using Simulink® Coder™.

See Also

[Current-Voltage Simscape Interface \(gnd\)](#) | [Voltage-Current Simscape Interface](#) | [Voltage-Current Simscape Interface \(gnd\)](#)

Current-Voltage Simscape Interface (gnd)

Ideal coupling between Simscape Electrical Specialized Power Systems and Simscape electrical circuits



Libraries:

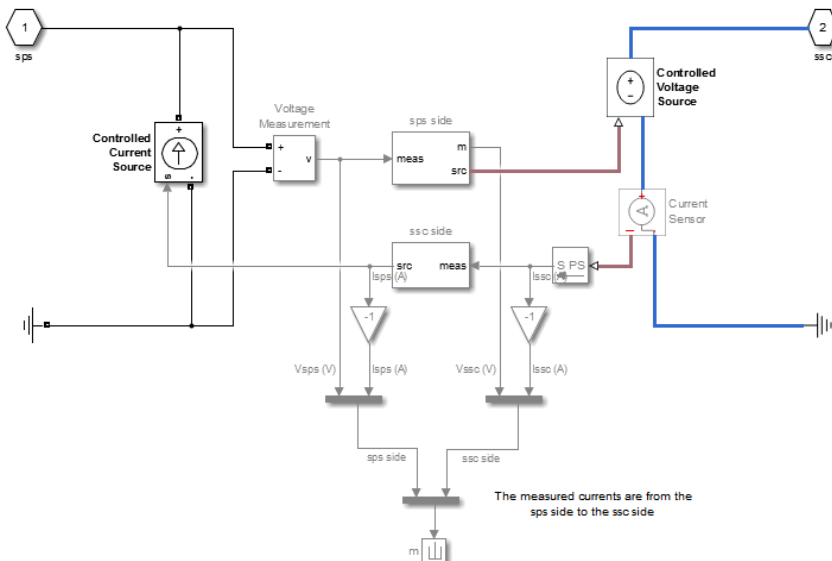
Simscape / Electrical / Specialized Power Systems / Utilities

Description

The Current-Voltage Simscape Interface (gnd) block connects Simscape Electrical Specialized Power Systems circuits with electrical elements from the Simscape Foundation library. This block preserves current through the block and voltage across the block, conserving electrical power.

- On the Simscape Electrical Specialized Power Systems side, the electrical connector port **sps** acts like a controlled current source referred to the ground, feeding current from the Simscape side to the connected Simscape Electrical Specialized Power Systems elements.
- On the Simscape side, the electrical connector port **ssc** acts like a controlled voltage source referred to the ground, feeding voltage from the Simscape Electrical Specialized Power Systems side to the connected Simscape elements.

The following figure gives an overview of the interface.



The block measures the voltage between the **sps** port and Simscape Electrical Specialized Power Systems electrical reference and propagates it to the **ssc** port, preserving the polarity with respect to the Simscape electrical reference. The block also measures the current flowing from the **ssc** port to the Simscape electrical reference and propagates it to the **sps** port.

You can specify a filter to break the algebraic loop between Simscape Electrical Specialized Power Systems and Simscape ports. You can specify the filter as a first order continuous transfer function or as a discrete unit delay block. You can apply the filter on the Simscape Electrical Specialized Power Systems side, to the measured voltage, or on the Simscape side, to the measured current. Depending on the measured signal where you apply the filter, the other signal is always a direct feedthrough.

If you do not specify a filter, direct feedthrough is applied on both measured signals. However, this configuration results in an algebraic loop containing the Voltage Measurement, Controlled Voltage Source, Current Sensor, and Controlled Current Source blocks, which might cause numerical issues.

Limitations

When simulating Simscape Electrical Specialized Power Systems models along with Simscape blocks, select the Simulink solver carefully.

Use the `ode23tb` solver for solving Simscape Electrical Specialized Power Systems circuits containing switches or nonlinear element. If you use other stiff solvers, Simscape Electrical Specialized Power Systems software gives you the following warning:

```
Warning: You have required continuous-time simulation of a system containing
switches or nonlinear elements. The ode23tb variable-step stiff solver with
relative tolerance set to 1e-4 is recommended in order to get best accuracy
and simulation performance. For some highly nonlinear models it may be
necessary to set the "Solver reset method" parameter to "Robust".
See "Improving Simulation Performance" chapter in Simscape Electrical
Specialized Power Systems documentation for additional information on how to
select an appropriate integration method.
'To ignore Simscape Electrical Specialized Power Systems warnings, select
"Disable Simscape Electrical SPS warnings" in the Powergui Preferences tab.';
```

The Simscape Electrical Specialized Power Systems warning can be disabled, as described in the warning message. However, if you use the `ode23tb` solver for solving models containing Simscape blocks, Simscape software gives you the following warning:

```
Warning: The solver chosen is not recommended if the model is stiff, which is
typical for models containing Simscape components. These solvers are typically
preferred for Simscape models:ode23t, ode15s or ode14x (fixed-step). To disable
this diagnostic, change the explicit solver diagnostic setting on the Simscape
panel of the model Configuration Parameters.
```

The Simscape warning can also be disabled, as described in the warning message.

Ports

Conserving

sps — Terminal of Specialized Power Systems circuit
`specialized electrical`

Specialized electrical conserving port associated with the terminal of the Specialized Power Systems circuit.

ssc — Terminal of Simscape circuit
`specialized electrical`

Specialized electrical conserving port associated with the terminal of the Simscape circuit.

Output

m — Measurement port
Simulink signal

Bus signal that contains the Simscape Electrical Specialized Power Systems current and voltage and the Simscape current and voltages.

Dependencies

To enable this port, select the **Show measurement ports** parameter.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Direct feedthrough (no filter) — Direct feedthrough
off (default) | on

When you select this parameter, the block applies direct feedthrough between Simscape Electrical Specialized Power Systems and Simscape ports on both sides.

Filter type — Type of filter
Low pass filter (continuous) (default) | Unit delay (discrete) | First-order input filtering | Second-order input filtering

Set this parameter to **Low pass filter (continuous) (default)** to implement a first order continuous transfer function filter between Simscape Electrical Specialized Power Systems and Simscape ports.

Set this parameter to **Unit delay (discrete)** to implement a discrete unit delay filter between Simscape Electrical Specialized Power Systems and Simscape ports.

Set this parameter to **First-order input filtering** to implement a first order continuous transfer function filter with first-order filtering of the Simulink-PS Converter block that provides one derivative. You can apply this filter only on the Simscape Electrical Specialized Power Systems side.

Set this parameter to **Second-order input filtering** to implement a first order continuous transfer function filter with second-order filtering of the Simulink-PS Converter block that provides derivatives one and two. You can apply this filter only on the Simscape Electrical Specialized Power Systems side.

Dependencies

To enable this parameter, clear the **Direct feedthrough (no filter)** parameter.

Low pass filter time constant (s) — Time constant of low pass filter
1e-6 (default) | positive scalar

Time constant of low pass filter, in seconds.

Dependencies

To enable this parameter, set **Filter type** to **Low pass filter (continuous)**, **First-order input filtering**, or **Second-order input filtering**.

Sample time (s) — Sample time of unit delay filter

50e-6 (default) | positive scalar

Sample time of the unit delay filter, in seconds.

Dependencies

To enable this parameter, set **Filter type** to Unit delay (discrete).

Initial voltage (V) — Initial voltage

0 (default) | nonnegative scalar

Initial voltage value of the measured Simscape Electrical Specialized Power Systems voltage.

Dependencies

To enable this parameter, set **Apply filter on** to Specialized Power Systems side.

Initial current (A) — Initial current

0 (default) | nonnegative scalar

Initial current value of the measured Simscape Electrical current.

Dependencies

To enable this parameter, set **Apply filter on** to Simscape side.

Apply filter on — Side to apply the filter on

inf (default) | nonnegative scalar

Set this parameter to Specialized Power Systems side to apply the filter on the measured Simscape Electrical Specialized Power Systems voltage.

Set this parameter to Simscape side to apply the filter on the measured Simscape current.

Dependencies

To enable this parameter, clear the **Direct feedthrough (no filter)** parameter and set **Filter type** to Low pass filter (continuous) or Unit delay (discrete).

Show measurement port — Option to show measurement port

off (default) | on

If selected, the block exposes a Simulink output that returns the Simscape Electrical Specialized Power Systems current and voltage and to the Simscape current and voltages in a bus signal.

Version History

Introduced in R2011b

Extended Capabilities

C/C++ Code Generation

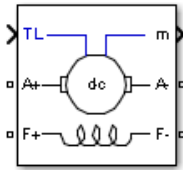
Generate C and C++ code using Simulink® Coder™.

See Also

[Current-Voltage Simscape Interface](#) | [Voltage-Current Simscape Interface](#) | [Voltage-Current Simscape Interface \(gnd\)](#)

DC Machine

Implement wound-field or permanent magnet DC machine



Libraries:

Simscape / Electrical / Specialized Power Systems / Electrical Machines

Description

The DC Machine block implements a wound-field or permanent magnet DC machine.

For the wound-field DC machine, you can access the field terminals (F+, F–) to use the machine model as a shunt-connected or a series-connected DC machine. The Simulink input T_L provides the torque applied to the shaft.

The armature circuit (A+, A–) consists of an inductor L_a and resistor R_a in series with a counter-electromotive force (CEMF) E .

The CEMF is proportional to the machine speed.

$$E = K_E \omega \quad (1-1)$$

K_E is the voltage constant and ω is the machine speed.

In a separately excited DC machine model, the voltage constant K_E is proportional to the field current I_f :

$$K_E = L_{af} I_f, \quad (1-2)$$

L_{af} is the field-armature mutual inductance.

The electromechanical torque that the DC Machine block develops is proportional to the armature current I_a .

$$T_e = K_T I_a, \quad (1-3)$$

K_T is the torque constant. The sign convention for T_e and T_L is:

$$\begin{aligned} T_e, T_L > 0: & \text{Motor mode} \\ T_e, T_L < 0: & \text{Generator mode} \end{aligned} \quad (1-4)$$

The torque constant is equal to the voltage constant.

$$K_T = K_E. \quad (1-5)$$

The armature circuit is connected between the A+ and A– ports of the DC Machine block. It is represented by a series R_a L_a branch in series with a Controlled Voltage Source and a Current Measurement block.

In the wound-field DC machine model, the field circuit is represented by an RL circuit. It is connected between the F+ and F– ports of the DC Machine block.

In the permanent magnet DC machine model, there is no field current as the magnets establish the excitation flux. K_E and K_T are constants.

The mechanical part computes the speed of the DC machine from the net torque applied to the rotor. The block uses the speed to implement the CEMF voltage E of the armature circuit.

The mechanical part implements this equation:

$$J \frac{d\omega}{dt} = T_e - T_L - B_m \omega - T_f,$$

where J is the inertia, B_m is the viscous friction coefficient, and T_f is the Coulomb friction torque.

Ports

Input

TL — Mechanical load torque, N*m
scalar

Mechanical load torque, in N*m.

Dependencies

To enable this port, set **Mechanical input** to Torque TL.

w — Machine speed, rad/s
scalar

Speed of the machine, in rad/s.

Dependencies

To enable this port, set **Mechanical input** to Speed w.

Output

m — Measurement signals
four elements vector

Vector of measurement signals. To demultiplex these signals, use the Bus Selector block in the Simulink library.

Name	Definition	Units
w	Speed ω_m	rad/s
iA	Armature current i_a	A
iF	Field current i_f	A
Te	Electrical torque T_e	N.m

Conserving

A+ — Positive terminal of armature
specialized electrical

Specialized electrical conserving port associated with the positive terminal of the armature circuit.

A- — Negative terminal of armature
specialized electrical

Specialized electrical conserving port associated with the negative terminal of the armature circuit.

F+ — Positive terminal of field circuit
specialized electrical

Specialized electrical conserving port associated with the positive terminal of the field circuit.

Dependencies

To enable this port, set **Field type** to Wound.

F- — Negative terminal of field circuit
specialized electrical

Specialized electrical conserving port associated with the negative terminal of the field circuit.

Dependencies

To enable this port, set **Field type** to Wound.

S — Machine shaft
mechanical rotational

Mechanical rotational conserving port associated with the machine shaft.

Dependencies

To enable this port, set **Mechanical input** to Mechanical rotational port.

Parameters

Configuration

Preset model — Set of predetermined electrical and mechanical parameters for DC machine
No (default) | 01: 5HP 240V 1750RPM Field:300V | 02: 5HP 240V 1750RPM Field:150V |
03: 5HP 500V 1750RPM Field:300V | 04: 10HP 240V 1750RPM Field:300V | 05: 10HP
500V 1750RPM Field:300V | 06: 20HP 240V 1750RPM Field:300V | 07: 20HP 500V
1750RPM Field:300V | 08: 30HP 240V 1750RPM Field:300V | 09: 30HP 500V 1750RPM
Field:300V | 10: 40HP 240V 1750RPM Field:300V | 11: 40HP 500V 1750RPM Field:300V
| 12: 50HP 240V 1750RPM Field:300V | 13: 50HP 500V 1750RPM Field:300V | 14: 75HP
500V 1750RPM Field:300V | 15: 100HP 500V 1750RPM Field:300V | 16: 125HP 500V
1750RPM Field:300V | 17: 150HP 500V 1750RPM Field:300V | 18: 175HP 500V 1750RPM
Field:300V | 19: 200HP 440V 1184RPM Field:310V | 20: 200HP 500V 1750RPM
Field:150V | 21: 200HP 500V 1750RPM Field:300V | 22: 225HP 500V 1750RPM
Field:300V | 23: 250HP 500V 1750RPM Field:300V

Option to choose a set of predetermined electrical and mechanical parameters for various DC machine ratings of power (HP), DC voltage (V), rated speed (rpm), and field voltage (V).

Select one of the preset models to load the corresponding electrical and mechanical parameters in the entries of the dialog box. Select **No** if you do not want to use a preset model, or if you want to modify some of the parameters of a preset model.

When you select a preset model, the electrical and mechanical parameters in the **Parameters** section of the dialog box become unmodifiable (unavailable). To start from a given preset model and then modify machine parameters:

- 1 Select the preset model that you want to initialize the parameters.
- 2 Change the **Preset model** parameter value to **No**. This does not change the machine parameters. By doing so, you break the connection with the particular preset model.
- 3 Modify the machine parameters as you want, then click **Apply**.

Dependencies

To enable this parameter, set **Field type** to Wound.

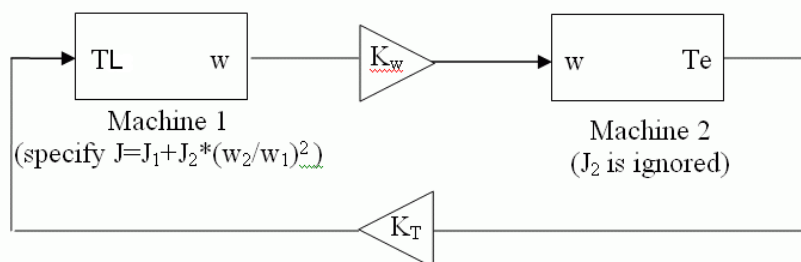
Mechanical input — Mechanical input

Torque TL (default) | Speed w | Mechanical rotational port

Torque applied to the shaft or the rotor speed as a Simulink input of the block, or represent the machine shaft by a Simscape rotational mechanical port:

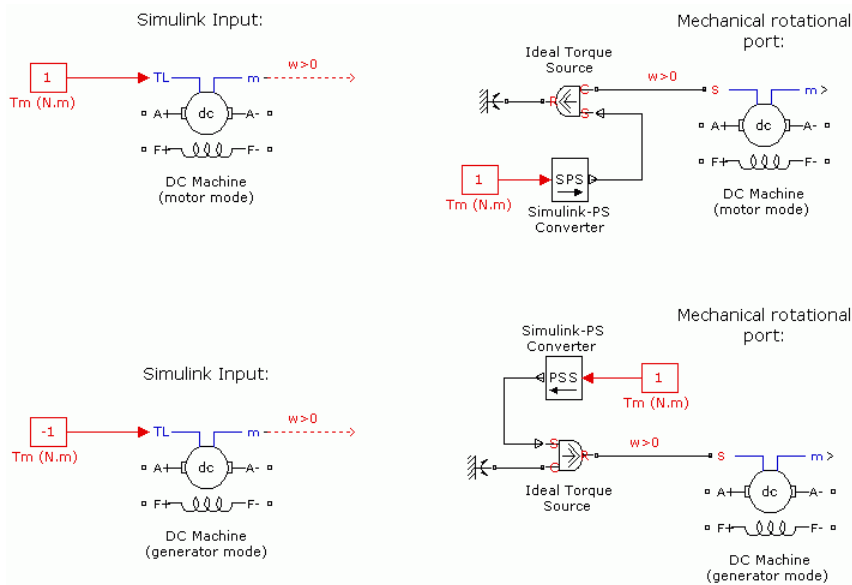
- **Torque TL** — Specify a torque input, in N*m. The block determines the machine speed by using the machine Inertia J and the difference between the applied mechanical load torque TL and the internal electromagnetic torque Te. The sign convention for the mechanical torque is: when the speed is positive, a positive torque signal indicates motor mode and a negative signal indicates generator mode.
- **Speed w** — Specify a speed input, in rad/s. The block imposes the machine speed and ignores the mechanical part of the model (inertia J). Use the speed as the mechanical input to model a mechanical coupling between two machines.

The next figure indicates how to model a stiff shaft interconnection in a motor-generator. The speed output of machine 1 (motor) is connected to the speed input of machine 2 (generator), while machine 2 electromagnetic torque output Te is applied to the mechanical load torque input TL of machine 1. The Kw factor takes into account speed units of both machines (rad/s) and gear box ratio. The KT factor takes into account torque units of both machines (N.m) and machine ratings. Also, as the inertia J2 is ignored in machine 2, J2 referred to machine 1 speed must be added to machine 1 inertia J1.



- **Mechanical rotational port** — Add to the block a Simscape mechanical rotational port that allows connection of the machine shaft with other Simscape blocks having mechanical rotational ports.

The next figure indicates how to connect an Ideal Torque Source block from the Simscape library to the machine shaft to represent the machine in motor mode, or in generator mode, when the rotor speed is positive.



Field type — Field type of machine

Wound (default) | Permanent magnet

Whether to model a wound-field machine or a permanent magnet DC machine.

Measurement output

Use signal names to identify bus labels — Whether to use signal names in measurement output
off (default) | on

Whether to use the signal names in the measurement output to identify the bus labels. Select this parameter for applications that require bus signal label to have only alphanumeric characters.

When you clear this parameter, the measurement output uses the signal definition to identify the bus labels. The labels contain non alphanumeric characters that are incompatible with some Simulink applications.

Parameters

Armature resistance and inductance [Ra (ohms) La (H)] — Armature resistance and inductance
[0.6 0.012] (default)

Armature resistance Ra, in ohms, and armature inductance La, in henries.

Field resistance and inductance [Rf (ohms) Lf (H)] — Field resistance and inductance
[240 120] (default)

Field resistance R_f , in ohms, and field inductance L_f , in henries.

Dependencies

To enable this parameter, in the **Configuration** section, set **Field type** to Wound.

Field-armature mutual inductance L_{af} (H) — Field-armature mutual inductance
1.8 (default)

Field-armature mutual inductance, in henries.

Dependencies

To enable this parameter, in the **Configuration** section, set **Field type** to Wound.

Specify — Machine constant for block parameterization

Torque constant (N.m/A) (default) | Back-emf constant (V/rpm)

For a permanent magnet DC machine, select the machine constant that you want to specify for block parameterization.

Dependencies

To enable this parameter, in the **Configuration** section, set **Field type** to Permanent magnet.

Back-emf constant (V/rpm) — Voltage per speed constant
1.8 (default)

Voltage per speed constant of the permanent magnet DC machine, in V/rpm.

Dependencies

To enable this parameter, in the **Configuration** section, set **Field type** to Permanent magnet and, in the **Parameters** section, set **Specify** to Back-emf constant.

Torque constant (N.m/A) — Torque per current constant
1.8 (default)

Torque per current constant of the permanent magnet DC machine, in N.m/A.

Dependencies

To enable this parameter, in the **Configuration** section, set **Field type** to Permanent magnet and, in the **Parameters** section, set **Specify** to Torque constant.

Total inertia J (kg.m²) — Total inertia of DC machine
1 (default)

Total inertia of the DC machine, in kg.m².

Viscous friction coefficient B_m (N.m.s) — Total friction coefficient of DC machine
0 (default)

Total friction coefficient of the DC machine, in N.m.s.

Coulomb friction torque T_f (N.m) — Coulomb friction torque constant of DC machine
0 (default)

Total Coulomb friction torque constant of the DC machine, in N.m.

Initial speed (rad/s) — Initial speed for DC machine

1 (default)

Initial speed for the DC machine, in rad/s, in order to start the simulation with a specific initial speed. To start the simulation in steady state, the initial value of the input torque signal T_L must be proportional to the initial speed.

Advanced

Sample time (-1 for inherited) — Sample time of block

-1 (default)

Sample time that the block uses. To inherit the sample time specified in the Powergui block, set this parameter to -1.

Version History

Introduced before R2006a

References

[1] *Analysis of Electric Machinery*, Krause et al., pp. 89–92.

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

Asynchronous Machine | Synchronous Machine SI Fundamental | Synchronous Machine pu Fundamental | Synchronous Machine pu Standard

DC Voltage Source

Implement DC voltage source



Libraries:

Simscape / Electrical / Additional Components / SPICE Sources

Description

The DC Voltage Source block implements an ideal DC voltage source. The positive terminal is represented by a plus sign on one port. You can modify the voltage at any time during the simulation.

Examples

The `power_dcVoltage` example illustrates the simulation of the transient response of a first-order RC circuit.

Ports

Conserving

+ — DC voltage source positive terminal
specialized electrical

Specialized electrical conserving port associated with the DC voltage source positive terminal.

- — DC voltage source negative terminal
specialized electrical

Specialized electrical conserving port associated with the DC voltage source negative terminal.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Amplitude — Source amplitude
100 (default) | nonnegative scalar

Amplitude of the source, in volts (V).

Measurements — Measurements option
None (default) | Voltage

Select **Voltage** to measure the voltage across the terminals of the DC Voltage Source block.

Place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurements** list box of the Multimeter block, the measurement is identified by a label followed by the block name:

Measurement	Label
Voltage	Usrc :

Version History

Introduced before R2006a

Extended Capabilities

C/C++ Code Generation

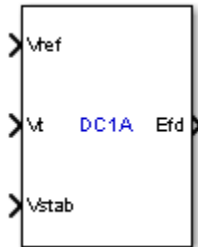
Generate C and C++ code using Simulink® Coder™.

See Also

AC Voltage Source | Controlled Voltage Source

DC1A Excitation System

Implements IEEE type DC1A excitation system model



Library

Simscape / Electrical / Specialized Power Systems / Electrical Machines / Synchronous Machine Control

Description

This block models a field controlled dc-commutator exciter with continuously acting voltage regulator. It uses self-excited shunt fields with a voltage regulator operating in a buck-boost mode. You can use this model to represent other types of excitation systems when detailed data is not available or when a simplified model is required [1].

This block is an adaptation of the DC1A excitation system of the IEEE 421 standard, copyright IEEE 2005, all rights reserved.

Parameters

- “Controllers Tab” on page 1-105
- “Exciter and Rectifier Tab” on page 1-106
- “Initial Values Tab” on page 1-106

Controllers Tab

Low-pass filter time constant

The time constant T_r of the first-order system representing the stator terminal voltage transducer. Default is $20e-3$.

Voltage regulator gain and time constant

The gain K_a and time constant T_a of the first-order system representing the main regulator. Default is $[46 \ 0.06]$.

Voltage regulator output limits

The voltage regulator output limits V_{Rmin} and V_{Rmax} , in p.u. Default is $[-0.9 \ 1.0]$.

Damping filter gain and time constant

The gain K_f and time constant T_f of the first-order system representing the derivative feedback. Default is [0.05 1.5].

Transient gain reduction lead and lag time constants

The time constants T_b and T_c of the first-order system representing the lead-lag compensator. Default is [0 0].

Exciter and Rectifier Tab**Exciter gain and time constant**

The gain K_e and time constant T_e of the first-order system representing the exciter. Default is [0.05 0.46].

Field voltage values

The exciter saturation function is defined as a multiplier of exciter alternator output voltage to represent the increase in exciter excitation requirements due to saturation [1]. The saturation function is determined by specifying two voltage points, E_{fd1} and E_{fd2} in p.u., on the air-gap line and constant resistance load saturation curve, and providing the corresponding two saturation multipliers Se_{Efd1} and Se_{Efd2} . Default is [3.1 2.3].

Typically, the voltage E_{fd1} is a value near the expected exciter maximum output voltage. The E_{fd2} value is about 75% of E_{fd1} .

Exciter saturation function values

The exciter saturation function is defined as a multiplier of exciter alternator output voltage to represent the increase in exciter excitation requirements due to saturation. The saturation function is determined by specifying two voltage points, E_{fd1} and E_{fd2} in p.u., on the air-gap line and Constant Resistance Load saturation curve and providing the corresponding two saturation multipliers Se_{Efd1} and Se_{Efd2} . Default is [0.33 0.10].

Se_{Efd1} and Se_{Efd2} multipliers are equal to $A-B / B$, A is the value of exciter field current on the Constant Resistance Load saturation curve corresponding to the selected E_{fd} voltage, and B the value of exciter field current on the air-gap line corresponding to the selected E_{fd} voltage [1].

If you do not want to model the saturation effect, set Se_{Efd1} and Se_{Efd2} values to zero.

Initial Values Tab**Initial values of terminal voltage and field voltage**

The initial values of terminal voltage V_{t0} and field voltage E_{fd0} , both in p.u. Initial terminal voltage is normally set to 1 pu. The V_{t0} and E_{fd0} values can be determined using the Powergui Load Flow tool. Default is [1 1].

Sample time

Specify a value greater than zero to discretize the block at the given sample time. Set to -1 to inherit the simulation type and sample time parameters of the Powergui block. Default is 0.

Ports

V_{ref}

The reference value of the stator terminal voltage, in p.u.

Vt

The measured value in p.u. of the stator terminal voltage of the controlled Synchronous Machine block.

Vstab

Connect this input to a power system stabilizer to provide additional stabilization of power system oscillations. When you do not use this option, connect to a Simulink ground block. The input is in p.u.

Efd

The field voltage to apply to the Vf input of the controlled Synchronous Machine block. The output is in p.u.

References

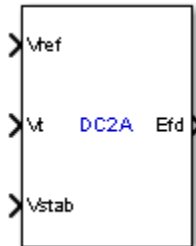
- [1] "IEEE Recommended Practice for Excitation System Models for Power System Stability Studies."
IEEE Standard, Vol. 421, No. 5, 2005 (Revision of *IEEE 521.5-1992*).

Version History

Introduced before R2006a

DC2A Excitation System

Implements IEEE type DC2A excitation system model



Library

Simscape / Electrical / Specialized Power Systems / Electrical Machines / Synchronous Machine Control

Description

This block models a field controlled dc-commutator exciter with continuously acting voltage regulators supplied from the generator or auxiliaries bus voltage. The voltage regulator output limits are proportional to terminal voltage V_t .

This block is an adaptation of the DC2A excitation system of the IEEE 421 standard, copyright IEEE 2005, all rights reserved.

Parameters

- “Controllers Tab” on page 1-108
- “Exciter and Rectifier Tab” on page 1-109
- “Initial Values Tab” on page 1-109

Controllers Tab

Low-pass filter time constant

The time constant T_r of the first-order system representing the stator terminal voltage transducer. Default is $20e-3$.

Voltage regulator gain and time constant

The gain K_a and time constant T_a of the first-order system representing the main regulator. Default is $[46 \ 0.06]$.

Voltage regulator output limits

The voltage regulator output limits V_{Rmin} and V_{Rmax} , in p.u. Default is $[-4.9 \ 4.95]$.

Damping filter gain and time constant

The gain K_f and time constant T_f of the first-order system representing the derivative feedback. Default is $[0.05 \ 1.5]$.

Transient gain reduction lead and lag time constants

The time constants T_b and T_c of the first-order system representing the lead-lag compensator. Default is [0 0].

Exciter and Rectifier Tab**Exciter gain and time constant**

The gain K_e and time constant T_e of the first-order system representing the exciter. Default is [0.05 0.46].

Field voltage values

The exciter saturation function is defined as a multiplier of exciter alternator output voltage to represent the increase in exciter excitation requirements due to saturation [1]. The saturation function is determined by specifying two voltage points, E_{fd1} and E_{fd2} in p.u., on the air-gap line and Constant Resistance Load Saturation curve, and providing the corresponding two saturation multipliers Se_{Efd1} and Se_{Efd2} . Default is [3.05 2.29].

Typically, the voltage E_{fd1} is a value near the expected exciter maximum output voltage. The E_{fd2} value is about 75% of E_{fd1} .

Exciter saturation function values

The exciter saturation function is defined as a multiplier of exciter alternator output voltage to represent the increase in exciter excitation requirements due to saturation. The saturation function is determined by specifying two voltage points, E_{fd1} and E_{fd2} in p.u., on the air-gap line and Constant Resistance Load saturation curve, and providing the corresponding two saturation multipliers Se_{Efd1} and Se_{Efd2} . Default is [0.279 0.117].

Se_{Efd1} and Se_{Efd2} multipliers are equal to $A-B / B$, A is the value of exciter field current on the Constant Resistance Load saturation curve corresponding to the selected E_{fd} voltage, and B the value of exciter field current on the air-gap line corresponding to the selected E_{fd} voltage [1].

If you do not want to model the saturation effect, set Se_{Ve1} and Se_{Ve2} values to zero.

Initial Values Tab**Initial values of terminal voltage and field voltage**

The initial values of terminal voltage V_{t0} and field voltage E_{fd0} , both in p.u. Initial terminal voltage is normally set to 1 pu. The V_{t0} and E_{fd0} values can be determined using the Powergui Load Flow tool. Default is [1 1].

Sample time

Specify a value greater than zero to discretize the block at the given sample time. Set to -1 to inherit the simulation type and sample time parameters of the Powergui block. Default is 0.

Ports

V_{ref}

The reference value of the stator terminal voltage, in p.u.

V_t

The measured value in p.u. of the stator terminal voltage of the controlled Synchronous Machine block.

Vstab

Connect this input to a power system stabilizer to provide additional stabilization of power system oscillations. When you do not use this option, connect to a Simulink ground block. The input is in p.u.

Efd

The field voltage to apply to the V_f input of the controlled Synchronous Machine block. The output is in p.u.

References

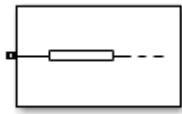
- [1] "IEEE Recommended Practice for Excitation System Models for Power System Stability Studies." *IEEE Standard*, Vol. 421, No. 5, 2005 (Revision of *IEEE 521.5-1992*).

Version History

Introduced before R2006a

Decoupling Line

Single-phase decoupling distributed parameters line



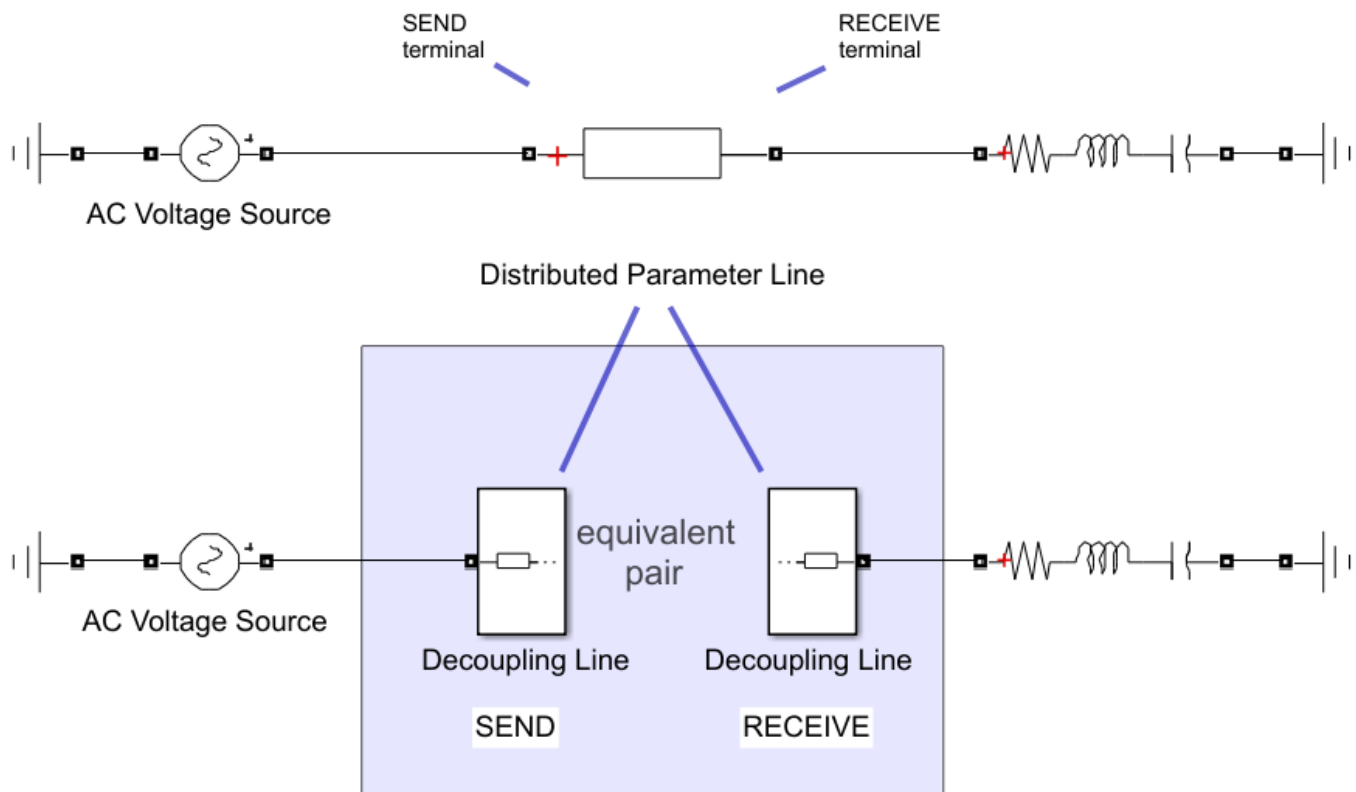
Libraries:

Simscape / Electrical / Specialized Power Systems / Power Grid Elements

Description

The Decoupling Line block implements a single-phase distributed parameters line model that transmits the electrical signal entering its conserving port to an associated Decoupling Line block in your model. The two Decoupling Line blocks form a pair that represents the send and receive ends of a single-phase transmission line.

The behavior of the pair of blocks is equivalent to using one Distributed Parameters Line block:



Refer to Distributed Parameters Line for a detailed description of the equations used to represent wave propagation phenomena and line-end reflections.

Use the Decoupling Line block pairs when you want to model a transmission line with no physical electrical connections between the send and receive ends of the line. This technique is used, for

example, in applications where a large network is decoupled into smaller sections that are simulated into different subsystems or models with no electrical connections between them. This decoupling technique is often used to model complex electrical networks on multi-core real-time computers, or to allow faster simulation of large networks.

The blocks in the pair can be associated by using the **Pairing ID** parameter, which transmits the electrical signal information travelling from one terminal to the other using internal Simulink Goto and From blocks or connecting the blocks with Simulink signals.

The parameters of each Decoupling Line block in a pair must be identical and correspond to the parameters you would enter in an equivalent Distributed Parameters Line block. For example, to model a 100 km transmission line with the given R, L, and C values, use two Decoupling Line blocks with a length of 100 km each and the same R, L, and C values.

To model a transmission line in your model, add two Decoupling Line blocks to your model and set the block parameters to the same values. Alternatively, use the Specify Decoupling Lines App to find, select, and automatically replace existing Distributed Parameters Line blocks in your model with equivalent pairs of Decoupling Line blocks. For more information, see Specify Decoupling Lines.

Limitations

- The Decoupling Line block can be used only in a discrete model, that is, when the **Simulation type** parameter of the powergui block is set to **Discrete**.

Ports

Input

r — Receiving voltage signal
Simulink input signal

Connect this input to the **s** output port of the other Decoupling Line block forming the pair.

Dependencies

To enable this port, select the **Show send and receive ports** parameter.

Output

s — Sending voltage signal
Simulink output signal

Connect this output to the **r** input port of the other Decoupling Line block forming the pair.

Dependencies

To enable this port, select the **Show send and receive ports** parameter.

Conserving

Cons 1 — Electric terminal of line
specialized electrical

Specialized electrical conserving port associated with the send terminal or the receive terminal of a transmission line (when the block is used with another Decoupling Line block).

Parameters

Parameters

Line length (km) — Transmission line length

100 (default) | positive scalar

Transmission line length, in km. Because the block is used in a pair with another Decoupling Line block, enter the same line length in each of the two blocks; do not split the value between each of the two blocks. For example, to model a 100 km line, specify 100 in each block, not 50 in one block and 50 in the other block.

Frequency used for R L C specification (Hz) — Line frequency

60 (default) | positive scalar

Specify the frequency used to compute the per-unit length of the resistance (R), inductance (I), and capacitance (C) matrices of the line model.

Resistance per unit length (ohms/km) — Line resistance

0.01273 (default) | positive scalar

Resistance R per unit length, in ohms/km (Ω/km).

Inductance per unit length (H/km) — Line inductance

0.9337e-3 (default) | positive scalar

Inductance I per unit length, in henries/km (H/km).

Capacitance per unit length (F/km) — Line capacitance

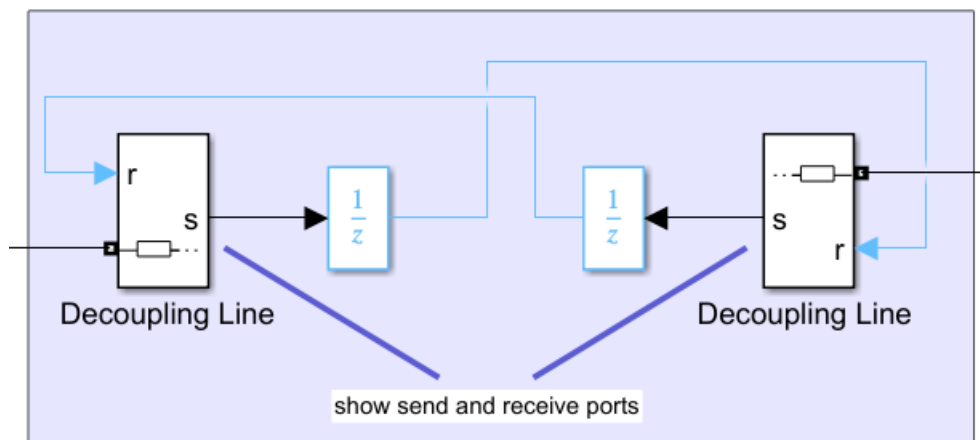
12.74e-9 (default) | positive scalar

Capacitance C per unit length, in farads/km (F/km).

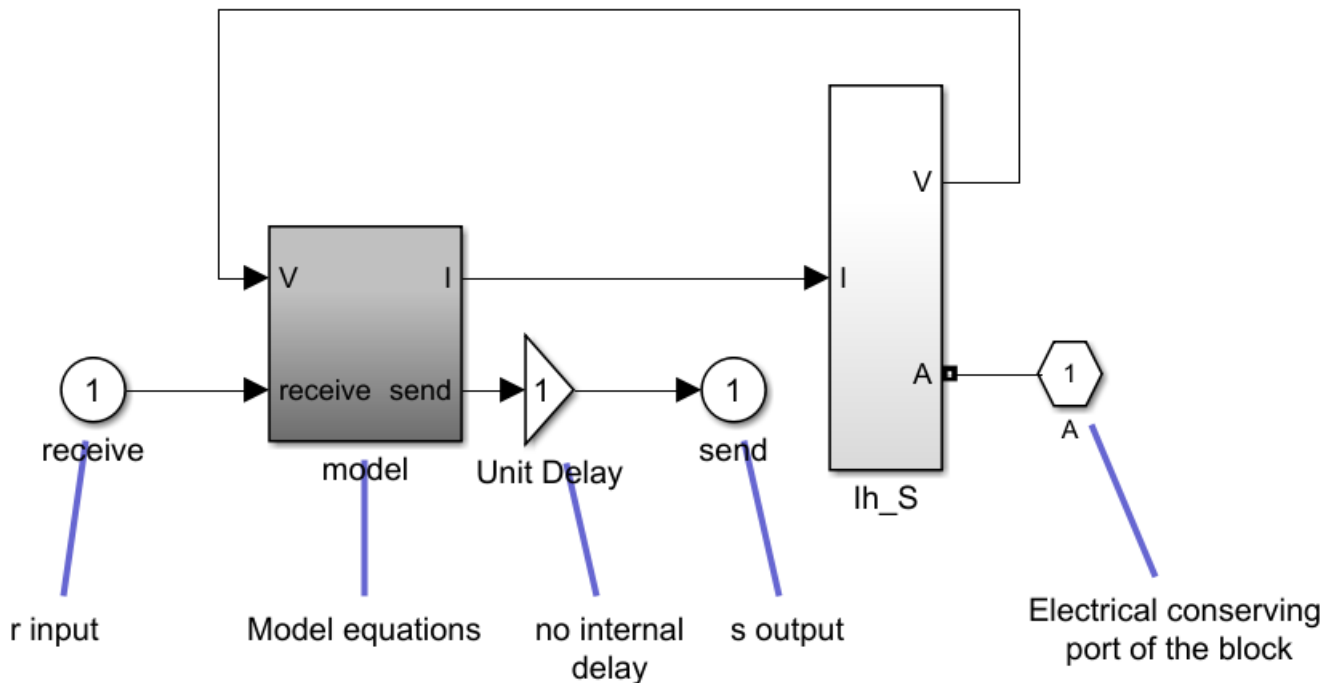
Show send and receive ports — Ports enabling

off (default) | on

Select to show the input and output ports **r** and **s**. When selected, the Decoupling Line blocks can be paired by connecting them with Simulink signals. You connect the **s** port of the first block to the **r** port of the second block, and connect the **s** port of the second block to the **r** port of the first block.

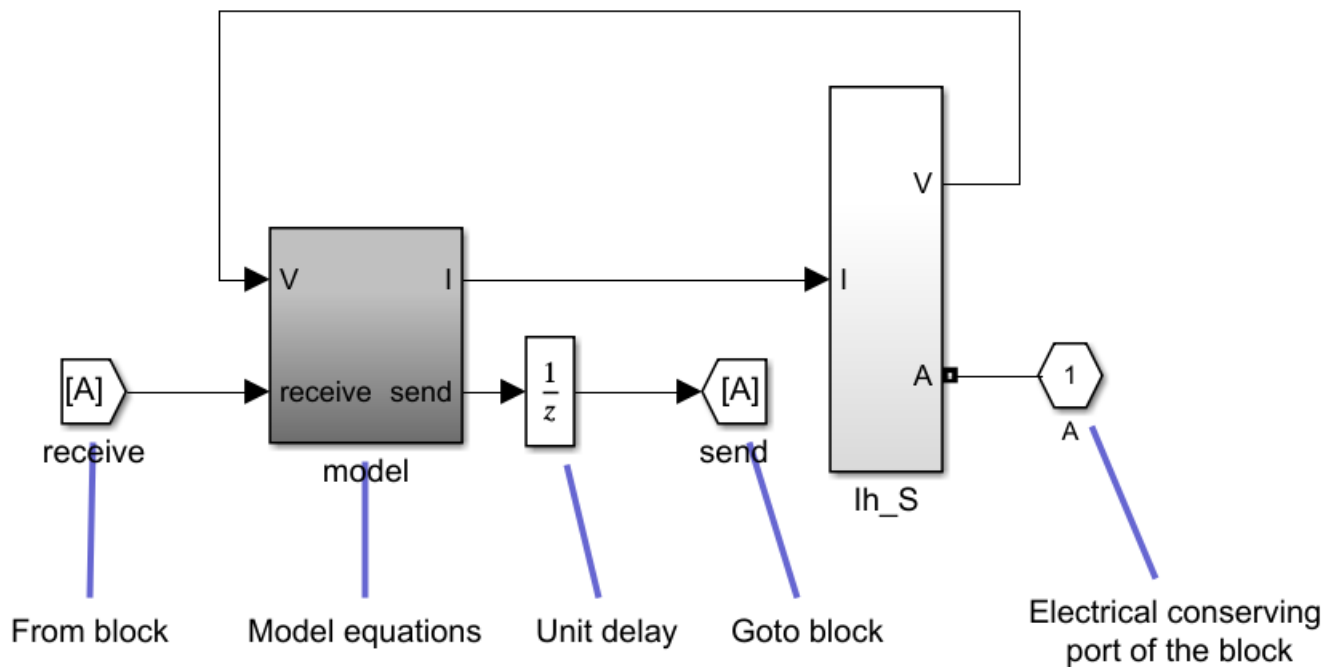


The internal model of the Decoupling Line block is directly connected to the external **r** and **s** block ports.



To break the algebraic loop created by this type of connection, you can add a Unit Delay block at the output of each Decoupling Line block.

When **Show send and receive ports** is cleared, the Decoupling Line blocks are associated by using the **Pairing ID** parameter, which transmits the electrical signal information travelling from one terminal to the other by the use of internal Simulink Goto and From blocks. A Unit Delay block is also added to the output signal path to break the algebraic loops. The sample time parameter of this delay corresponds to the sample time delay specified in the powergui block.



Pairing ID — Pairing identifier
L001 (default)

When the **Show send and receive ports** parameter is cleared, internal Goto and From blocks are used to pair two Decoupling Line blocks in your model. In both blocks, the first part of the pairing identifier is the same, but you must append **_S** to the end of the identifier of one block, and **_R** to the end of the identifier of the other block. An error is produced if the identifier does not end in **_S** or **_R**. For example, suppose you add two Decoupling Line blocks to your model. To pair the blocks, you must set the **Pairing ID** parameter of one block to **L001_S** and the other block to **L001_R**.

Initial Values

Initial voltage (V) — Initial voltage
1 (default) | scalar

Initial value of the terminal voltage, in volts.

Initial voltage angle (Deg) — Initial angle
0 (default) | scalar

Angle of the initial voltage, in degrees.

Initial current (A) — Initial current
1 (default) | scalar

Initial value of the current flowing out the block, in Amperes.

Initial current angle (Deg) — Initial angle
0 (default) | scalar

Angle of the initial current, in degrees.

Version History

Introduced in R2020b

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

[Decoupling Line \(Three-Phase\)](#) | [Distributed Parameters Line](#) | [Specify Decoupling Lines](#)

Decoupling Line (Three-Phase)

Three-phase decoupling distributed parameters line



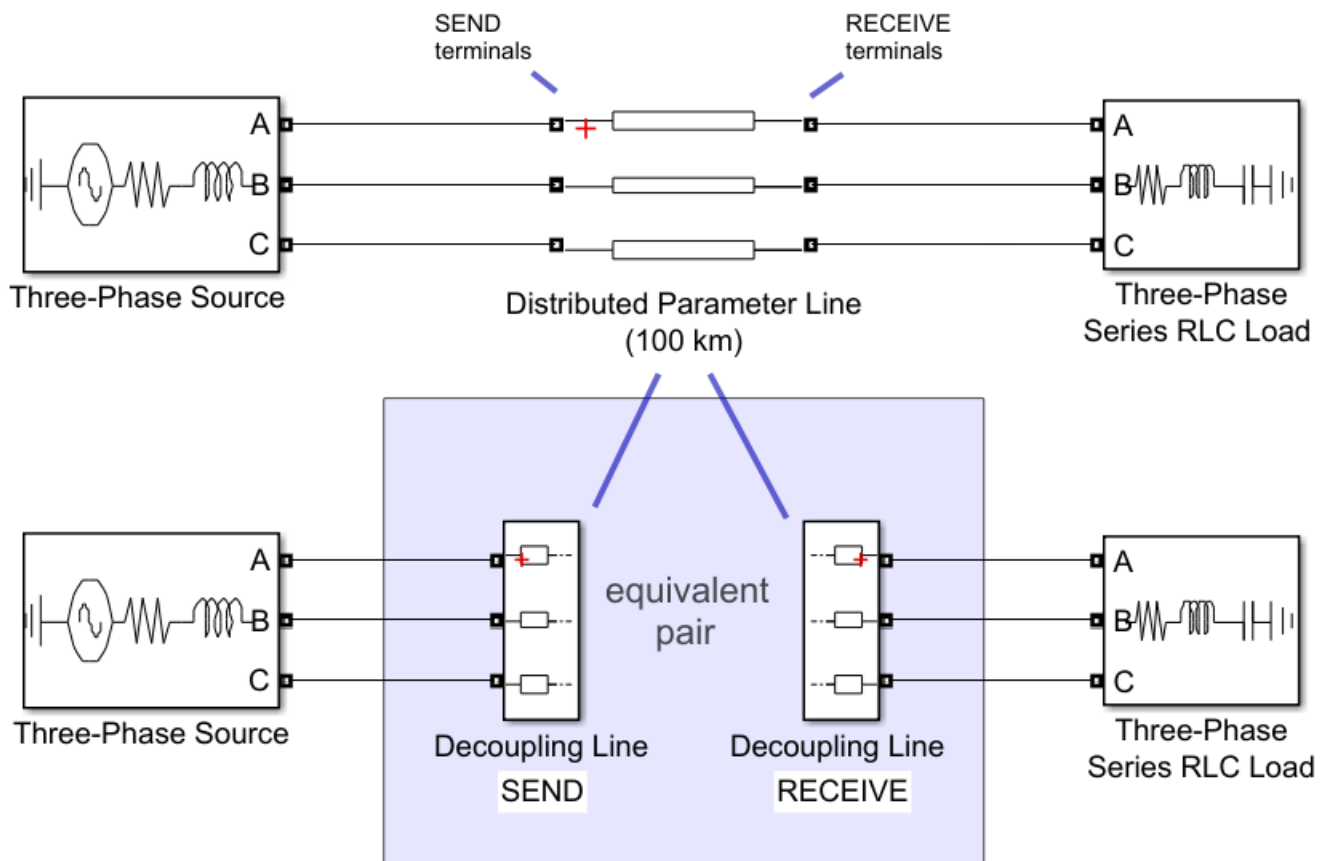
Libraries:

Simscape / Electrical / Specialized Power Systems / Power Grid Elements

Description

The Decoupling Line (Three-Phase) block implements a three-phase distributed parameters line model that transmits the electrical signal entering its conserving ports to an associated Decoupling Line (Three-Phase) block in your model. The two Decoupling Line (Three-Phase) blocks form a pair that represents the send and receive ends of a three-phase transmission line.

The behavior of the pair of blocks is equivalent to using one Distributed Parameters Line block:



Refer to Distributed Parameters Line for a detailed description of the equations used to represent wave propagation phenomena and line-end reflections.

Use the Decoupling Line (Three-Phase) block pairs when you want to model a transmission line with no physical electrical connections between the send and receive ends of the line. This technique is used, for example, in applications where a large network is decoupled into smaller sections that are simulated into different subsystems or models with no electrical connections between them. This decoupling technique is often used to model complex electrical networks on multi-core real-time computers, or to allow faster simulation of large networks.

The blocks in the a pair can be associated by using the **Pairing ID** parameter, which transmits the electrical signal information travelling from one terminal to the other by using internal Simulink Goto and From blocks or connecting the blocks with Simulink signals.

The parameters of each Decoupling Line (Three-Phase) block in the a pair must be identical and correspond to the parameters you would enter in an equivalent Distributed Parameters Line block. For example, to model a 100 km transmission line with the given R, L, and C values, use two Decoupling Line (Three-Phase) blocks with a length of 100 km each and the same R, L, and C values.

To model a transmission line in your model, add two Decoupling Line (Three-Phase) blocks to your model and set the block parameters to the same values. Alternatively, use the Specify Decoupling Lines App to find, select, and automatically replace existing Distributed Parameters Line blocks in your model with equivalent pairs of Decoupling Line (Three-Phase) blocks. For more information, see Specify Decoupling Lines.

Limitations

- The Decoupling Line (Three-Phase) block can be used only in a discrete model, that is, when the **Simulation type** parameter of the powergui block is set to **Discrete**.

Ports

Input

r — Receiving voltage signal
Simulink input signal

Connect this input to the **s** output port of the other Decoupling Line (Three-Phase) block forming the pair.

Dependencies

To enable this port, select the **Show send and receive ports** parameter.

Output

s — Sending voltage signal
Simulink output signal

Connect this output to the **r** input port of the other Decoupling Line (Three-Phase) block forming the pair.

Dependencies

To enable this port, select the **Show send and receive ports** parameter.

Conserving

Cons 1 — Electric terminal of line
specialized electrical

Specialized electrical conserving port associated with the send terminal or the receive terminal of a transmission line when the block is used in pair with another Decoupling Line (Three-Phase) block.

Cons 2 — Electric terminal of line
specialized electrical

Specialized electrical conserving port associated with the send terminal or the receive terminal of a transmission line when the block is used in pair with another Decoupling Line (Three-Phase) block.

Cons 3 — Electric terminal of line
specialized electrical

Specialized electrical conserving port associated with the send terminal or the receive terminal of a transmission line when the block is used in pair with another Decoupling Line (Three-Phase) block.

Parameters

Parameters

Line length (km) — Transmission line length
100 (default) | positive scalar

Transmission line length, in km. Because the block is used with another Decoupling Line (Three-Phase) block, enter the same line length in each of the two blocks; do not split the value between each of the two blocks. For example, to model a 100 km line, specify 100 in each block, not 50 in one block and 50 in the other block.

Frequency used for rlc specification (Hz) — Line frequency
60 (default) | positive scalar

Specify the frequency used to compute the per-unit length of the resistance R, inductance I, and capacitance C matrices of the line model.

Resistance per unit length (ohms/km) [3x3 matrix] or [r1 r0] — Line resistance
[0.01273 0.3864] (default) | two-element vector | three-element vector | 3-by-3 matrix

Resistance R per unit length, specified as a two-element vector or a 3-by-3 matrix, in ohms/km (Ω/km).

- For a symmetrical line, specify either the 3-by-3 matrix or the sequence parameters.
- For a three-phase continuously transposed line, enter the positive and zero-sequence resistances [r1 r0].
- For asymmetrical lines, specify the complete 3-by-3 resistance matrix.

Inductance per unit length (H/km) [3x3 matrix] or [I1 I0] — Line inductance
[0.9337e-3 4.1264e-3] (default) | two-element vector | three-element vector | 3-by-3 matrix

Inductance I per unit length, specified as a two-element vector or a 3-by-3 matrix, in henries/km (H/km).

- For a symmetrical line, specify either the 3-by-3 matrix or the sequence parameters.
- For a three-phase continuously transposed line, enter the positive and zero-sequence inductances [l1 l0].
- For asymmetrical lines, specify the complete 3-by-3 inductance matrix.

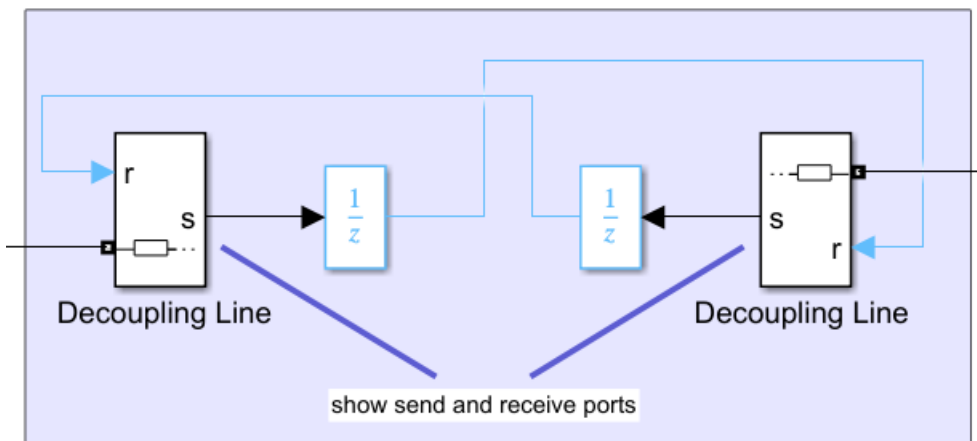
Capacitance per unit length (F/km) [3x3 matrix] or [c1 c0] — Line capacitance
[12.74e-9 7.751e-9] (default) | two-element vector | three-element vector | 3-by-3 matrix

Capacitance C per unit length, specified as a two-element vector or a 3-by-3 matrix, in farads/km (F/km).

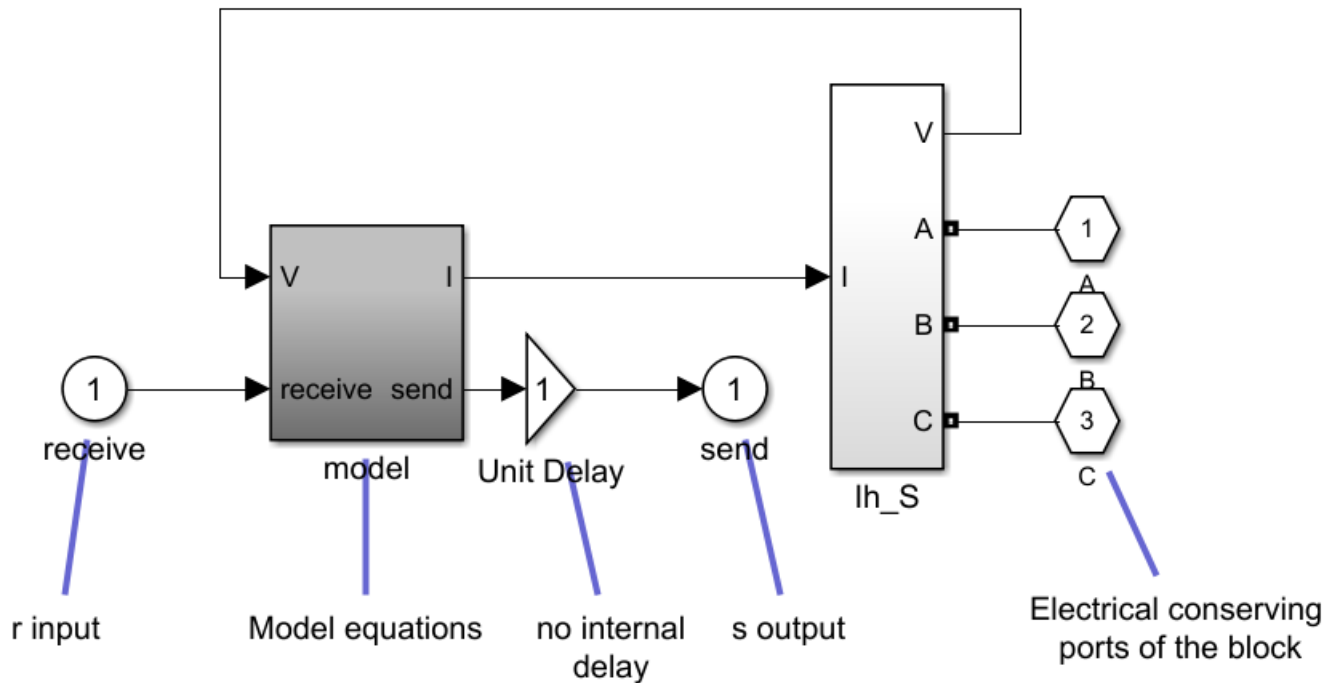
- For a symmetrical line, specify either the 3-by-3 matrix or the sequence parameters.
- For a three-phase continuously transposed line, enter the positive and zero-sequence capacitances [c1 c0].
- For asymmetrical lines, specify the complete 3-by-3 capacitance matrix.

Show send and receive ports — Ports enabling
off (default) | on

Select to show the **r** and **s** input and output ports. When selected, the Decoupling Line (Three-Phase) blocks can be paired by connecting them with Simulink signals. You connect the **s** output of the first block to the **r** input of the second block, and connect the **s** port of the second block to the **r** port of the first block

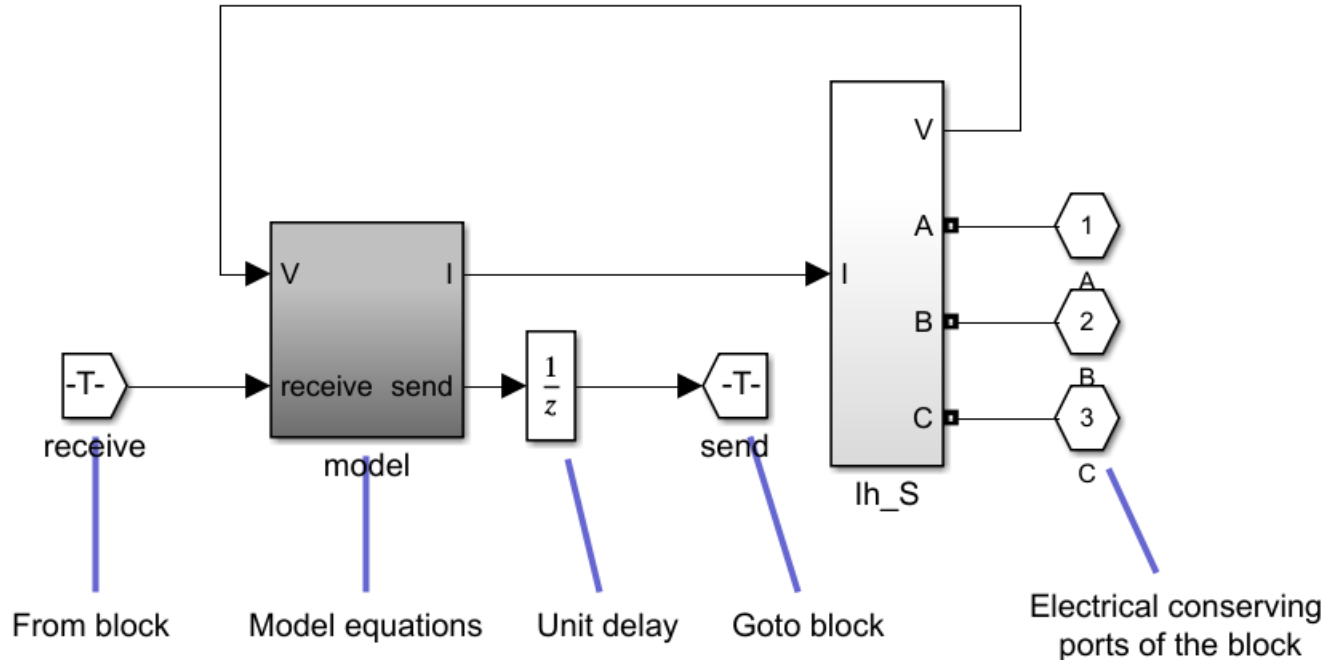


The internal model of the Decoupling Line (Three-Phase) block is directly connected to the external **r** and **s** block ports.



To break the algebraic loop created by this type of connection, you can add Unit Delay block at the output of each Decoupling Line (Three-Phase) block.

When **Show send and receive ports** is cleared, the Decoupling Line (Three-Phase) blocks are associated by using the **Pairing ID** parameter, which transmits the electrical signal information travelling from one terminal to the other by the use of internal Simulink Goto and From blocks. A Unit Delay block is also added to the output signal path to break the algebraic loops. The sample time parameter of this delay corresponds to the sample time delay specified in the powergui block.



Pairing ID — Pairing identifier

L001 (default)

When the **Show send and receive ports** parameter is cleared, internal Goto and From blocks are used to pair two Decoupling Line (Three-Phase) blocks in your model. In both blocks, the first part of the pairing identifier is the same, but you must append **_S** to the end of the identifier of one block, and **_R** to the end of the identifier of the other block. An error is produced if the identifier does not end in **_S** or **_R**. For example, suppose you add two Decoupling Line (Three-Phase) blocks to your model. To pair the blocks, you must set the **Pairing ID** parameter of one block to **L001_S** and the other block to **L001_R**.

Initial Values

Initial voltage (V) — Initial voltage

[1 1 1] (default) | three-element vector

Initial value of the terminal voltage, in volts.

Initial voltage angle (Deg) — Initial angle

[0 -120 120] (default) | three-element vector

Angle of the initial voltage, in degrees.

Initial current (A) — Initial current

[1 1 1] (default) | three-element vector

Initial value of the current flowing out the block, in Amperes.

Initial current angle (Deg) — Initial angle

[0 -120 120] (default) | three-element vector

Angle of the initial current, in degrees.

Version History

Introduced in R2020b

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

[Decoupling Line](#) | [Distributed Parameters Line](#) | [Specify Decoupling Lines](#)

Digital Flickermeter

Implement digital flickermeter as described in IEC 61000-4-15 standard



Libraries:

Simscape / Electrical / Specialized Power Systems / Sensors and Measurements

Description

The Digital Flickermeter block implements a digital flickermeter as described in the IEC 61000-4-15 standard. The model implements the first four modules of the design. The statistical analysis (module 5) is implemented by an external function for offline analysis of recorded simulation signal.

Note You must have a Control System Toolbox™ license to use the Digital Flickermeter block.

Ports

Input

enable — Option to enable flicker measurement

1 | 0

Set this input to 1 to enable the flicker measurement. The flickermeter is normally enabled after a given period of time, when the analyzed signal (input U) is stable.

u — Analyzed signal

Analyzed signal.

Output

S5 — Instantaneous flicker sensation

Instantaneous flicker sensation measured for the analyzed signal.

m — Measurement signal

vector

This output contains these measurements:

- Output 1: Half-cycle, rms-normalized voltage (pu)
- Output 2: Weighted voltage fluctuation
- Output 3: Square root of Output_5
- Output 4: One-minute dose (integral) of instant perception
- Output 5: Instantaneous flicker sensation (Output S5)

- **u**: The analyzed signal
- **Init**: initialization period time

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Open statistical analysis — Open statistical tool

Opens the Statistical Analysis of Instantaneous Flicker Level module tool.

Network frequency — Frequency of the analyzed signal

60 Hz (default) | 50 Hz

Frequency of the analyzed signal. Choose between 60 Hz and 50 Hz.

Nominal input phase voltage (Vrms) — Nominal voltage value of analyzed signal

$25\sqrt{3}$ (default)

Nominal voltage value of the analyzed signal, in volts rms.

Maximum input voltage (pu) — Maximum input voltage value of input voltage adaptor

5.0 (default)

Maximum input voltage value of the input voltage adaptor (module 1), in per unit.

Maximum instantaneous sensation (pu) — Maximum value of instantaneous flicker sensation

1000 (default)

Maximum value of the instantaneous flicker sensation measured by the block, in per unit.

Sample time (s) — Sampling rate

0.001 (default)

Sampling rate used to compute the instantaneous flicker sensation.

Scaling factor for sensation perceptibility — Scaling factor of flickermeter for measuring sensation perceptibility

1.24×10^6 (default)

Scaling factor of the flickermeter for measuring the sensation perceptibility.

Version History

Introduced in R2015b

References

- [1] *Electromagnetic compatibility (EMC) – Part 4: Testing and measurement techniques – Section 15: Flickermeter – Functional and design specifications*, IEC-61000-4-15, 2003-02.
- [2] Gutierrez, J. J., J. Ruiz, A. Lazkano, and L. A. Leturiondo. "Measurement of Voltage Flicker: Application to Grid-connected Wind Turbines." In *Advances in Measurement Systems*, edited

by Milind Kr Sharma, 365–380. InTech, DOI: 10.5772/8731, 2010. Available from: <http://www.intechopen.com/books/advances-in-measurement-systems/measurement-of-voltage-flicker-application-to-grid-connected-wind-turbines>.

Extended Capabilities

C/C++ Code Generation

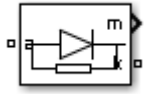
Generate C and C++ code using Simulink® Coder™.

See Also

`power_flicker`

Diode

Implement diode model

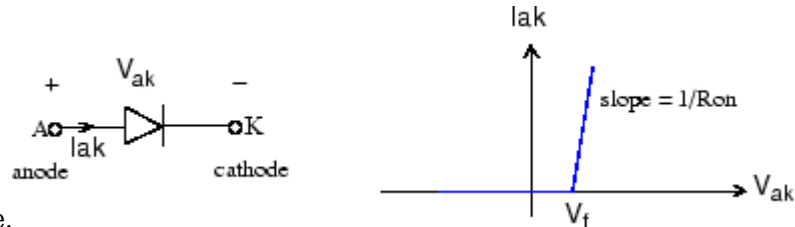


Libraries:

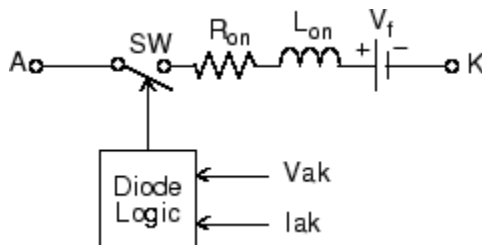
Simscape / Electrical / Specialized Power Systems / Power Electronics

Description

The diode is a semiconductor device that is controlled by its own voltage V_{ak} and current I_{ak} . When a diode is forward biased ($V_{ak} > 0$), it starts to conduct with a small forward voltage V_f across it. It turns off when the current flow into the device becomes 0. When the diode is reverse biased ($V_{ak} < 0$), it stays in the off state.



The Diode block is simulated by a resistor, an inductor, and a DC voltage source connected in series with a switch. The switch operation is controlled by the voltage V_{ak} and the current I_{ak} .



The Diode block also contains a series Rs-Cs snubber circuit that can be connected in parallel with the diode device (between nodes A and K).

Examples

The `power_diode` example illustrates a single pulse rectifier consisting of a Diode block, an RL load, and an AC Voltage source block.

Assumptions and Limitations

- The Diode block implements a macro model of a diode device. It does not take into account either the geometry of the device or the complex physical processes underlying the state change [1]. The leakage current in the blocking state and the reverse-recovery (negative) current are not considered. In most circuits, the reverse current does not affect converter or other device characteristics.

- Depending on the value of the inductance L_{on} , the diode is modeled either as a current source ($L_{on} > 0$) or as a variable topology circuit ($L_{on} = 0$). The Diode block cannot be connected in series with an inductor, a current source, or an open circuit, unless its snubber circuit is in use.
- The inductance L_{on} is forced to 0 if you choose to discretize your circuit.

Ports

Output

m — Measurement vector
Simulink signal

Measurement vector containing two signals. You can demultiplex these signals by using the Bus Selector block provided in the Simulink library.

Signal	Definition	Units
1	Diode current	A
2	Diode voltage	V

Dependencies

To enable this port, check the **Show measurement port** parameter.

Conserving

a — Anode
specialized electrical

Specialized electrical conserving port associated with the anode.

k — Cathode
specialized electrical

Specialized electrical conserving port associated with the cathode.

Parameters

Resistance R_{on} (Ohms) — Diode internal resistance
0.001 (default)

Diode internal resistance R_{on} , in ohms (Ω). Default is 0.001. The **Resistance R_{on}** parameter cannot be set to 0 when the **Inductance L_{on}** parameter is set to 0.

Inductance L_{on} (H) — Diode internal inductance
0 (default)

Diode internal inductance L_{on} , in henries (H). Default is 0. The **Inductance L_{on}** parameter cannot be set to 0 when the **Resistance R_{on}** parameter is set to 0.

Forward voltage V_f (V) — Diode forward voltage
0.8 (default)

Forward voltage of the diode device, in volts (V). Default is 0.8.

Initial current Ic (A) — Initial current

0 (default)

Specifies an initial current flowing in the diode device. Default is 0. It is usually set to 0 to start the simulation with the diode device blocked. If the **Initial Current IC** parameter is set to a value greater than 0, the steady-state calculation considers the initial status of the diode as closed.

Initializing all states of a power electronic converter is a complex task. Therefore, this option is useful only with simple circuits.

Snubber resistance Rs (Ohms) — Snubber resistance

500 (default)

Snubber resistance, in ohms (Ω). Default is 500. Set the **Snubber resistance Rs** parameter to `inf` to eliminate the snubber from the model.

Snubber capacitance Cs (F) — Snubber capacitance

250e-9 (default)

Snubber capacitance in farads (F). Default is 250e-9. Set the **Snubber capacitance Cs** parameter to 0 to eliminate the snubber, or to `inf` to get a resistive snubber.

Show measurement port — Measurement port visibility option

on (default) | off

If selected, adds a Simulink output to the block returning the diode current and voltage. Default is selected.

Version History

Introduced before R2006a

References

- [1] Rajagopalan, V., *Computer-Aided Analysis of Power Electronic Systems*, Marcel Dekker, Inc., New York, 1987.
- [2] Mohan, N., T.M. Undeland, and W.P. Robbins, *Power Electronics: Converters, Applications, and Design*, John Wiley & Sons, Inc., New York, 1995.

Extended Capabilities

C/C++ Code Generation

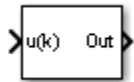
Generate C and C++ code using Simulink® Coder™.

See Also

Thyristor | Universal Bridge

Discrete Shift Register

Implement serial-in, parallel-out shift register



Library

Simscape / Electrical / Specialized Power Systems / Control

Description

The Discrete Shift Register block outputs a vector containing the last N samples of the input signal. When the input contains more than one signal, the block outputs the last N samples of each signal in the following order:

$$\text{Out} = [u_1(k), u_1(k-1), u_1(k-2), u_1(k-3), u_2(k), u_2(k-1), u_2(k-2), u_2(k-3)]$$

This example shows the block output for an input containing two signals, represented by u_1 and u_2 , and a number of samples $N = 4$, represented by the k to $k-3$ indices. The dimension of the output vector is $4 \times 2 = 8$.

Parameters

Number of samples N

Specify the number of samples, or stages, of the register. The minimum value is 1. Default is 32.

Initial inputs

Specify the initial value of the $N-1$ samples preceding time 0. Enter a scalar value or a vector of the same size as the input signal. Default is 0.

Sample time

Specify the time interval between the samples. Default is $50\text{e-}6$.

Characteristics

Direct Feedthrough	Yes
Sample Time	Discrete
Dimensionalized	Yes
Scalar Expansion	Yes, of the parameter Initial inputs
Zero-Crossing Detection	No

Examples

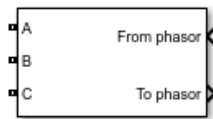
The `power_DiscreteShiftRegister` example shows various uses of the Discrete Shift Register block.

Version History

Introduced in R2013a

Discrete-to-Phasor Interface

Connect Discrete network to Discrete Phasor network



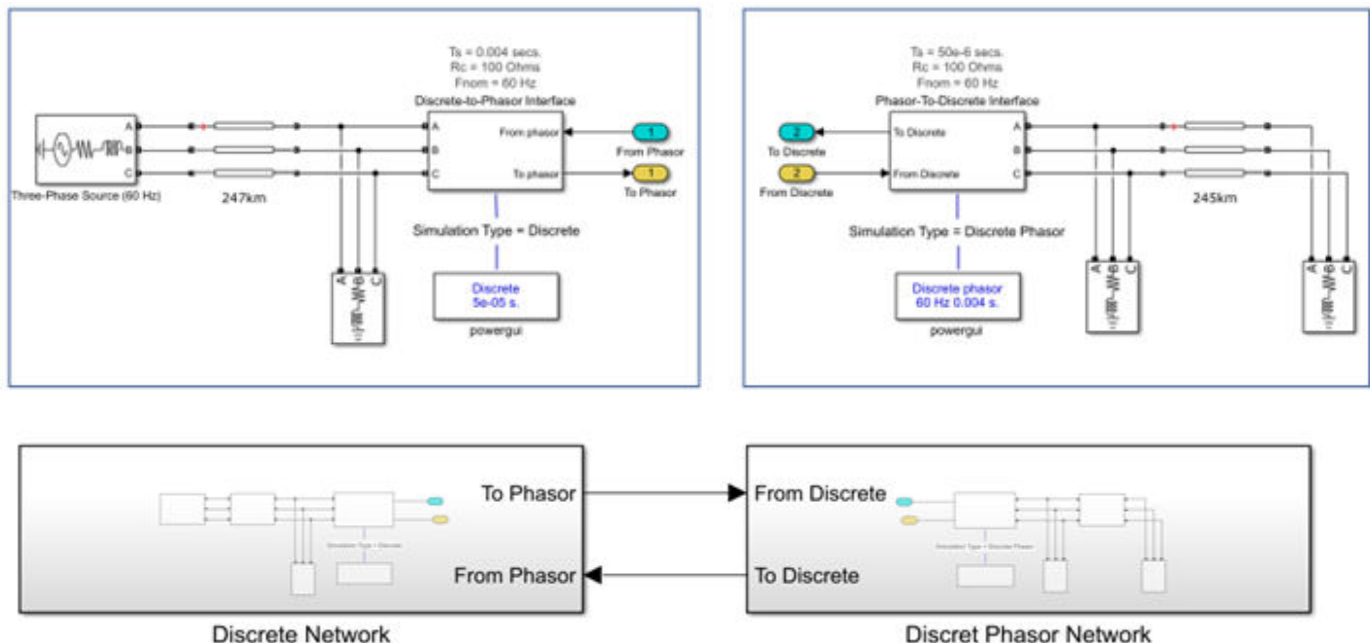
Libraries:

Simscape / Electrical / Specialized Power Systems / Utilities

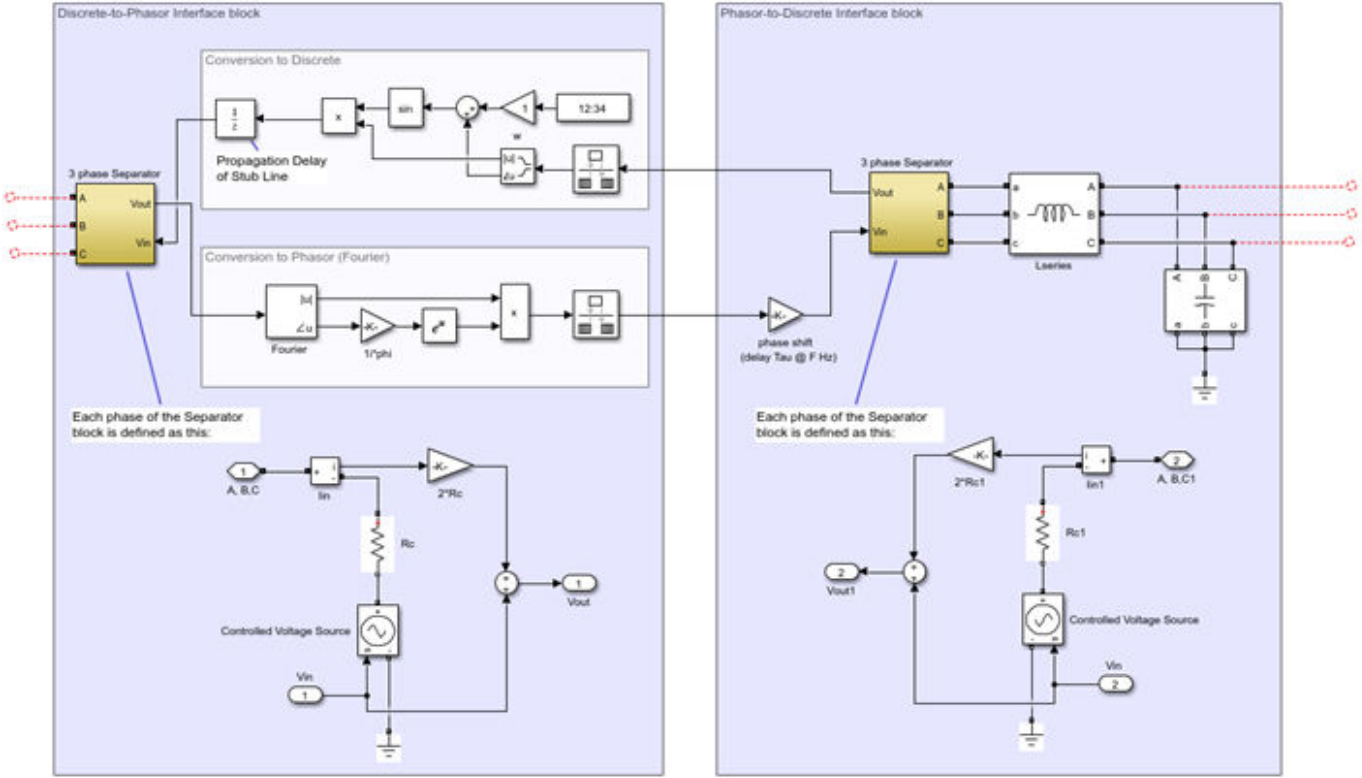
Description

Use the Discrete-to-Phasor Interface block with a Phasor-to-Discrete block to connect a network simulated in Discrete mode to a network simulated in Discrete Phasor mode. The Discrete-to-Phasor Interface block must be used in a subsystem or Model Reference that has a Powergui block with **Simulation type** set to **Discrete**. The electrical network that this subsystem contains is called the Discrete Network.

An associated Discrete-to-Phasor Interface block must be used in another subsystem or Model Reference that has a Powergui block with **Simulation type** set to **Discrete phasor**. The electrical network that this subsystem contains is called the Discrete Phasor Network.



The Discrete-to-Phasor Interface block and the Phasor-to-Discrete Interface block form a decoupling device system. Each phase terminal of the Discrete-to-Phasor Interface block consists of a controlled Voltage Source block in series with a resistor and a Current Measurement block, as shown in the left section of this model:



The Discrete-to-Phasor Interface block converts the voltages measured at the A, B, and C terminals of the block into phasor voltages that are sent to the Phasor-to-Discrete Interface block. It also converts the phasor voltages received by the Phasor-to-Discrete block into discretized voltages. Because the Discrete-to-Phasor Interface block and the Phasor-to-Discrete Interface block act together to model a single decoupling system, the values specified for the resistance R_c , nominal frequency F , and sample time T_s parameters must be the same in the two blocks.

Ports

Conserving

A — Phase A terminal
specialized electrical

Electrical conserving port associated with the phase A terminal of the block.

B — Phase B terminal
specialized electrical

Electrical conserving port associated with the phase B terminal of the block.

C — Phase C terminal
specialized electrical

Electrical conserving port associated with the phase C terminal of the block.

Simulink

From phasor — Three-phase voltage
Vector of 3 elements

Signal representing the three-phase voltage coming from the Phasor-to-Discrete Interface block. Connect this input to an Inport block of the subsystem or Model Reference containing the block.

To phasor — Three-phase voltage
Vector of 3 elements

Signal that measures the three-phase voltage at the terminals of the Discrete-to-Phasor Interface block. Connect this signal to an Outport block of the subsystem or Model Reference containing the block.

Parameters

Rc (Ohms) — Decoupling Resistance
100 (default) | positive scalar

Resistance of the decoupling equivalent model, in ohms. This value must be the same as the value of the **Rc (Ohms)** parameter of the associated Phasor-to-Discrete Interface block.

Nominal frequency (Hz) — Network nominal frequency
60 (default) | positive scalar

Nominal frequency of the system, in hertz. The block uses this value to convert the phasor and discrete signals coming into and going out of the block. The value must be same as the value of the **Nominal frequency** parameter of the associated Phasor-to-Discrete Interface block.

Sample time of phasor network (s) — Phasor network sample time
1e-4 (default) | positive scalar

Sample time, in seconds, of the Discrete Phasor network. The block uses this value to convert the phasor and discrete signals coming into and going out of the block. This value must be the same as the value of the **Sample time** parameter of the Powergui block of the Discrete Phasor network.

Version History

Introduced in R2023a

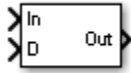
Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using Simulink® Coder™.

See Also
Phasor-to-Discrete Interface

Discrete Variable Time Delay

Delay signal by variable time value



Library

Simscape / Electrical / Specialized Power Systems / Control

Description

The Discrete Variable Time Delay block delays the input signal by the value specified in the D input. At each simulation time step, the Discrete Variable Time Delay block saves the time and the input value in an internal buffer and outputs the previous input value determined by the delay input.

When the delay input is not an integer of the time step, the Discrete Variable Time Delay block performs linear interpolation. When the **Direct feedthrough** parameter is selected, the delay can be less than the simulation time step.

Parameters

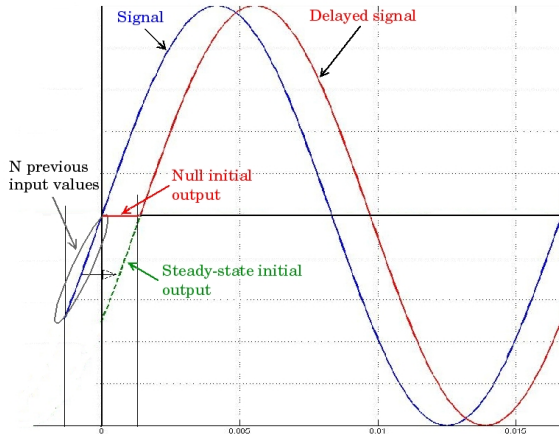
Maximum delay (s)

Specify the expected maximum delay of the D input delay, in seconds. The **Maximum delay** parameter determines the input buffer size. When the maximum delay is exceeded, the delay is clipped to the maximum delay value. Default is 1.

Initial conditions

Specify the initial conditions of the signal. The block outputs the specified values until the first delayed sample of signal is out. Default is 0.

Set to 0 to implement null initial output, or specify the N previous input values of signal to start from steady-state. The required number of input values is $N = \text{ceil}(D1/T_s)$ where D1 is the value of the delay at $t = 0$, and T_s the sample time value. The figure shows the details of the initial conditions.



Suppose, for example, that the signal is a 60 Hz sine wave, the initial delay is set to $D1 = 10 \cdot T_s$ by the D input of the block, and the sample time is equal to $50 \cdot 10^{-6}$ s. The N previous input values of the signal are determined as follows:

```
N = ceil(10*Ts)/Ts; % N=10
t = [-N*Ts:Ts:-1*Ts]; % time vector of the N previous inputs
S_init = sin(2*pi*60*t); % The N previous input values of signal
```

Sample time

Specify the time interval between the samples. Default is $50 \cdot 10^{-6}$. Minimum value is $1 \cdot 10^{-8}$.

Direct feedthrough

When this check box is selected, the block can handle a delay less than the simulation time step. Otherwise the block extrapolates the input values. Default is selected.

Inputs and Outputs

In

The signal. The input can be a scalar signal or a vector of signals. The dimension of the In and D inputs must be the same.

D

The value of the delay to apply to the signal. The input can be a scalar delay signal or a vector of delay signals. The dimensions of the In and D inputs must be the same.

Out

Returns the delayed signal.

Characteristics

Direct Feedthrough	Yes
Sample Time	Discrete
Scalar Expansion	Yes, of input D and of the Initial conditions parameter
Dimensionalized	Yes
Zero-Crossing Detection	No

Example

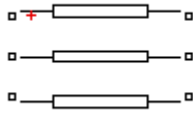
The `power_DiscreteVariableTimeDelay` example shows various uses of the Discrete Variable Time Delay block.

Version History

Introduced in R2013a

Distributed Parameters Line

Implement N-phase distributed parameter transmission line model with lumped losses



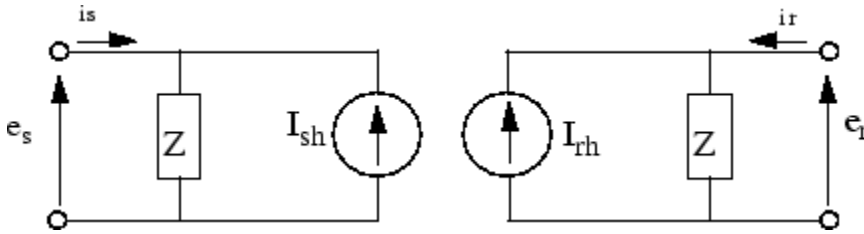
Libraries:

Simscape / Electrical / Specialized Power Systems / Power Grid Elements

Description

The Distributed Parameters Line block implements an N-phase distributed parameter line model with lumped losses. The model is based on the Bergeron's traveling wave method used by the Electromagnetic Transient Program (EMTP) [1]. In this model, the lossless distributed LC line is characterized by two values (for a single-phase line): the surge impedance $Z_c = \sqrt{l/c}$ and the wave propagation speed $v = 1/\sqrt{lc}$. l and c are the per-unit length inductance and capacitance.

The figure shows the two-port model of a single-phase line.



For a lossless line ($r = 0$), the quantity $e + Z_c i$, where e is the line voltage at one end and i is the line current entering the same end, must arrive unchanged at the other end after a transport delay τ .

$$\tau = \frac{d}{v}$$

where d is the line length and v is the propagation speed.

The model equations for a lossless line are:

$$e_r(t) - Z_c i_r(t) = e_s(t - \tau) + Z_c i_s(t - \tau)$$

$$e_s(t) - Z_c i_s(t) = e_r(t - \tau) + Z_c i_r(t - \tau)$$

knowing that

$$i_s(t) = \frac{e_s(t)}{Z} - I_{sh}(t)$$

$$i_r(t) = \frac{e_r(t)}{Z} - I_{rh}(t)$$

In a lossless line, the two current sources I_{sh} and I_{rh} are computed as:

$$I_{sh}(t) = \frac{2}{Z_C} e_r(t - \tau) - I_{rh}(t - \tau)$$

$$I_{rh}(t) = \frac{2}{Z_C} e_s(t - \tau) - I_{sh}(t - \tau)$$

When losses are taken into account, new equations for I_{sh} and I_{rh} are obtained by lumping $R/4$ at both ends of the line and $R/2$ in the middle of the line:

$$R = \text{total resistance} = r \times d$$

The current sources I_{sh} and I_{rh} are then computed as follows:

$$I_{sh}(t) = \left(\frac{1+h}{2} \right) \left(\frac{1+h}{Z} e_r(t - \tau) - h I_{rh}(t - \tau) \right) + \left(\frac{1-h}{2} \right) \left(\frac{1+h}{Z} e_s(t - \tau) - h I_{sh}(t - \tau) \right)$$

$$I_{rh}(t) = \left(\frac{1+h}{2} \right) \left(\frac{1+h}{Z} e_s(t - \tau) - h I_{sh}(t - \tau) \right) + \left(\frac{1-h}{2} \right) \left(\frac{1+h}{Z} e_r(t - \tau) - h I_{rh}(t - \tau) \right)$$

where

$$Z = Z_C + \frac{r}{4}$$

$$h = \frac{Z_C - \frac{r}{4}}{Z_C + \frac{r}{4}}$$

$$Z_C = \sqrt{\frac{l}{c}}$$

$$\tau = d\sqrt{lc}$$

r, l, c are the per unit length parameters, and d is the line length. For a lossless line, $r = 0$, $h = 1$, and $Z = Z_C$.

For multiphase line models, modal transformation is used to convert line quantities from phase values (line currents and voltages) into modal values independent of each other. The previous calculations are made in the modal domain before being converted back to phase values.

In comparison to the PI section line model, the distributed line represents wave propagation phenomena and line end reflections with much better accuracy.

Assumptions and Limitations

This model does not represent accurately the frequency dependence of RLC parameters of real power lines. Indeed, because of the skin effects in the conductors and ground, the R and L matrices exhibit strong frequency dependence, causing an attenuation of the high frequencies.

Ports

Conserving

Cons 1 — Electric terminal
specialized electrical

Specialized electrical conserving port associated with terminal 1 on the sending end of the line.

Cons 2 — Electric terminal
specialized electrical

Specialized electrical conserving port associated with terminal 2 on the sending end of the line.

Cons 3 — Electric terminal
specialized electrical

Specialized electrical conserving port associated with terminal 3 on the sending end of the line.

Cons 4 — Electric terminal
specialized electrical

Specialized electrical conserving port associated with terminal 1 on the receiving end of the line.

Cons 5 — Electric terminal
specialized electrical

Specialized electrical conserving port associated with terminal 2 on the receiving end of the line.

Cons 6 — Electric terminal
specialized electrical

Specialized electrical conserving port associated with terminal 3 on the receiving end of the line.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Number of phases [N] — Number of phases
3 (default) | scalar greater than or equal to 1

Number of phases, N, of the model. The block icon dynamically changes according to the number of phases that you specify. When you apply the parameters or close the dialog box, the number of inputs and outputs is updated.

Line length (km) — Line section length
100 (default) | positive scalar

Line length in kilometers (km).

Frequency used for rlc specification (Hz) — Frequency used for specification of per unit length rlc line parameters
60 (default) | positive scalar

Frequency used to compute the per unit length resistance r , inductance l , and capacitance c matrices of the line model.

Resistance per unit length (Ohms/km) [NxN matrix] or [r1 r0 r0m] — Resistance per unit length

[0.01273 0.3864] (default) | matrix of N-by-N elements

Resistance r per unit length, as an N-by-N matrix in ohms/km (Ω/km).

For a symmetrical line, you can either specify the N-by-N matrix or the sequence parameters. For a two-phase or three-phase continuously transposed line, you can enter the positive and zero-sequence resistances [$r1$ $r0$]. For a symmetrical six-phase line you can enter the sequence parameters plus the zero-sequence mutual resistance [$r1$ $r0$ $r0m$].

For asymmetrical lines, you must specify the complete N-by-N resistance matrix.

Inductance per unit length (H/km) [NxN matrix] or [l1 l0 l0m] — Inductance per unit length

[0.9337e-3 4.1264e-3] (default) | matrix of N-by-N elements

Inductance l per unit length, as an N-by-N matrix in henries/km (H/km).

For a symmetrical line, you can either specify the N-by-N matrix or the sequence parameters. For a two-phase or three-phase continuously transposed line, you can enter the positive and zero-sequence inductances [$l1$ $l0$]. For a symmetrical six-phase line, you can enter the sequence parameters plus the zero-sequence mutual inductance [$l1$ $l0$ $l0m$].

For asymmetrical lines, you must specify the complete N-by-N inductance matrix.

Capacitance per unit length (F/km) [NxN matrix] or [c1 c0 c0m] — Capacitance per unit length

[12.74e-9 7.751e-9] (default) | matrix of N-by-N elements

Capacitance c per unit length, as an N-by-N matrix in farads/km (F/km).

For a symmetrical line, you can either specify the N-by-N matrix or the sequence parameters. For a two-phase or three-phase continuously transposed line, you can enter the positive and zero-sequence capacitances [$c1$ $c0$]. For a symmetrical six-phase line you can enter the sequence parameters plus the zero-sequence mutual capacitance [$c1$ $c0$ $c0m$].

For asymmetrical lines, you must specify the complete N-by-N capacitance matrix.

Note The powergui block provides the **RLC Line Parameters** tool, which calculates resistance, inductance, and capacitance per unit of length based on the line geometry and the conductor characteristics.

Measurements — Block measurements

None (default) | Phase-to-ground voltages

Select **Phase-to-ground voltages** to measure the sending end and receiving end voltages for each phase of the line model. Default is **None**.

Place a Multimeter block in your model to display the selected measurements during the simulation.

In the **Available Measurements** list box of the Multimeter block, the measurement is identified by a label followed by the block name:

Measurement	Label
Phase-to-ground voltages, sending end	Us_ph1_gnd:
Phase-to-ground voltages, receiving end	Ur_ph1_gnd:

Version History

Introduced before R2006a

References

[1] Dommel, H., "Digital Computer Solution of Electromagnetic Transients in Single and Multiple Networks," *IEEE Transactions on Power Apparatus and Systems*, Vol. PAS-88, No. 4, April, 1969.

Extended Capabilities

C/C++ Code Generation

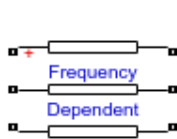
Generate C and C++ code using Simulink® Coder™.

See Also

Distributed Parameters Line (Frequency-Dependent) | PI Section Line | Power Line Parameters | power_lineparam

Distributed Parameters Line (Frequency-Dependent)

Distributed parameters line or cable model with frequency-dependent parameters



Libraries:

Simscape / Electrical / Specialized Power Systems / Power Grid Elements

Description

The Distributed Parameters Line (Frequency-Dependent) block implements a multiphase distributed parameter line or cable model with high-fidelity frequency-dependent parameters. The block performs accurate transient response simulations for a wide frequency range (typically from DC to 100 kHz or more).

The geometry of the line or cable and the frequency-dependent model parameters are defined in a MAT file that is associated with the block. The MAT file must be edited with the Power Line Parameters app or with the Power Cable Parameters app, which computes the RLC line or cable parameters for a defined range of frequencies and applies a fitting algorithm that converts the line parameters into a rational model. Rational fitting for this model is performed by using the vector fitting (VF) procedure. For more information on the phase domain line model and the state-space analysis, see [1].

Ports

Conserving

Cons 1 — Electric terminal
specialized electrical

Specialized electrical conserving port associated with terminal 1 on the sending end of the line.

Cons 2 — Electric terminal
specialized electrical

Specialized electrical conserving port associated with terminal 2 on the sending end of the line.

Cons 3 — Electric terminal
specialized electrical

Specialized electrical conserving port associated with terminal 3 on the sending end of the line.

Cons 4 — Electric terminal
specialized electrical

Specialized electrical conserving port associated with terminal 1 on the receiving end of the line.

Cons 5 — Electric terminal
specialized electrical

Specialized electrical conserving port associated with terminal 2 on the receiving end of the line.

Cons 6 — Electric terminal
specialized electrical

Specialized electrical conserving port associated with terminal 3 on the receiving end of the line.

Parameters

Edit geometry parameters — Opens MAT file
(default)

Opens the Power Line Parameters app for the specified MAT file that contains geometrical parameters of an overhead line.

Opens the Power Cable Parameters app for the specified MAT file that contains geometrical parameters of a cable arrangement.

File name — Name of MAT file
Line_735kV.mat (default) | string

Name of a MAT file that defines the geometry of the line or cable and the frequency-dependent model parameters. If the MAT file is not on the MATLAB path, specify the full path name of the file.

Browse — Browse for file
(default)

Select a MAT file from the directory browser.

Line length (km) — Line or cable length
100 (default) | positive scalar

This read-only parameter is the value of the `DATA.LineLength` variable saved in the specified MAT file. This parameter is the length of the line, in km.

Number of phases — Number of phases defined for line or cable
3 (default) | positive integer

This read-only parameter is the value of the `DATA.Phases` variable saved in the specified MAT file. This parameter is the number of phases defined for the line.

Frequency range — Frequency range of line or cable parameters
1.00e-02 Hz to 1.00e+05 Hz, 141 points. (default)

This read-only parameter is the value of the `DATA.FrequencyRange` variable saved in the specified MAT file. This parameter is a detailed description of the frequency range of the internal model.

Comments — Complementary information
example of a 735 KV three-phase line... (default)

This read-only parameter displays the contents of the `GUI.Comments` variable saved in the specified MAT file. This text provides complementary information about the line geometry and the type of conductors.

Measurements — Measure voltages

None (default) | Phase-to-ground voltages

Select **None** to make no measurements.

Select **Phase-to-ground voltages** to measure the sending end and receiving end voltages for each phase of the line model.

Place a Multimeter block in your model to display the selected measurements during simulation.

In the **Available Measurements** list box of the Multimeter block, the measurement is identified by a label followed by the block name:

Measurement	Label
Phase-to-ground voltages, sending end	Us_ph1_gnd
Phase-to-ground voltages, receiving end	Ur_ph1_gnd

Version History

Introduced in R2020b

References

- [1] Ramos-Leanos, O., and R. Iracheta. "Wide-Band Line Model Implementation in Matlab for EMT Analysis." Paper presented at the IEEE North American Power Symposium (NAPS), Arlington, TX, 26-28 Sept. 2010. <https://ieeexplore.ieee.org/document/5618959>
- [2] Ramos-Leanos, O. "Wideband Line/Cable Models for Real-Time and Off-Line Simulations of Electromagnetic Transients." Paper presented at Diss. École Polytechnique de Montréal, April 2013. https://publications.polymtl.ca/1134/1/2013_OctavioRamosLeanos.pdf
- [3] Ramos-Leanos, Octavio, José Luis Naredo, Jean Mahseredjian, Christian Dufour, José Alberto Gutierrez-Robles, and Ilhan Kocar. "A Wideband Line/Cable Model for Real-Time Simulations of Power System Transients." *IEEE Transactions on Power Delivery* 27, no. 4 (October 2012): 2211–18. <https://ieeexplore.ieee.org/document/6263255>
- [4] Iracheta, R., and O. Ramos-Leanos. "Improving computational efficiency of FD line model for real-time simulation of EMTS." Paper presented at the IEEE North American Power Symposium (NAPS), Arlington, TX, 26-28 Sept. 2010. <https://ieeexplore.ieee.org/document/5618953>

Extended Capabilities

C/C++ Code Generation

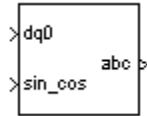
Generate C and C++ code using Simulink® Coder™.

See Also

Distributed Parameters Line | Power Line Parameters | `power_lineparam`

dq0_to_abc Transformation

Perform Park transformation from dq0 reference frame to abc reference frame



Library

powerlib_extras/Measurements, powerlib_extras/Discrete Measurements

Note The Transformations section of the Control and Measurements library contains the dq0 to abc to block. This is an improved version of the dq0_to_abc Transformation block. The new block features a mechanism that eliminates duplicate continuous and discrete versions of the same block by basing the block configuration on the simulation mode. If your legacy models contain the dq0_to_abc Transformation block, they continue to work. However, for best performance, use the dq0 to abc block in your new models.

Description

The dq0_to_abc Transformation block performs the reverse of the so-called Park transformation, which is commonly used in three-phase electric machine models. It transforms three quantities (direct axis, quadratic axis, and zero-sequence components) expressed in a two-axis reference frame back to phase quantities. The following transformation is used:

$$\begin{aligned} V_a &= V_d \sin(\omega t) + V_q \cos(\omega t) + V_0 \\ V_b &= V_d \sin(\omega t - 2\pi/3) + V_q \cos(\omega t - 2\pi/3) + V_0 \\ V_c &= V_d \sin(\omega t + 2\pi/3) + V_q \cos(\omega t + 2\pi/3) + V_0, \end{aligned}$$

where ω = rotation speed (rad/s) of the rotating frame.

The transformation is the same for the case of a three-phase current; you simply replace the V_a , V_b , V_c , V_d , V_q , and V_0 variables with the I_a , I_b , I_c , I_d , I_q , and I_0 variables.

The dq0_to_abc Transformation block is used in the model of the Synchronous Machine block where the stator quantities are referred to the rotor. The Park transformation then eliminates time-varying inductances by referring the stator and rotor quantities to a fixed or rotating reference frame. The I_d and I_q currents represent the two DC currents flowing in the two equivalent rotor windings (d winding on the same axis as the field winding, and q winding in quadratic) producing the same flux as the stator I_a , I_b , and I_c currents.

Inputs and Outputs

dq0

Connect to the first input a vectorized signal containing the sequence components [d q 0] to be converted.

`sin_cos`

Connect to the second input a vectorized signal containing the $[\sin(\omega t) \cos(\omega t)]$ values, where ω is the rotation speed of the reference frame.

`abc`

The output is a vectorized signal containing the three-phase sinusoidal quantities [phase A phase B phase C], in the same units as the dq0 input signal.

Examples

See the example of the `abc_to_dq0` Transformation block for an example using the dq0 to abc block (the improved version of the `dq0_to_abc` Transformation block).

References

[1] Krause, P. C. *Analysis of Electric Machinery*. New York: McGraw-Hill, 1994, p.135.

Version History

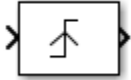
Introduced before R2006a

See Also

abc to dq0, dq0 to abc | `abc_to_dq0` Transformation

Edge Detector

Detect change in logical signal state



Library

Simscape / Electrical / Specialized Power Systems / Control

Description

The Edge Detector block outputs a needle impulse when a change in the logical input is detected.

The Edge Detector block can be programmed to detect a rising edge (when the input goes from false to true), a falling edge (when the input goes from true to false), or either edge (any change in input is detected).

Parameters

Edge detection

Select the detection criteria triggering the output impulse. Choices are **Rising** (default), **Falling**, or **Either**.

Initial condition of previous input

Specify the initial value of the previous input. The Edge Detector block can detect an edge at $t = 0$ if the input is different from the specified value. Default is 0.

Sample time (-1 for inherited)

Specify the sample time of the block, in seconds. Set to 0 to implement a continuous block. Default is 0.

Inputs and Outputs

The input signal must be a Boolean signal. The output is Boolean.

Characteristics

Direct Feedthrough	Yes
Sample Time	Specified in the Sample Time parameter
Scalar Expansion	Yes
Dimensionalized	Yes
Zero-Crossing Detection	Yes

Examples

The `power_Logic` example shows the operation of the Edge Detector block.

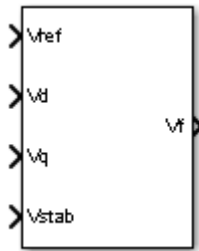
The model sample time is parameterized with the variable `Ts` (default value `Ts = 50e-3`). To simulate a continuous Edge Detector block, specify `Ts = 0` in the MATLAB Command Window.

Version History

Introduced in R2013a

Excitation System

Provide excitation system for synchronous machine and regulate its terminal voltage in generating mode

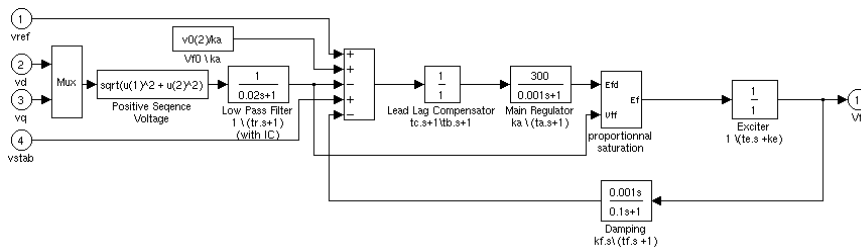


Libraries:

Simscape / Electrical / Specialized Power Systems / Electrical Machines / Synchronous Machine Control

Description

The Excitation System block is a Simulink system implementing a DC exciter described in [1], without the exciter's saturation function. The basic elements that form the Excitation System block are the voltage regulator and the exciter.



The exciter is represented by the following transfer function between the exciter voltage V_{fd} and the regulator's output ef :

$$\frac{V_{fd}}{ef} = \frac{1}{Ke + sTe}.$$

Ports

Input

vref — Stator terminal voltage
scalar

Desired value, in pu, of the stator terminal voltage.

vd — v_d component of terminal voltage
scalar

v_d component, in pu, of the terminal voltage.

vq — v_q component of terminal voltage
scalar

v_q component, in pu, of the terminal voltage.

vstab — Additional stabilization of power system oscillations
scalar

Connect this input to a power system stabilizer to provide additional stabilization of power system oscillations.

Output

Vf — Field voltage
scalar

Field voltage, in pu, for the Synchronous Machine block.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Low-pass filter time constant Tr(s): — Time constant Tr of first-order system
20e-3 (default)

Time constant Tr, in seconds (s), of the first-order system that represents the stator terminal voltage transducer.

Regulator gain and time constant [Ka() Ta(s)] — Gain Ka and time constant Ta of first-order system
[300, 0.001] (default)

Gain Ka and time constant Ta, in seconds (s), of the first-order system representing the main regulator.

Exciter [Ke() Te(s)] — Gain Ke and time constant Te of first-order system
[1, 0] (default)

Gain Ke and time constant Te, in seconds (s), of the first-order system representing the exciter.

Transient gain reduction [Tb(s) Tc(s)] — Time constants Tb and Tc of first-order system
[0, 0] (default)

Time constants Tb, in seconds (s), and Tc, in seconds (s), of the first-order system representing a lead-lag compensator.

Damping filter gain and time constant [Kf() Tf(s)] — Gain Kf and time constant Tf of first-order system
[0.001, 0.1] (default)

Gain Kf and time constant Tf, in seconds (s), of the first-order system representing a derivative feedback.

Regulator output limits and gain [Efmin, Efmax (pu), Kp()] — Regulator output limits and gain
[-11.5, 11.5, 0] (default)

Limits Efmin and Efmax are imposed on the output of the voltage regulator. The upper limit can be constant and equal to Efmax, or variable and equal to the rectified stator terminal voltage Vtf times a proportional gain Kp. If Kp is set to 0, the former applies. If Kp is set to a positive value, the latter applies.

Initial values of terminal voltage and field voltage [Vt0 (pu) Vf0(pu)] — Initial values of terminal voltage and field voltage
[1.0 1.28] (default)

Initial values of terminal voltage Vt0 (pu) and field voltage Vf0 (pu). When set correctly, they allow you to start the simulation in steady state. Initial terminal voltage should normally be set to 1 pu. Both Vt0 and Vf0 values are automatically updated by the load flow utility of the Powergui block.

Version History

Introduced before R2006a

References

- [1] "Recommended Practice for Excitation System Models for Power System Stability Studies," *IEEE Standard 421.5-1992*, August, 1992.

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

Generic Power System Stabilizer | Hydraulic Turbine and Governor | Multiband Power System Stabilizer | Steam Turbine and Governor | Synchronous Machine SI Fundamental | Synchronous Machine pu Fundamental | Synchronous Machine pu Standard

First-Order Filter

Implement first-order filter



Libraries:
Simscape / Electrical / Specialized Power Systems / Control / Filters

Description

The First-Order Filter block implements one of the following transfer functions based on the **Filter type** selected in the block parameters window.

Low-pass filter:

$$H(s) = \frac{1}{1 + Ts}$$

High-pass filter:

$$H(s) = \frac{Ts}{1 + Ts}$$

s = Laplace operator

T = time constant

Characteristics

The key characteristics of the First-Order Filter block are:

- The input accepts a vectorized input of N signals and implements N filters. This feature is particularly useful for designing controllers in three-phase systems ($N = 3$).
- You can initialize filter states for specified DC and AC inputs.
- The block enables you to compute and plot filter response.

Direct Feedthrough	Yes
Sample Time	Specified in the Sample Time parameter Continuous if Sample Time = 0
Scalar Expansion	Yes, of the parameters
States	One state per filter
Dimensionalized	Yes

Examples

Use First-Order Filter Block with Lowpass and Highpass Filters

The `power_FirstOrderFilter` example shows various uses of the First-Order Filter block using two **Filter type** parameter settings (Lowpass and Highpass).

The model sample time is parameterized with variable T_s (default value $T_s = 50e-6$). To simulate continuous filters, specify $T_s = 0$ at the MATLAB command prompt before you start.

Ports

Input

Port_1 — Filter input

scalar | vector

First-order filter input signal, specified as a scalar or a vector.

Output

Port_1 — Filter output

scalar | vector

First-order filter output signal, returned as a scalar or vector.

Parameters

Filter type — Low-pass or high-pass filter

Lowpass (default) | Highpass

Determine the type of first-order filter. Set to Lowpass to implement a low-pass filter, set to Highpass to implement a high-pass filter.

Time constant (s) — Filter time constant

$10e-3$ (default) | positive scalar

First-order filter time constant, specified in seconds.

Sample time — Sample time of block

0 (default) | nonnegative scalar

Sample time of block, specified in seconds. Set to 0 to implement a continuous block.

Initialize filter states — AC and DC initial input

on (default) | off

When selected, the filter states are initialized according to the **AC initial input: [Mag, Phase (degrees), Freq (Hz)]** and **DC initial input** parameters.

AC initial input: [Mag, Phase (degrees), Freq (Hz)] — Magnitude of initial AC component

[0, 0, 60] (default) | three-element vector | N -by-3 matrix

Specify the magnitude of the initial AC component of the input signal, its phase, in degrees, and its frequency, in hertz. When the input is vectorized (N signals), specify an N -by-3 matrix, where each row of the matrix corresponds to a particular input. Phase can be any value. Mag and Freq must be positive.

Dependencies

To enable this parameter, select **Initialize filter states**.

DC initial input — Initial DC component of input signal

0 (default) | scalar | 1-by- N vector

Specify the value of the initial DC component of the input signal. When the input signal is vectorized, specify a 1-by- N vector, where each value corresponds to a particular input.

Dependencies

To enable this parameter, select **Initialize filter states**.

Plot filter response — Plot filter response and Bode diagram

off (default) | on

When selected, the filter step response and its Bode diagram (magnitude and phase of transfer function as a function of frequency) are plotted in a figure.

Frequency range (Hz): [Start, End, Inc.] — Frequency range to plot filter Bode diagram

[0, 200, 1] (default) | three-element vector of nonnegative scalars

The frequency range used for plotting the filter Bode diagram. Specify a vector containing the starting frequency, the end frequency, and the incremental frequency, in hertz.

Dependencies

To enable this parameter, select **Plot filter response**.

Version History

Introduced in R2013a

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

Fourier

Perform Fourier analysis of signal



Libraries:

Simscape / Electrical / Specialized Power Systems / Sensors and Measurements

Description

The Fourier block performs a Fourier analysis of the input signal over a running window of one cycle of the fundamental frequency of the signal. The Fourier block can be programmed to calculate the magnitude and phase of the DC component, the fundamental, or any harmonic component of the input signal.

Recall that a signal $f(t)$ can be expressed by a Fourier series of the form

$$f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cos(n\omega t) + b_n \sin(n\omega t)$$

where n represents the rank of the harmonics. ($n = 1$ corresponds to the fundamental component.) The magnitude and phase of the selected harmonic component are calculated by these equations:

$$|H_n| = \sqrt{a_n^2 + b_n^2}$$

$$\angle H_n = \text{atan}\left(\frac{a_n}{b_n}\right)$$

where

$$a_n = \frac{2}{T} \int_{t-T}^t f(t) \cos(n\omega t) dt$$

$$b_n = \frac{2}{T} \int_{t-T}^t f(t) \sin(n\omega t) dt$$

$$T = \frac{1}{f_1}$$

f_1 : Fundamental frequency

As this block uses a running average window, one cycle of simulation must complete before the outputs give the correct magnitude and angle. For the first cycle of simulation, the outputs are held to the values specified by the initial input parameter.

Characteristics

Sample Time	Specified in the Sample Time parameter. Continuous if Sample Time = 0.
Scalar Expansion	Yes, of the parameters.
Dimensionalized	Yes.

Examples

The `power_Fourier` model shows two applications of the Fourier measurement block. The upper part of the model shows how to use the Fourier block to compute the DC component, fundamental value, and second and third harmonic content of the same input signal. In the lower part of the model, the block computes the fundamental value (magnitude and phase) of three different inputs. For this application, specify initial values for the three output signals in the dialog box of the mask.

The model sample time is parameterized by the `Ts` variable set to a default value of 50e-6 s. Set `Ts` to 0 in the command window to simulate the model in continuous mode.

Port

Input

Input — Signal to analyze

scalar | vector

Signal to analyze. Typical input signals are voltages or currents measured by Current Measurement or Voltage Measurement blocks.

Output

|u| — Magnitude of specified harmonic component

scalar

Magnitude of the specified harmonic component, in the same units as the input signal.

∠u — Phase of specified harmonic component

scalar

Phase, in degrees, of the specified harmonic component.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Fundamental frequency (Hz) — Frequency of fundamental signal

60 (default) | scalar

Fundamental frequency, in hertz, of the input signal.

Harmonic n (0 = DC, 1 = fundamental) — Harmonic component for Fourier analysis

1 (default) | nonnegative scalar

Harmonic component for the Fourier analysis. Enter 0 to analyze the DC component. Enter 1 to analyze the fundamental frequency, or enter a number corresponding to the desired harmonic.

Initial input [Mag, Phase (degrees)] — Initial magnitude and phase of output signal

[0, 0] (default) | scalar

Initial magnitude and phase in degrees of the output signal.

Sample time — Block sample time

0 (default) | nonnegative scalar

Sample time of the block, in seconds. Set to 0 to implement a continuous block.

Version History

Introduced in R2013a

Extended Capabilities

C/C++ Code Generation

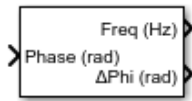
Generate C and C++ code using Simulink® Coder™.

See Also

abc_to_dq0 Transformation

Frequency (Phasor)

Measure phasor signal frequency



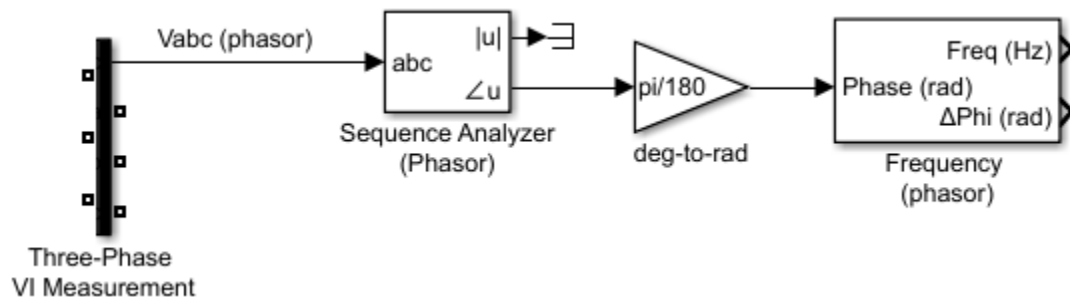
Libraries:

Simscape / Electrical / Specialized Power Systems / Sensors and Measurements

Description

The Frequency (Phasor) block measures the frequency of a phasor voltage or current by calculating the derivative of the phase angle variation of the signal, with respect to a synchronous phasor rotating at the specified nominal frequency. A low-pass filter is filtering out the high frequency components of the computed frequency. The unwrapped phase angle is computed and provided at the **ΔPhi** output.

This diagram shows an example of the Frequency (phasor) block measuring the frequency of a three-phase voltage phasor.



Use this block to measure the frequency of a phasor signal in a power system.

Ports

Input

Phase (rad) — Phase angle
scalar

Phase angle variation of the measured signal, in rad.

Data Types: `single` | `double`

Output

Freq (Hz) — Frequency
scalar

Frequency of the input signal, in Hz.

Data Types: single | double

ΔPhi (rad) — Phase angle
scalar

Unwrapped phase angle without 2π discontinuities, in rad.

Data Types: single | double

Parameters

Nominal frequency (Hz) — Fundamental frequency
60 (default) | positive scalar

Nominal frequency of the input signal, in Hz.

Time constant T for derivative (s) — Filter time constant
0.1 (default) | scalar in the range [0.01,0.1]

The low-pass filter time constant, in seconds.

Version History

Introduced in R2018a

Extended Capabilities

C/C++ Code Generation

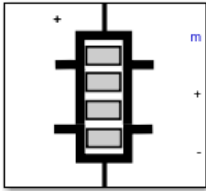
Generate C and C++ code using Simulink® Coder™.

See Also

PMU (PLL-Based, Positive-Sequence)

Fuel Cell Stack

Implement generic hydrogen fuel cell stack model



Libraries:

Simscape / Electrical / Specialized Power Systems / Sources

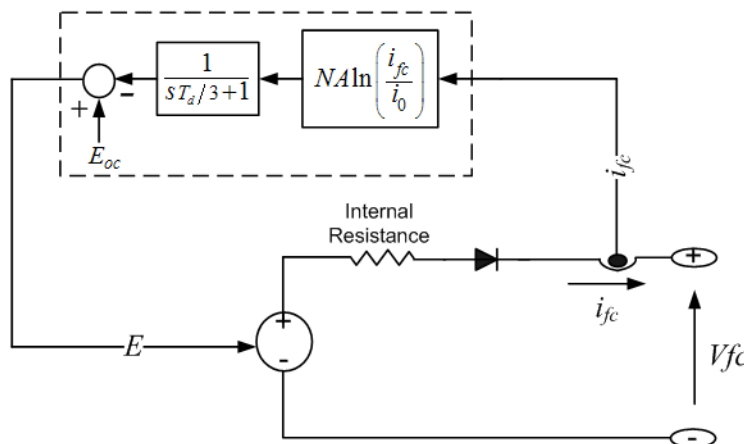
Description

The Fuel Cell Stack block implements a generic model parameterized to represent most popular types of fuel cell stacks fed with hydrogen and air.

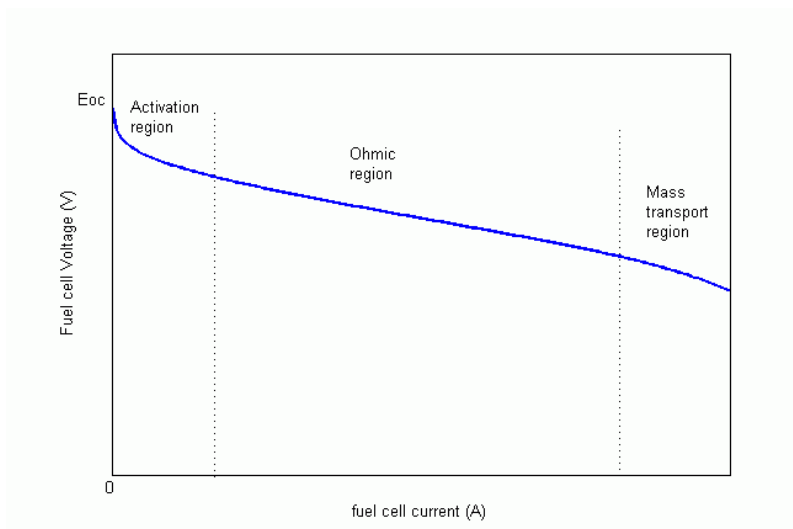
The block represents two versions of the stack model: a simplified model and a detailed model. You can switch between the two models by selecting the level in the mask under **Model detail level** in the block dialog box.

Simplified Model

This model is based on the equivalent circuit of a fuel cell stack shown below:



The simplified model represents a particular fuel cell stack operating at nominal conditions of temperature and pressure. The parameters of the equivalent circuit can be modified based on the polarization curve obtained from the manufacturer datasheet. You just have to input in the mask the value of the voltage at 0 and 1 A, the nominal and the maximum operating points, for the parameters to be calculated. A typical polarization curve consists of three regions:



The first region represents the activation voltage drop due to the slowness of the chemical reactions taking place at electrode surfaces. Depending on the temperature and operating pressure, type of electrode, and catalyst used, this region is more or less wide. The second region represents the resistive losses due to the internal resistance of the fuel cell stack. Finally, the third region represents the mass transport losses resulting from the change in concentration of reactants as the fuel is used.

Detailed Model

The detailed model represents a particular fuel cell stack when the parameters such as pressures, temperature, compositions and flow rates of fuel and air vary. You can select which parameters to vary on the **Signal variation** pane on the block dialog box. These variations affect the open circuit voltage (E_{oc}), the exchange current (i_0), and the Tafel slope (A). E_{oc} , i_0 and A are modified as follows:

$$E_{oc} = K_c E_n$$

$$i_0 = \frac{zFk(P_{H_2} + P_{O_2})\Delta v}{Rh} e^{\frac{-\Delta G}{RT}}$$

$$A = \frac{RT}{z\alpha F},$$

where:

$$R = 8.3145 \text{ J/(mol K)}$$

$$F = 96485 \text{ A s/mol}$$

z = Number of moving electrons

E_n = Nernst voltage, which is the thermodynamics voltage of the cells and depends on the temperatures and partial pressures of reactants and products inside the stack (V)

α = Charge transfer coefficient, which depends on the type of electrodes and catalysts used

P_{H_2} = Partial pressure of hydrogen inside the stack (Pa)

P_{O_2} = Partial pressure of oxygen inside the stack (Pa)

k = Boltzmann's constant = 1.38×10^{-23} J/K

h = Planck's constant = 6.626×10^{-34} J s

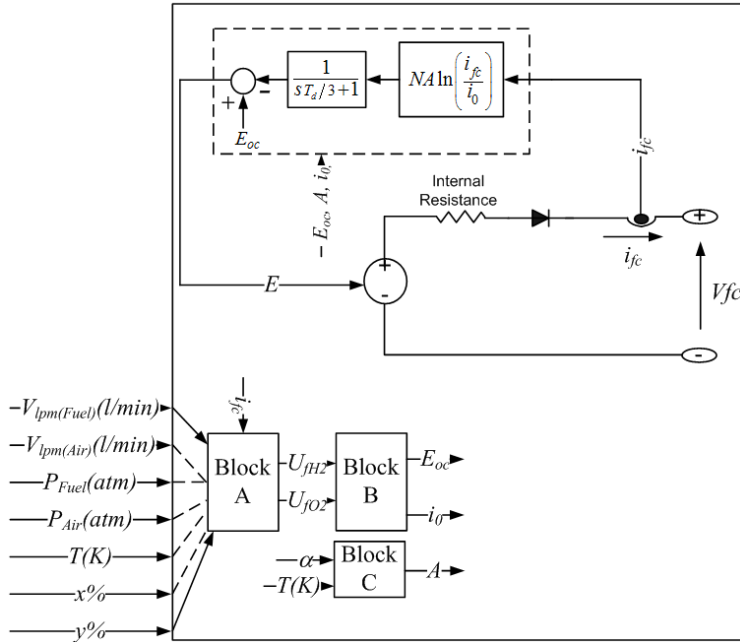
Δv = Activation barrier volume factor (m^3). The size of activation barrier (ΔG) is computed assuming $\Delta v = 1 \text{ m}^3$.

ΔG = Size of the activation barrier which depends on the type of electrode and catalyst used (J/mol)

T = Temperature of operation (K)

K_c = Voltage constant at nominal condition of operation

The equivalent circuit is the same as for the simplified model, except that the parameters E_{oc} , i_0 and A have to be updated on-line as shown below:



The rates of conversion (utilizations) of hydrogen (U_{fH_2}) and oxygen (U_{fO_2}) are determined in Block A as follows:

$$\begin{cases} U_{fH_2} = \frac{n_{H_2}^r}{n_{H_2}^{in}} = \frac{60000RTNi_{fc}}{zFP_{fuel}V_{lpm(fuel)}x\%} \\ U_{fO_2} = \frac{n_{O_2}^r}{n_{O_2}^{in}} = \frac{60000RTNi_{fc}}{2zFP_{air}V_{lpm(air)}y\%} \end{cases}$$

where

P_{fuel} = Absolute supply pressure of fuel (atm)

P_{air} = Absolute supply pressure of air (atm)

$V_{lpm(fuel)}$ = Fuel flow rate (l/min)

$V_{lpm(air)}$ = Air flow rate (l/min)

x = Percentage of hydrogen in the fuel (%)

y = Percentage of oxygen in the oxidant (%)

N = Number of cells

The 60000 constant comes from the conversion from the liter/min flow rate used in the model to m³/s (1 liter/min = 1/60000 m³/s).

The partial pressures and the Nernst voltage are determined in Block B as follows:

$$\begin{cases} P_{H_2} = (1 - U_{fH_2})x\%P_{fuel} \\ P_{H_2O} = (w + 2y\%U_{fO_2})P_{air} \\ P_{O_2} = (1 - U_{fO_2})y\%P_{air} \end{cases}$$

and

$$E_n = \begin{cases} 1.229 + (T - 298)\frac{-44.43}{zF} + \frac{RT}{zF}\ln(P_{H_2}P_{O_2}^{1/2}) & \text{when } T \leq 100^\circ C \\ 1.229 + (T - 298)\frac{-44.43}{zF} + \frac{RT}{zF}\ln\left(\frac{P_{H_2}P_{O_2}^{1/2}}{P_{H_2O}}\right) & \text{when } T > 100^\circ C \end{cases}$$

where

P_{H_2O} = Partial pressure of water vapor inside the stack (atm)

w = Percentage of water vapor in the oxidant (%)

From the partial pressures of gases and the Nernst voltage, the new values of the open circuit voltage (E_{oc}) and the exchange current (i_0) can be calculated.

Block C calculates the new value of the Tafel slope (A).

The parameters α , ΔG , and K_c are calculated based on the polarization curve at nominal conditions of operation along with some additional parameters, such as the low heating value (LHV) efficiency of the stack, composition of fuel and air, supply pressures and temperatures. They can be easily obtained from the manufacturer datasheet.

The nominal rates of conversion of gases are calculated as follows:

$$U_{fH_2} = \frac{\eta_{nom}\Delta h^0(H_2O(gas))N}{zFV_{nom}}$$

$$U_{fO_2} = \frac{60000RT_{nom}NI_{nom}}{2zFP_{air,nom}V_{lpm(air),nom} \cdot 0.21}$$

where:

η_{nom} = Nominal LHV efficiency of the stack (%).

$$\Delta h^0(H_2O(gas)) = 241.83 \times 10^3 \text{ J/mol.}$$

V_{nom} = Nominal voltage (V).

I_{nom} = Nominal current (A).

$V_{lpm(air)nom}$ = Nominal air flow rate (l/min).

P_{airnom} = Nominal absolute air supply pressure (Pa).

T_{nom} = Nominal operating temperature (K).

From these rates of conversion, the nominal partial pressures of gases and the Nernst voltage can be derived. With E_{oc} , i_0 and A known and assuming that the stack operates at constant rates of conversion or utilizations at nominal condition, α , ΔG , and K_c can be determined.

The dynamics of the fuel cell are represented if you specify the response time and the parameters for flow dynamics (peak utilization and corresponding voltage undershoot) on the **Fuel Cell Dynamics** pane on the dialog box.

The response time (T_d) @ 95% is used to model the "charge double layer" phenomenon due to the build-up of charges at electrode/electrolyte interface. This affects only the activation voltage ($NA \ln(i_{fo}/i_0)$) as shown on the equivalent circuits.

The peak utilization ($U_{fO2(peak)}$) and the corresponding voltage undershoot (V_u) are used to model the effect of oxygen depletion (due to the air compressor delay) on the cell output voltage. The Nernst voltage is modified due to this effect as follows:

$$E_n = \begin{cases} E_n - K(U_{fO2} - U_{fO2(nom)}) & U_{fO2} > U_{fO2(nom)} \\ E_n & U_{fO2} \leq U_{fO2(nom)} \end{cases}$$

where

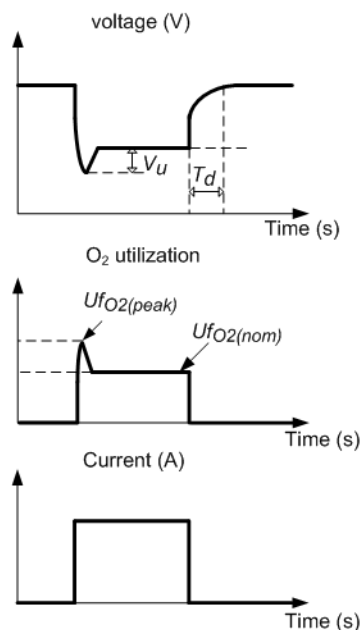
K = voltage undershoot constant

$U_{fO2(nom)}$ = nominal oxygen utilization

K is determined as follows:

$$K = \frac{V_u}{K_c(U_{fO2(peak)} - U_{fO2(nom)})}.$$

Current step and interrupt tests must be made on a real stack to represent with accuracy its dynamics. The figure below shows the stack response from these tests and the required parameters (T_d , $U_{fO2(peak)}$ and V_u).



The response time (T_d) depends on the fuel cell stack itself and is usually given on the datasheet. The parameters for flow dynamics ($U_{fO2(peak)}$ and V_u) depend on the dynamics of external equipment (compressor, regulator, and loads) and they are not provided by manufacturers as their values vary with the user application. For simulation, assume values of $U_{fO2(peak)}$ between 60% to 70% and V_u between 2-5% of the stack nominal voltage.

Assumptions and Limitations

These are the assumptions of this block:

- The gases are ideal.
- The stack is fed with hydrogen and air.
- The stack is equipped with a cooling system which maintains the temperature at the cathode and anode exits stable and equal to the stack temperature.
- The stack is equipped with a water management system to maintain the humidity inside the cell at appropriate level at any load.
- The cell voltage drops are due to reaction kinetics and charge transport as most fuel cells do not operate in the mass transport region.
- Pressure drops across flow channels are negligible.
- The cell resistance is constant at any condition of operation.

These are the limitations of this block:

- Chemical reaction dynamics caused by partial pressure changes of chemical species inside the cell are not considered.
- The stack output power is limited by the fuel and air flow rates supplied.
- The effect of temperature and humidity of the membrane on the internal resistance is not considered.

- The flow of gases or water through the membrane is not considered.

Ports

Output

m — Measurement vector
Simulink signal

The Simulink output of the block is a vector containing 11 signals. You can demultiplex these signals by using the Bus Selector block provided in the Simulink library.

Signal	Definition	Units	Symbol	Model detail level (Signal availability)	
				Detailed	Simplified
1	Voltage	V	V_{fc}	Yes	Yes
2	Current	I	I_{fc}	Yes	Yes
3	Stack Efficiency	%	η	Yes	No(Set to 0)
4	Stack consumption [Air, Fuel]	slpm	V_{slpm}	Yes	No(Set to 0)
5	Flow Rate [Air, Fuel]	lpm	Fr_{lpm}	Yes	No(Set to 0)
6	Stack consumption [Air, Fuel]	lpm	V_{lpm}	Yes	No(Set to 0)
7	Utilization [Oxygen, Hydrogen]	%	U_f	Yes	No(Set to 0)
8	Slope of Tafel curve		A	Yes	No(Set to 0)
9	Exchange current	A	i_0	Yes	No(Set to 0)
10	Nernst voltage	V	E_n	Yes	No(Set to 0)
11	Open circuit voltage	V	E_{oc}	Yes	No(Set to 0)

Conserving

+ — Positive terminal
specialized electrical

Specialized electrical conserving port associated with the positive terminal.

- — Negative terminal
specialized electrical

Specialized electrical conserving port associated with the negative terminal.

Parameters

Parameters

Preset model — Set of predetermined polarization curves

No (User-Defined) (default) | PEMFC - 1.26 kW - 24 Vdc | PEMFC - 6 kW - 45 Vdc |
PEMFC - 50 kW - 625 Vdc | AFC - 2.4 kW - 48 Vdc | SOFC - 3 kW - 100 Vdc | SOFC -
25 kW - 630 Vdc

Provides a set of predetermined polarization curves and parameters for particular fuel cell stacks found on the market:

- No (User-Defined) (default)
- PEMFC - 1.26 kW - 24 Vdc
- PEMFC - 6 kW - 45 Vdc
- PEMFC - 50 kW - 625 Vdc
- AFC - 2.4 kW - 48 Vdc
- SOFC - 3 kW - 100 Vdc
- SOFC - 25 kW - 630 Vdc

Select one of these preset models to load the corresponding parameters in the entries of the dialog box. Select No (User-Defined) if you do not want to use a preset model.

Model detail Level — Detail level

Simplified (default) | Detailed

Provide access to the two versions of the model:

- Simplified
- Detailed (default)

When a simplified model is used, there is no variable under the **signal variation** tab.

Voltage at 0A and 1A [V_0(V), V_1(V)] — Voltage at 0A and 1A

[65,63] (default)

Voltage at 0 A and 1 A of the stack (Volts). Assuming nominal and constant gases utilizations. Default is [65 63].

Nominal operating point [Inom(A), Vnom(V)] — Nominal operating point

[133.3,45] (default)

Rated current (Ampere) and rated voltage (Volts) of the stack. Assuming nominal and constant gases utilizations. Default is [133.3 45].

Maximum operating point [Iend(A), Vend(V)] — Maximum operating point

[225,37] (default)

Current (Ampere) and voltage (Volts) of the stack at maximum power. Assuming nominal and constant gases utilizations. Default is [225 37].

Number of cells — Number of cells

65 (default)

Number of cells in series in the stack. This parameter is available only for a detailed model. Default is 65.

Nominal stack efficiency (%) — Nominal stack efficiency

55 (default)

Rated efficiency of the stack relative to the low heating value (LHV) of water. This parameter is available only for a detailed model. Default is 55.

Operating temperature (Celsius) — Operating temperature
65 (default)

Nominal temperature of operation in degrees Celsius. This parameter is available only for a detailed model. Default is 65.

Nominal Air flow rate (lpm) — Nominal Air flow rate
300 (default)

Rated air flow rate (l/min). This parameter is available only for a detailed model. Default is 300.

Nominal supply pressure [Fuel (atm), Air (atm)] — Nominal supply pressure
[1.5, 1] (default)

Rated supply pressure (absolute) of fuel and air in atm. This parameter is available only for a detailed model. Default is [1.5 1].

Nominal composition (%) [H2 O2 H2O(Air)] — Nominal composition
[99.95, 21, 1] (default)

Rated percentage of hydrogen (x) in the fuel, oxygen (y) and water (w) in the oxidant. This parameter is available only for a detailed model. Default is [99.95 21 1].

Peak O2 utilization (%) — Peak O2 utilization
80 (default)

Enter the peak oxygen utilization at nominal condition of operation. Default is 80. This parameter is available only when the **Specify Fuel Cell Dynamics?** check box is selected and the **Air flow rate** check box is selected in the **Signal Variation** tab.

Voltage undershoot (V) @ peak O2 utilization — Voltage undershoot
10 (default)

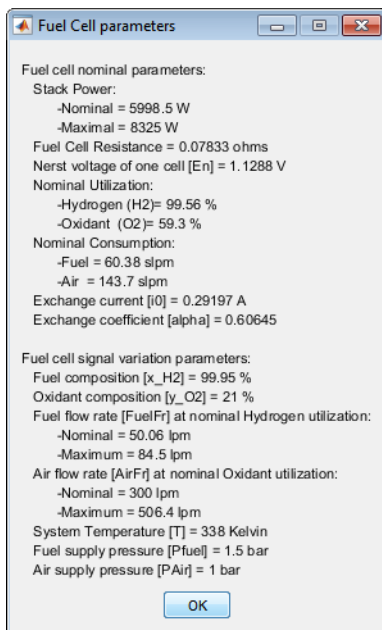
Enter the voltage undershoot (Volts) at peak oxygen utilization at nominal condition of operation. Default is 10. This parameter is available only when the **Specify Fuel Cell Dynamics?** check box is selected and the **Air flow rate** check box is selected in the **Signal Variation** tab.

Plot V-I characteristic — Plot V-I characteristic

Plots a figure containing two graphs. The first graph represents the stack voltage (Volts) vs current (A) and the second graph represents the stack power (kW) vs current (A). This button is available only for a detailed model.

View cell parameters — View cell parameters

Presents the overall parameters of the stack. This button is available only for a detailed model. The dialog box is shown below.



Signal variation

The **Signal Variation** tab is available only if **Model detail level** is set to **Detailed**. It provides a list of parameters that can be varied. Select a check box for a variable to input a corresponding signal to the block. The following signals can be input to the block:

Fuel composition [x_H2(%)] — Hydrogen in fuel

off (default) | on

Percentage of hydrogen in the fuel. Default is cleared.

Oxidant composition [y_O2(%)] — Oxygen in oxidant

off (default) | on

Percentage of oxygen in the oxidant. Default is cleared.

Fuel flow rate [FuelFr(lpm)] — Fuel flow rate

off (default) | on

Fuel flow rate in liter per minutes. Default is cleared.

Air flow rate [AirFr(lpm)] — Air flow rate

off (default) | on

Air flow rate in liter per minutes. Default is cleared.

System Temperature [T(Kelvin)] — Temperature of operation

off (default) | on

Temperature of operation in Kelvin. Default is cleared.

Fuel supply pressure [Pfuel(atm)] — Fuel supply pressure

off (default) | on

Fuel supply pressure in atm. Default is cleared.

Air supply pressure [PAir(atm)] — Air supply pressure
off (default) | on

Air supply pressure in atm. Default is cleared.

Fuel Cell Dynamics

Specify Fuel Cell Dynamics? — Fuel Cell dynamics options
off (default) | on

Whether you want to specify the fuel cell dynamics. Select the check box to enter the fuel cell response time in seconds. Default is cleared.

Fuel Cell response time (sec) — Fuel Cell response time
1 (default)

Enter the response time of the cell (at 95% of the final value). Default is 1. This parameter becomes available only when the **Specify Fuel Cell Dynamics?** check box is selected.

Version History

Introduced in R2008a

R2024b: Unit for the Nominal supply pressure [Fuel (atm), Air (atm)] parameter has changed to atm
Behavior changed in R2024b

The **Nominal supply pressure [Fuel (atm), Air (atm)]** parameter now only accepts values in atm instead of bar.

R2024b: Calculation of air flow rate has changed when there is no signal variation
Behavior changed in R2024b

If, in the **Signal Variation** settings, you clear the **Fuel flow rate [FuelFr(Ipm)]** and **Air flow rate [AirFr(Ipm)]** parameters, the block now obtains the air flow rate directly from the value of the **Nominal Air flow rate (Ipm)** parameter and calculates the fuel flow rate by rearranging the equation for hydrogen utilization by making flow rate the subject, based on the nominal conditions.

References

- [1] Njoya, S. M., O. Tremblay, and L. -A. Dessaint. *A generic fuel cell model for the simulation of fuel cell vehicles*. Vehicle Power and Propulsion Conference, 2009, VPPC '09, *IEEE* . Sept. 7–10, 2009, pp. 1722–29.
- [2] Motapon, S.N., O. Tremblay, and L. -A. Dessaint. "Development of a generic fuel cell model: application to a fuel cell vehicle simulation." *Int. J. of Power Electronics*. Vol. 4, No. 6, 2012, pp. 505–22.

Extended Capabilities

C/C++ Code Generation

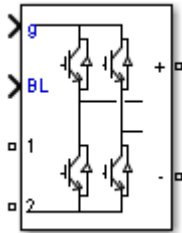
Generate C and C++ code using Simulink® Coder™.

See Also

Battery | PV Array

Full-Bridge Converter

Implement full-bridge power converter



Libraries:

Simscape / Electrical / Specialized Power Systems / Power Electronics

Description

The Full-Bridge Converter block implements a full-bridge power converter. You can choose from four model types:

- **Switching devices** — The converter is modeled with IGBT/diode pairs controlled by firing pulses produced by a PWM generator. This model provides the most accurate simulation results.
- **Switching function** — The converter is modeled by a switching-function model. The switches are replaced with two voltage sources and two diodes on the AC side and with two current sources on the DC side.

The converter is controlled by firing pulses produced by a PWM generator (0/1 signals) or by firing pulses averaged over a specified period (PWM averaging: signals between 0 and 1). Both modes of operation produce harmonics normally generated by a PWM-controlled converter and also correctly simulate rectifying operation as well as blanking time. This model type is well-suited for real-time simulation.

- **Average model (Uref-controlled)** — The converter is modeled using a switching-function model directly controlled by the reference voltage signals (Uref). A PWM generator is not required.
- **Average model (Uref-controlled, no rectifier mode)** — The block uses the voltage source directly controlled by the reference voltage to model the converter. The model does not require a PWM generator and does not simulate the rectifier mode. This setting provides the fastest simulations.

When doing real-time simulation, use the **Switching function** setting with the firing pulses averaged, or the **Average model (Uref-controlled)** or **Average model (Uref-controlled, no rectifier mode)** settings.

Ports

Input

g — Gating signal to control converter
vector of four elements

Vectorized gating signal to control the converter. The gating signal contains four firing pulses to control Q1, Q2, Q3, and Q4.

Dependencies

To enable this port, set **Model type** to `Switching devices` or `Switching function`.

Uref — Reference voltage signal to control converter
vector of three elements

Reference voltage signal to control the converter.

Dependencies

To enable this port, set **Model type** to `Average model (Uref-controlled)` or `Average model (No rectifier mode)`.

BL — Input to block firing pulses
scalar

Input port to block all firing pulses to the converter. To block all firing pulses to the converter, apply a signal value of 1 to this port.

Conserving

+ — Positive terminal
specialized electrical

Specialized electrical conserving port associated with the positive terminal.

- — Negative terminal
specialized electrical

Specialized electrical conserving port associated with the negative terminal.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Model type — Model type
`Switching devices (default)` | `Switching function` | `Average model (Uref-controlled)` | `Average model (No rectifier mode)`

Select one of these values:

- **Switching devices** — The converter is modeled with IGBT/diode pairs controlled by firing pulses produced by a PWM generator.
- **Switching function** — The converter is modeled by a switching-function model. The switches are replaced with two voltage sources and two diodes on the AC side and with two current sources on the DC side.
- **Average model (Uref-controlled)** — The converter is modeled using a switching-function model directly controlled by the reference voltage signals (Uref).

- **Average model (No rectifier mode)** — The block uses the voltage source directly controlled by the reference voltage to model the converter. The model does not require a PWM generator and does not simulate the rectifier mode.

Device on-state resistance (Ohms) — Internal resistance of switching devices

1e-3 (default) | positive scalar

Internal resistance of the switching devices, in ohms.

Dependencies

To enable this parameter, set **Model type** to Switching devices.

Snubber resistance (Ohms) — Snubber resistance

1e6 (default) | positive scalar

Snubber resistance, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to Switching devices.

Snubber capacitance (F) — Snubber capacitance

inf (default) | nonnegative scalar

Snubber capacitance, in farads. Set the snubber capacitance to `0` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to Switching devices.

Diode on-state resistance (Ohms) — Internal resistance of diodes

1e-3 (default) | positive scalar

Internal resistance of the diodes, in ohms.

Dependencies

To enable this parameter, set **Model type** to Switching function or Average model (Uref-controlled).

Diode snubber resistance (Ohms) — Snubber resistance of diodes

1e6 (default) | positive scalar

Snubber resistance of the diodes, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to Switching function or Average model (Uref-controlled).

Diode snubber capacitance (F) — Snubber capacitance of diodes

inf (default) | nonnegative scalar

Snubber capacitance of the diodes, in farads. Set the snubber capacitance to `0` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to `Switching function` or `Average model (Uref-controlled)`.

Diode forward voltage (V) — Forward voltage across diode
`1e-3` (default) | nonnegative scalar

Forward voltage, in volts, across the diode when it is conducting.

Dependencies

To enable this parameter, set **Model type** to `Switching function` or `Average model (Uref-controlled)`.

DC side snubber resistance (Ohms) — Snubber resistance across current sources
`inf` (default) | positive scalar

Snubber resistance across the two current sources, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to `Switching function`, `Average model (Uref-controlled)`, or `Average model (No rectifier mode)`.

AC side source resistance (Ohms) — Source resistance on AC side
`1e-3` (default) | positive scalar

Source resistance on the AC side, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to `Average model (No rectifier mode)`.

Version History

Introduced in R2015b

Extended Capabilities

C/C++ Code Generation

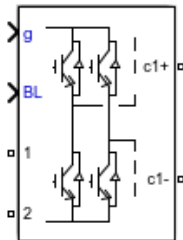
Generate C and C++ code using Simulink® Coder™.

See Also

Half-Bridge Converter | Two-Level Converter

Full-Bridge MMC (External DC Links)

Full-bridge MMC power converter with external DC connections



Libraries:

Simscape / Electrical / Specialized Power Systems / Power Electronics

Description

The Full-Bridge MMC (External DC Links) block implements a full-bridge modular multilevel converter with external DC links. The converter consists of multiple series-connected power modules. Each power module consists of one H-Bridge with external DC outputs.

You can choose from three model types:

- **Switching devices** — The converter uses IGBT/diode pairs.
- **Switching function** — The converter is based on a switching-function model.
- **Average model (Uref-controlled)** — The converter is modeled using a switching-function model directly controlled by the reference voltage signals.

Ports

Input

g — Firing pulse gate
vector

Gate signal that controls the converter. The gate signal contains the firing pulses to control four switches at each power module in the converter (four times the **Number of power modules** pulses).

Dependencies

This port is visible only when you set the **Model type** parameter to **Switching devices** or **Switching function**.

Data Types: single | double | Boolean

Uref — Reference
vector

Reference voltage signal that controls the converter. The vectorized signal contains one reference voltage for each power module in the converter.

Dependencies

This port is visible only when you set the **Model type** parameter to **Average model (Uref-controlled)**.

Data Types: `single` | `double`

BL — Firing pulse block
scalar

Input used to block all firing pulses when the signal is 1. To block all firing pulses to the converter, input a value of 1.

Data Types: `single` | `double` | `Boolean`

Output

c1+ — Positive DC terminal
scalar

Positive DC terminal of the first power module. Additional positive DC terminals, labeled **c2+**, **c3+**, **c4+**, and so on, are added to the block for the corresponding number of power modules defined by the **Number of power modules** parameter.

c1- — Negative DC terminal
scalar

Negative DC terminal of the first power module. Additional negative DC terminals, labeled **c2-**, **c3-**, **c4-**, and so on, are added to the block for the corresponding number of power modules defined by the **Number of power modules** parameter.

1 — Output terminal
vector

Output terminal 1 of the converter.

2 — Output terminal
vector

Output terminal 2 of the converter.

Parameters

Model type — Converter model
`Switching devices` | `Switching function` | `Average model (Uref-controlled)`

Specify the model type to use for the switching devices. You can choose from three model types:

- **Switching devices** — The converter uses IGBT/diode pairs. A multilevel PWM generator produces firing pulses (0/1 signals), which trigger switching in the converter.
- **Switching function** — The converter is based on a switching-function model. The model uses two voltage sources and two diodes on the AC side, and two current sources on the DC side.

The converter is controlled by firing pulses produced by a PWM generator (0/1 signals) or by firing pulses averaged over a specified period (PWM averaging: signals from 0 through 1). Both

modes of operation produce harmonics normally generated by a PWM-controlled converter, and also correctly simulate the rectifying operation and blanking time. This model type is suitable for real-time simulation

- **Average model (Uref-controlled)** — The converter is modeled using a switching-function model directly controlled by the reference voltage signals. A PWM generator is not required. This model provides the fastest simulations.

Dependencies

Each **Model type** option enables a particular set of parameters.

Number of power modules — Number of modules

1 (default)

Number of series-connected power modules that are in the converter.

Device on-state resistance (Ohms) — Resistance

1e-3 (default)

Internal resistance of the switching devices.

Dependencies

This parameter is enabled when the **Model type** parameter is set to **Switching devices**.

Snubber resistance (Ohms) — Resistance

1e6 (default) | inf

To eliminate the snubbers, set the snubber resistance to `inf`.

Dependencies

This parameter is enabled when the **Model type** parameter is set to **Switching devices**.

Snubber capacitance (F) — Capacitance

inf (default)

Snubber resistance of the switching device. To eliminate the snubbers, set the snubber capacitance to 0.

Dependencies

This parameter is enabled when the **Model type** parameter is set to **Switching devices**.

Diode on-state resistance (Ohms) — Resistance

1e-3 (default)

Internal resistance of the diodes,

Dependencies

This parameter is enabled when the **Model type** parameter is set to **Switching function** or **Average model (Uref-controlled)**.

Diode snubber resistance (Ohms) — Resistance

1e6 (default) | inf

Snubber resistance of the diode device. To eliminate the snubbers, set the snubber resistance to `inf`.

Dependencies

This parameter is enabled when the **Model type** parameter is set to `Switching function` or `Average model (Uref-controlled)`.

Diode snubber capacitance (F) — Capacitance

`inf` (default) | 0

Snubber capacitance of the diode. To eliminate the snubbers, set the snubber capacitance to 0.

Dependencies

This parameter is enabled when the **Model type** parameter is set to `Switching function` or `Average model (Uref-controlled)`.

Diode forward voltage (V) — Voltage

`1e-3` (default)

Forward voltage across the diode when it is conducting.

Dependencies

This parameter is enabled when the **Model type** parameter is set to `Switching function` or `Average model (Uref-controlled)`.

Current source snubber resistance (Ohms) — Resistance

`inf` (default)

The snubber resistance across the two current sources. To eliminate the snubbers, set the snubber resistance to `inf`.

Dependencies

This parameter is enabled when the **Model type** parameter is set to `Switching function` or `Average model (Uref-controlled)`.

Sample time (s) — Computation rate

`10e-6` (default)

Sample time of the block. To implement a continuous block, set to 0.

Dependencies

This parameter is enabled when the **Model type** parameter is set to `Switching function` or `Average model (Uref-controlled)`.

Version History

Introduced in R2017b

Extended Capabilities

C/C++ Code Generation

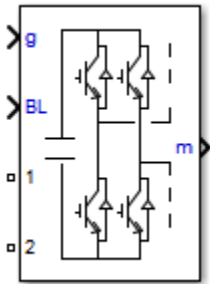
Generate C and C++ code using Simulink® Coder™.

See Also

Full-Bridge MMC | Full-Bridge Converter | Half-Bridge MMC | PWM Generator (Multilevel)

Full-Bridge MMC

Implement a full-bridge modular multilevel converter



Library

Simscape / Electrical / Specialized Power Systems / Power Electronics

Description

The Full-Bridge MMC block implements a full-bridge modular multilevel converter. The converter consists of multiple series-connected power modules. Each power module consists of one H-bridge and one capacitor on the DC side.

You can choose from three model types:

- **Switching devices** — The converter uses IGBT/diode pairs. A multilevel PWM generator produces firing pulses (0/1 signals), which trigger switching in the converter.
- **Switching function** — The converter is based on a switching-function model. The model uses two voltage sources and two diodes on the AC side, and two current sources on the DC side.

The converter is controlled by firing pulses produced by a PWM generator (0/1 signals) or by firing pulses averaged over a specified period (PWM averaging: signals from 0 through 1). Both modes of operation produce harmonics normally generated by a PWM-controlled converter, and also correctly simulate the rectifying operation and blanking time. This model type is suitable for real-time simulation.

- **Average model (Uref-controlled)** — The converter is modeled using a switching-function model directly controlled by the reference voltage signals. A PWM generator is not required. This model provides the fastest simulations.

Parameters

Model type

Specify the model type to use:

- Switching devices (default)
- Switching function

- Average model (Uref-controlled)

Number of power modules

Specify the number of series-connected power modules that are in the converter. The default value is 1.

Capacitor value (F)

Specify the capacitance, in farads, of the capacitors connected on the DC side of each power module. You can specify a single value that sets all the capacitances to the same value, or specify a vector containing different capacitance values for every power module. The default value is $10\text{e-}3$.

Capacitor initial voltage (V)

Specify the capacitor initial voltage, in volts, of the capacitors connected on the DC side of each power module. You can specify a single value that sets all the capacitor initial voltage to the same value, or specify a vector containing different initial voltage for every power module. The default value is 1000.

Device on-state resistance (Ohms)

Internal resistance of the switching devices, in ohms. This parameter is available only when you set the **Model type** parameter to **Switching devices**. The default value is $1\text{e-}3$.

Snubber resistance (Ohms)

The snubber resistance, in ohms. To eliminate the snubbers, set the snubber resistance to **inf**. This parameter is available only when you set the **Model type** parameter to **Switching devices**. The default value is $1\text{e}6$.

Snubber capacitance (F)

The snubber capacitance, in farads. To eliminate the snubbers, set the snubber capacitance to 0. This parameter is available only when you set the **Model type** parameter to **Switching devices**. The default value is **inf**.

Diode on-state resistance (Ohms)

Internal resistance of the diodes, in ohms. This parameter is available only when you set the **Model type** parameter to **Switching function** or **Average model (Uref-controlled)**. The default value is $1\text{e-}3$.

Diode snubber resistance (Ohms)

The snubber resistance, in ohms. To eliminate the snubbers, set the snubber resistance to **inf**. This parameter is available only when you set the **Model type** parameter to **Switching function** or **Average model (Uref-controlled)**. The default value is $1\text{e}6$.

Diode snubber capacitance (F)

The snubber capacitance in farads. To eliminate the snubbers, set the snubber capacitance to 0. This parameter is available only when you set the **Model type** parameter to **Switching function** or **Average model (Uref-controlled)**. The default value is **inf**.

Diode forward voltage (V)

Forward voltage, in volts, across the diode when it is conducting. This parameter is available only when you set the **Model type** parameter to **Switching function** or **Average model (Uref-controlled)**. The default value is $1\text{e-}3$.

Sample time (s)

Sample time of the block. To implement a continuous block, set to 0.

This parameter is available only when you set the **Model type** parameter to **Switching function** or **Average model (Uref-controlled)**. The default value is $10e-6$.

Inputs and Outputs

g

Vectorized gating signal that controls the converter. The gating signal contains the firing pulses to control four switches at each power module in the converter (four times the **Number of power modules** pulses). This port is visible only when you set the **Model type** parameter to **Switching function** or **Switching function**.

Uref

Vectorized reference voltage signal that controls the converter. The vectorized signal contains one reference voltage for each power module in the converter. This port is visible only when you set the **Model type** parameter to **Average model (Uref-controlled)**.

BL

You can block all firing pulses to the converter by applying a scalar signal value of 1 at the BL input. It is not possible to block an individual module. When the input signal is 1, all modules in the MMC are blocked.

1

Output terminal 1 of the converter.

2

Output terminal 2 of the converter.

m

Measurement signal containing the voltage of capacitors connected on the DC side of each power module.

Version History

Introduced in R2016b

See Also

Full-Bridge Converter | Half-Bridge MMC | PWM Generator (Multilevel)

Topics

"Switching Function Converter Controlled by Averaged Firing Pulses"

"MMC-STATCOM with 22 Power Modules per Phase"

Fundamental (PLL-Driven)

Compute fundamental value of signal



Library

Simscape / Electrical / Specialized Power Systems / Sensors and Measurements

Description

The Fundamental (PLL-Driven) block computes the fundamental value of the input 3 over a running window of one cycle of fundamental frequency given by input 1. The reference frame required for the computation is given by the input 2.

Based on the Fourier analysis of a periodic signal, the fundamental value of a signal $f(t)$ can be expressed as

$$\text{Fundamental}(f(t)) = a\cos(\omega_0 t) + b\sin(\omega_0 t)$$

$$a = \frac{2}{T} \int_{(t-T)}^t f(t)\cos(\omega_0 t)dt$$

$$b = \frac{2}{T} \int_{(t-T)}^t f(t)\sin(\omega_0 t)dt$$

$$T = \frac{1}{f_0} \quad f_0: \text{Fundamental frequency}$$

$$\omega_0 = 2 \times \pi \times f_0$$

The magnitude and phase of the fundamental is calculated by

$$\text{Magnitude} = \sqrt{a^2 + b^2} \quad \text{Phase} = \text{atan}(b/a)$$

To resolve these equations, the block uses the input 1 (Freq) for f_0 and input 2 (wt) for $\omega_0 t$. These two input signals are normally connected to the outputs of a PLL block.

As this block uses a running average window, one cycle of simulation must complete before the outputs give the correct magnitude and angle. For the first cycle of simulation, the outputs are held to the values specified by the initial input parameter.

Parameters

Initial frequency (Hz)

Specify the frequency of the first cycle of simulation. Default is 60.

Minimum frequency (Hz)

Specify the minimum frequency value that sets the buffer size of the Variable Time Delay block used inside the block to compute the fundamental value. Default is 45.

Initial input [Mag, Phase-relative-to-PLL (degrees)]

Specify the initial magnitude and phase in degrees of the input signal. Default is [1, 0].

Sample time

Specify the sample time of the block, in seconds. Set to 0 to implement a continuous block. Default is 0.

Inputs and Outputs**Freq**

Fundamental frequency (Hz) required by the computation. This input is normally connected to the output Freq of a PLL block.

wt

Angle of the reference frame (rad/s) required for the computation. This input is normally connected to the output wt of a PLL block.

In

Connects to the signal to be analyzed. Typical input signals are voltages or currents measured by the Current Measurement or Voltage Measurement block.

|u|Magnitude

Returns the magnitude of the fundamental in the same unit as the input signal.

∠uPhase

Returns the phase of the fundamental, in degrees, relative to the reference frame wt (input 2).

Characteristics

Sample Time	Specified in the Sample Time parameter Continuous if Sample Time = 0
Scalar Expansion	Yes, of the parameters
Dimensionalized	Yes, for Input 3 (In)

Examples

The power_FundamentalPLLDriven model compares the Fourier block (with a specified fundamental frequency of 60 Hz) output to the Fundamental (PLL-Driven) block output. The PLL-Driven block outputs accurate magnitude and phase even if the fundamental frequency of the input signal varies during the simulation.

The model sample time is parameterized by the Ts variable set to a default value of 50e-6 s. Set Ts to 0 in the command window to simulate the model in continuous mode.

Version History

Introduced in R2013a

Generic Power System Stabilizer

Implement generic power system stabilizer for synchronous machine



Libraries:

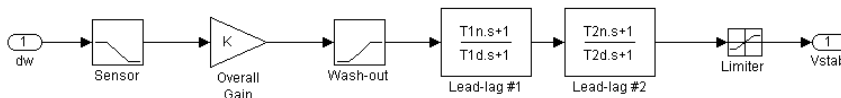
Simscape / Electrical / Specialized Power Systems / Electrical Machines / Synchronous Machine Control

Description

Note This block requires that you have a Control System Toolbox license. Otherwise, trying to simulate a model containing this block produces an error.

The Generic Power System Stabilizer (PSS) block can be used to add damping to the rotor oscillations of the synchronous machine by controlling its excitation. The disturbances occurring in a power system induce electromechanical oscillations of the electrical generators. These oscillations, also called power swings, must be effectively damped to maintain the system stability. The output signal of the PSS is used as an additional input (*vstab*) to the Excitation System block. The PSS input signal can be either the machine speed deviation, *dw*, or its acceleration power, $P_a = P_m - P_{eo}$ (difference between the mechanical power and the electrical power).

The Generic Power System Stabilizer is modeled by the following nonlinear system:



To ensure a robust damping, the PSS should provide a moderate phase advance at frequencies of interest in order to compensate for the inherent lag between the field excitation and the electrical torque induced by the PSS action.

The model consists of a low-pass filter, a general gain, a washout high-pass filter, a phase-compensation system, and an output limiter. The general gain *K* determines the amount of damping produced by the stabilizer. The washout high-pass filter eliminates low frequencies that are present in the *dw* signal and allows the PSS to respond only to speed changes. The phase-compensation system is represented by a cascade of two first-order lead-lag transfer functions used to compensate the phase lag between the excitation voltage and the electrical torque of the synchronous machine.

Ports

Input

In — Synchronous machine speed deviation or acceleration power
scalar

Input port that accepts one of these two inputs:

- The synchronous machine speed deviation $\Delta\omega$ signal (in pu)
- The synchronous machine acceleration power $P_a = P_m - P_{eo}$, which is the difference between the machine mechanical power and output electrical power (in pu)

Output

Vstab — Stabilization voltage
scalar

Stabilization voltage, in per unit. Connect this output port to the **Vstab** input port of the Excitation System block that you use to control the terminal voltage of the synchronous machine.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Sensor time constant — Sensor time constant
30e-3 (default) | positive scalar

Time constant, in seconds (s), of the first-order low-pass filter used to filter the block input signal.

Gain — Gain of generic power system stabilizer
20 (default)

Overall gain K of the generic power system stabilizer.

Wash-out time constant — Wash-out time constant
2 (default) | positive scalar

Time constant, in seconds (s), of the first-order high-pass filter used by the washout system of the model.

Lead-lag #1 time constants: [Tnum Tden] — Numerator and denominator time constants 1
[50e-3 20e-3] (default) | vector of two positive scalars

Numerator time constant T_{1n} and denominator time constant T_{1d} , in seconds (s), of the first lead-lag transfer function.

Lead-lag #2 time constants: [Tnum Tden] — Numerator and denominator time constants 2
[3 5.4] (default) | vector of two positive scalars

Numerator time constant T_{2n} and denominator time constant T_{2d} , in seconds (s), of the second lead-lag transfer function.

Output limits: [Vsmín Vsmáx] — Output limits
[-0.15 0.15] (default) | vector of two scalars

Limits V_{Smin} and V_{Smax} , in pu, imposed on the output of the stabilizer.

Initial input — Initial DC voltage input
0 (default)

Initial DC voltage, in pu, of the block input signal. Specification of this parameter is required to initialize all states and start the simulation in steady state with **vstab** set to zero.

Plot frequency response — Plot frequency response
off (default) | on

Select this option to display a plot of the frequency response of the stabilizer when you click the **Apply** button.

Magnitude in dB — Plot magnitude of frequency response
on (default) | off

Select this option to plot the magnitude of the frequency response to plot, in dB.

Dependencies

To enable this parameter, select the **Plot frequency response** parameter.

Frequency range — Frequency range
logspace(-2,2,500) (default)

Frequency range used to plot the frequency response of the stabilizer.

Dependencies

To enable this parameter, select the **Plot frequency response** parameter.

Version History

Introduced before R2006a

References

[1] Kundur, P., *Power System Stability and Control*, McGraw-Hill, 1994, Section 12.5.

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

Multiband Power System Stabilizer | Excitation System

Ground

Provide connection to ground



Library

Simscape / Electrical / Specialized Power Systems / Utilities

Description

The Ground block implements a connection to the ground.

See Also

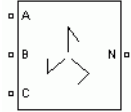
Neutral

Version History

Introduced before R2006a

Grounding Transformer

Implement three-phase grounding transformer providing a neutral in three-wire system

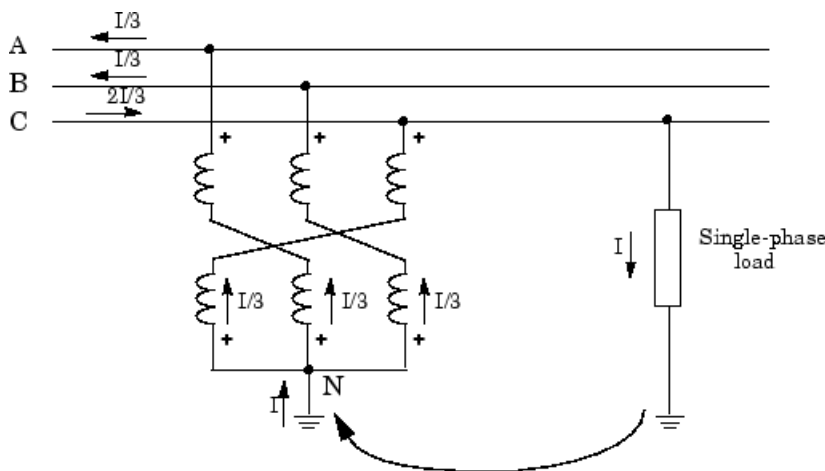


Library

Simscape / Electrical / Specialized Power Systems / Power Grid Elements

Description

Grounding transformers are used in utility distribution networks and in some power electronic converters in order to provide a neutral point in a three-wire system. This transformer is a three-phase two-winding transformer with winding 1 and winding 2 connected in zig zag as shown in the figure below.



The figure shows a single-phase load connected between phase C and ground in a three-wire system. The current I absorbed by the load returns to the source through the ground and the neutral of the grounding transformer. Because of the zig zag connection and the opposite winding polarities of upper and lower windings, the grounding transformer offers a low impedance in zero-sequence while keeping a very high impedance to positive-sequence. In other words, only a zero-sequence current can flow through the three windings. By definition, a zero-sequence current is a set of three-phase currents having same magnitude and phase. Therefore, the neutral current I shares into three equal currents $I/3$. Because the three currents flowing in the grounding transformer are equal, the neutral point stays fixed and the line-to-neutral voltages remain balanced.

The grounding transformer is modeled by three two-winding transformers having a 1:1 voltage ratio. Assume six identical windings with:

R = winding resistances
 X = winding leakage reactances

R_{mag}, X_{mag} = parallel resistance and reactance of the magnetizing branch

The positive-sequence impedance Z_1 and the zero-sequence impedance Z_0 of the grounding transformer are given by:

$$Z_1 = R_1 + jX_1 = 3 \frac{jR_{mag}X_{mag}}{(R_{mag} + X_{mag})}$$

$$Z_0 = R_0 + jX_0 = 2(R + jX).$$

The zero-sequence reactance X_0 is the most important parameter of the grounding transformer. In order to minimize voltage unbalance, reactance X_0 should be kept as low as possible.

Parameters

Units

Specify the units used to enter the parameters of the Grounding Transformer block. Select **pu** to use per unit. Select **SI** to use SI units. Changing the **Units** parameter from **pu** to **SI**, or from **SI** to **pu**, will automatically convert the parameters displayed in the mask of the block. The per unit conversion is based on the transformer rated power P_n in VA, nominal frequency f_n in Hz, and nominal voltage V_n , in Vrms, of the windings. Default is **pu**.

Nominal power and frequency

The nominal power rating, in volt-amperes (VA), and nominal frequency, in hertz (Hz), of the transformer. Note that the nominal parameters have no impact on the transformer model when the **Units** parameter is set to **SI**. Default is [100e6 60].

Nominal voltage

The nominal phase-to-phase voltage V_n of the Grounding Transformer, in volts RMS (Vrms). Default is 25e3.

Zero-sequence resistance and reactance

The zero-sequence resistance R_0 and the zero-sequence reactance X_0 in pu or in ohms. Default is [0.025 0.75] when the **Units** parameter is **pu** and [0.15625 4.6875] when the **Units** parameter is **SI**.

Magnetization branch

The shunt resistance R_m modeling the transformer core losses and the magnetizing reactance X_m modeling the magnetization current, in pu or ohms. Default is [500 500] when the **Units** parameter is **pu** and [3125 3125] when the **Units** parameter is **SI**.

These values define the active power losses P and the reactive power losses Q required for magnetizing the grounding transformer.

$$P = \frac{V_n^2}{R_m}$$

$$Q = \frac{V_n^2}{X_m}.$$

As the nominal voltage across each of the six windings is the nominal line-to-line voltage divided by 3 ($V_n/3$), the three impedances effectively connected across one of the two windings on each leg are $R_{mag}=R_m/3$ and $X_{mag}=X_m/3$.

Measurements

Select **Voltages** to measure the line-to-neutral voltages at the terminals of the Grounding Transformer block.

Select **Currents** to measure the currents flowing into the three terminals of the Grounding Transformer block.

Select **All voltages and currents** to measure voltages and currents.

Default is **None**.

Place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurements** list box of the Multimeter block, the measurements are identified by a label followed by the block name.

Measurement	Label
line-to-neutral voltages	Uan
currents	Ian

Limitations

Saturation of the grounding transformer is not modeled.

See Also

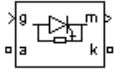
Linear Transformer, Multimeter, Three-Phase Transformer (Two Windings), Three-Phase Transformer (Three Windings)

Version History

Introduced in R2008a

GTO

Implement gate turn off (GTO) thyristor model



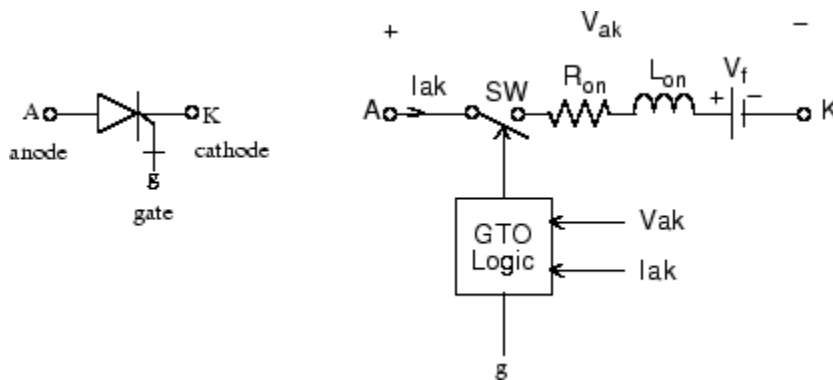
Library

Simscape / Electrical / Specialized Power Systems / Power Electronics

Description

The gate turnoff (GTO) thyristor is a semiconductor device that can be turned on and off via a gate signal. Like a conventional thyristor, the GTO thyristor can be turned on by a positive gate signal ($g > 0$). However, unlike the thyristor, which can be turned off only at a zero crossing of current, the GTO can be turned off at any time by the application of a gate signal equal to 0.

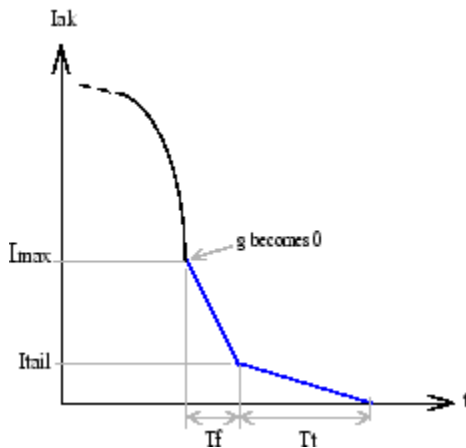
The GTO thyristor is simulated as a resistor R_{on} , an inductor L_{on} , and a DC voltage source V_f connected in series with a switch. The switch is controlled by a logical signal depending on the voltage V_{ak} , the current I_{ak} , and the gate signal g .



The V_f , R_{on} , and L_{on} parameters are the forward voltage drop while in conduction, the forward conducting resistance, and the inductance of the device. The GTO block also contains a series Rs-Cs snubber circuit that can be connected in parallel with the GTO device (between terminal ports A and K).

The GTO thyristor turns on when the anode-cathode voltage is greater than V_f and a positive pulse signal is present at the gate input ($g > 0$). When the gate signal is set to 0, the GTO thyristor starts to block but its current does not stop instantaneously.

Because the current extinction process of a GTO thyristor contributes significantly to the turnoff losses, the turnoff characteristic is built into the model. The current decrease is approximated by two segments. When the gate signal becomes 0, the current I_{ak} first decreases from the value I_{max} (value of I_{ak} when the GTO thyristor starts to open) to $I_{max}/10$, during the fall time (T_f), and then from $I_{max}/10$ to 0 during the tail time (T_t). The GTO thyristor turns off when the current I_{ak} becomes 0. The latching and holding currents are not considered.



Parameters

Resistance R_{on}

The internal resistance R_{on} , in ohms (Ω). Default is 0.001. The **Resistance R_{on}** parameter cannot be set to 0 when the **Inductance L_{on}** parameter is set to 0.

Inductance L_{on}

The internal inductance L_{on} , in henries (H). Default is 0. The **Inductance L_{on}** parameter is normally set to 0 except when the **Resistance R_{on}** parameter is set to 0.

Forward voltage V_f

The forward voltage of the GTO thyristor device, in volts (V). Default is 1.

Initial current I_c

You can specify an initial current flowing in the GTO thyristor. It is usually set to 0 in order to start the simulation with the device blocked. Default is 0.

If the **Initial Current I_C** parameter is set to a value greater than 0, the steady-state calculation considers the initial status of the GTO as closed. Initializing all states of a power electronic converter is a complex task. Therefore, this option is useful only with simple circuits.

Snubber resistance R_s

The snubber resistance, in ohms (Ω). Default is 1e5. Set the **Snubber resistance R_s** parameter to ∞ to eliminate the snubber from the model.

Snubber capacitance C_s

The snubber capacitance, in farads (F). Default is ∞ . Set the **Snubber capacitance C_s** parameter to 0 to eliminate the snubber, or to ∞ to get a resistive snubber.

Show measurement port

If selected, add a Simulink output to the block returning the GTO current and voltage. Default is selected.

Inputs and Outputs

g

Simulink signal to control the gate of the GTO.

m

The Simulink output of the block is a vector containing two signals. You can demultiplex these signals by using the Bus Selector block provided in the Simulink library.

Signal	Definition	Units
1	GTO current	A
2	GTO voltage	V

Assumptions and Limitations

The GTO block implements a macro model of a real GTO thyristor. It does not take into account either the geometry of the device or the underlying physical processes of the device [1].

The GTO block requires a continuous application of the gate signal ($g > 0$) in order to be in the on state (with $I_{ak} > 0$). The latching current and the holding current are not considered. The critical value of the derivative of the reapplied anode-cathode voltage is not considered.

Depending on the value of the inductance L_{on} , the GTO is modeled either as a current source ($L_{on} > 0$) or as a variable topology circuit ($L_{on} = 0$). The GTO block cannot be connected in series with an inductor, a current source, or an open circuit, unless its snubber circuit is in use.

The inductance L_{on} is forced to 0 if you choose to discretize your circuit.

References

[1] Mohan, N., T.M. Undeland, and W.P. Robbins, *Power Electronics: Converters, Applications, and Design*, John Wiley & Sons, Inc., New York, 1995.

See Also

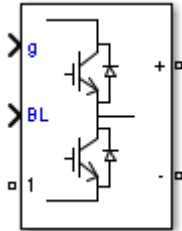
IGBT, Mosfet, Three-Level Bridge, Thyristor, Universal Bridge

Version History

Introduced before R2006a

Half-Bridge Converter

Implement half-bridge power converter



Libraries:

Simscape / Electrical / Specialized Power Systems / Power Electronics

Description

The Half-Bridge Converter block implements a half-bridge power converter. You can choose from four model types:

- **Switching devices** — The converter is modeled with IGBT/diode pairs controlled by firing pulses produced by a PWM generator. This model provides the most accurate simulation results.
- **Switching function** — The converter is modeled by a switching-function model. The switches are replaced with two voltage sources and two diodes on the AC side and with two current sources on the DC side.

The converter is controlled by firing pulses produced by a PWM generator (0/1 signals) or by firing pulses averaged over a specified period (PWM averaging: signals between 0 and 1). Both modes of operation produce harmonics normally generated by a PWM-controlled converter and also correctly simulate rectifying operation as well as blanking time. This model type is well-suited for real-time simulation.

- **Average model (Uref-controlled)** — The converter is modeled using a switching-function model directly controlled by the reference voltage signals (Uref). A PWM generator is not required.
- **Average model (No rectifier mode)** — The block uses the voltage source directly controlled by the reference voltage to model the converter. The model does not require a PWM generator and does not simulate the rectifier mode. This setting provides the fastest simulations.

When doing real-time simulation, use the **Switching function** setting with the firing pulses averaged, or the **Average model (Uref-controlled)** or **Average model (No rectifier mode)** settings.

Ports

Input

g — Gating signal to control converter
vector of two elements

Vectorized gating signal to control the converter. The gating signal contains two firing pulses to control Q1 and Q2.

Dependencies

To enable this port, set **Model type** to `Switching devices` or `Switching function`.

Uref — Reference voltage signal to control converter
vector of three elements

Reference voltage signal to control the converter.

Dependencies

To enable this port, set **Model type** to `Average model (Uref-controlled)` or `Average model (No rectifier mode)`.

BL — Input to block firing pulses
scalar

Input port to block all firing pulses to the converter. To block all firing pulses to the converter, apply a signal value of 1 to this port.

Conserving

+ — Positive terminal
specialized electrical

Specialized electrical conserving port associated with the positive terminal.

- — Negative terminal
specialized electrical

Specialized electrical conserving port associated with the negative terminal.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Model type — Model type

`Switching devices (default)` | `Switching function` | `Average model (Uref-controlled)` | `Average model (No rectifier mode)`

Select one of these values:

- **Switching devices** — The converter is modeled with IGBT/diode pairs controlled by firing pulses produced by a PWM generator.
- **Switching function** — The converter is modeled by a switching-function model. The switches are replaced with two voltage sources and two diodes on the AC side and with two current sources on the DC side.
- **Average model (Uref-controlled)** — The converter is modeled using a switching-function model directly controlled by the reference voltage signals (Uref).
- **Average model (No rectifier mode)** — The block uses the voltage source directly controlled by the reference voltage to model the converter. The model does not require a PWM generator and does not simulate the rectifier mode.

Device on-state resistance (Ohms) — Internal resistance of switching devices

1e-3 (default) | positive scalar

Internal resistance of the switching devices, in ohms.

Dependencies

To enable this parameter, set **Model type** to Switching devices.

Snubber resistance (Ohms) — Snubber resistance

1e6 (default) | positive scalar

Snubber resistance, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to Switching devices.

Snubber capacitance (F) — Snubber capacitance

`inf` (default) | nonnegative scalar

Snubber capacitance, in farads. Set the snubber capacitance to `0` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to Switching devices.

Diode on-state resistance (Ohms) — Internal resistance of diodes

1e-3 (default) | positive scalar

Internal resistance of the diodes, in ohms.

Dependencies

To enable this parameter, set **Model type** to Switching function or Average model (Uref-controlled).

Diode snubber resistance (Ohms) — Snubber resistance of diodes

1e6 (default) | positive scalar

Snubber resistance of the diodes, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to Switching function or Average model (Uref-controlled).

Diode snubber capacitance (F) — Snubber capacitance of diodes

`inf` (default) | nonnegative scalar

Snubber capacitance of the diodes, in farads. Set the snubber capacitance to `0` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to Switching function or Average model (Uref-controlled).

Diode forward voltage (V) — Forward voltage across diode

1e-3 (default) | nonnegative scalar

Forward voltage, in volts, across the diode when it is conducting.

Dependencies

To enable this parameter, set **Model type** to Switching function or Average model (Uref-controlled).

DC side snubber resistance (Ohms) — Snubber resistance across current sources

inf (default) | positive scalar

Snubber resistance across the two current sources, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to Switching function, Average model (Uref-controlled), or Average model (No rectifier mode).

AC side source resistance (Ohms) — Source resistance on AC side

1e-3 (default) | positive scalar

Source resistance on the AC side, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to Average model (No rectifier mode).

Version History

Introduced in R2015b

Extended Capabilities

C/C++ Code Generation

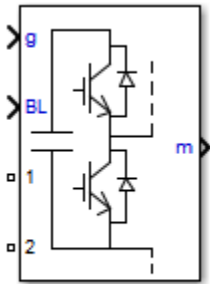
Generate C and C++ code using Simulink® Coder™.

See Also

Full-Bridge Converter

Half-Bridge MMC

Implement a half-bridge modular multilevel converter



Library

Simscape / Electrical / Specialized Power Systems / Power Electronics

Description

The Half-Bridge MMC block implements a half-bridge modular multilevel converter. The converter consists of multiple series-connected power modules. Each power module consists of one half-bridge and one capacitor on the DC side.

You can choose from four model types:

- **Switching devices** — The converter is modeled with IGBT/diode pairs. A multilevel PWM generator produces firing pulses (0/1 signals), which trigger switching in the converter.
- **Switching function** — The converter is based on a switching-function model. The model uses two voltage sources and two diodes on the AC side, and two current sources on the DC side.

The converter is controlled by firing pulses produced by a PWM generator (0/1 signals) or by firing pulses averaged over a specified period (PWM averaging: signals from 0 through 1). Both modes of operation produce harmonics normally generated by a PWM-controlled converter and also correctly simulate the rectifying operation and blanking time. This model type is suitable for real-time simulation.

- **Average model (Uref-controlled)** — The converter is modeled using a switching function model directly controlled by the reference voltage signals. A PWM generator is not required. This model provides the fastest simulations.
- **Aggregate model** — The converter is modeled using a switching-function model where only one equivalent module is used to represent all modules. This aggregate model represents control system dynamics, converter harmonics, and circulating currents phenomena. However, because a single virtual capacitor represents all capacitors of one arm, the model assumes that capacitor voltages of all power modules are balanced. As a result, a capacitor voltage-balancing scheme cannot be simulated. The aggregate model runs much faster than a detailed model that would use two switching devices and one capacitor for each individual power module of one arm. This aggregate model is also suitable for real-time simulation.

Parameters

Model type

Specify the model type to use:

- Switching devices (default)
- Switching function
- Average model (Uref-controlled)
- Aggregate model

Number of power modules

Specify the number of series-connected power modules that are in the converter. The default value is 1.

Capacitor value (F)

Specify the capacitance, in farads, of the capacitors connected on the DC side of each power module. You can specify a single value that sets all the capacitances to the same value, or specify a vector containing different capacitance values for every power module. The default value is $10\text{e-}3$.

Capacitor initial voltage (V)

Specify the capacitor initial voltage, in volts, of the capacitors connected on the DC side of each power module. You can specify a single value that sets all the capacitor initial voltage to the same value, or specify a vector containing different initial voltage for every power module. The default value is 1000.

Device on-state resistance (Ohms)

Internal resistance of the switching devices, in ohms. This parameter is available only when you set the **Model type** parameter to Switching devices. The default value is $1\text{e-}3$.

Snubber resistance (Ohms)

The snubber resistance, in ohms. To eliminate the snubbers, set the snubber resistance to `inf`. This parameter is available only when you set the **Model type** parameter to Switching devices. The default value is $1\text{e}6$.

Snubber capacitance (F)

The snubber capacitance, in farads. To eliminate the snubbers, set the snubber capacitance to 0. This parameter is available only when you set the **Model type** parameter to Switching devices. The default value is `inf`.

Diode on-state resistance (Ohms)

Internal resistance of the diodes, in ohms. This parameter is available only when you set the **Model type** parameter to Switching function, Average model (Uref-controlled), or Aggregate model. The default value is $1\text{e-}3$.

Diode snubber resistance (Ohms)

The snubber resistance, in ohms. To eliminate the snubbers, set the snubber resistance to `inf`. This parameter is available only when you set the **Model type** parameter to Switching function, Average model (Uref-controlled), or Aggregate model. The default value is $1\text{e}6$.

Diode snubber capacitance (F)

The snubber capacitance in farads. To eliminate the snubbers, set the snubber capacitance to 0. This parameter is available only when you set the **Model type** parameter to Switching

function, Average model (Uref-controlled), or Aggregate model. The default value is `inf`.

Diode forward voltage (V)

Forward voltage, in volts, across the diode when it is conducting. This parameter is available only when you set the **Model type** parameter to Switching function, Average model (Uref-controlled), or Aggregate model. The default value is `1e-3`.

Sample time (s)

Specify the sample time of the block. To implement a continuous block, set to `0`.

This parameter is available only when you set the **Model type** parameter to Switching function, Average model (Uref-controlled), or Aggregate model. The default value is `10e-6`.

Inputs and Outputs

g

Vectorized gating signal that controls the converter. The gating signal contains the firing pulses to control two switches at each power module in the converter (double the **Number of power modules** pulses). This port is visible only when you set the **Model type** parameter to Switching devices or Switching function.

Uref

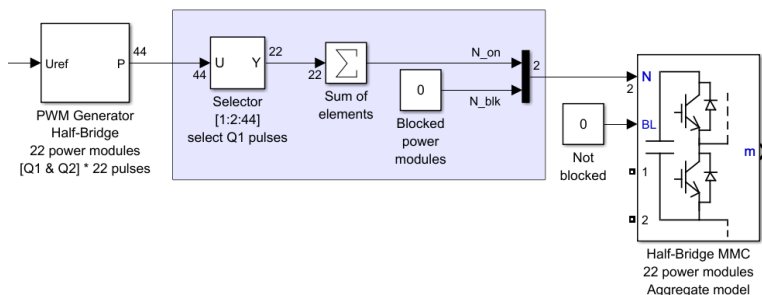
Vectorized reference voltage signal that controls the converter. The vectorized signal contains one reference voltage for each power module in the converter. This port is visible only when you set the **Model type** parameter to Average model (Uref-controlled).

N

Vectorized signal of two elements [N_{on} , N_{blk}]. N_{on} represents the number of power modules in the *ON* state, and N_{blk} the number of power modules in the blocked state.

The number of power modules in the blocked state is normally set to zero for a nominal operation of the converter. The number of power modules in the *ON* state is obtained by summing all the pulses that trigger the upper switching devices (Q1) that are in the *ON* state.

N_{blk} must be equal to `0` if the converter is deblocked and equal to the number of power modules if blocked. Then, as for the Average model, it is not possible to block an individual module.



This port is visible only when you set the **Model type** parameter to Aggregate model.

BL

You can block all firing pulses to the converter by applying a scalar signal value of 1 at the BL input. It is not possible to block an individual module. When the input signal is 1, all modules in the MMC are blocked.

1

Output terminal 1 of the converter.

2

Output terminal 2 of the converter.

m

Measurement signal containing the voltage of capacitors connected on the DC side of each power module. If you set the **Model type** parameter to **Aggregate model**, the measurement has only one element that gives the average value of all the capacitors.

Version History

Introduced in R2016b

See Also

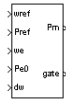
Half-Bridge Converter | Full-Bridge MMC | PWM Generator (Multilevel)

Topics

“Switching Function Converter Controlled by Averaged Firing Pulses”

Hydraulic Turbine and Governor

Model hydraulic turbine and proportional-integral-derivative (PID) governor system

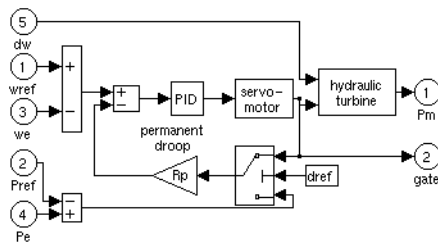


Libraries:

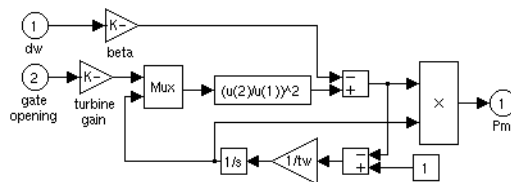
Simscape / Electrical / Specialized Power Systems / Electrical Machines / Synchronous Machine Control

Description

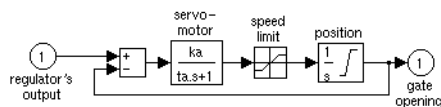
The Hydraulic Turbine and Governor block implements a nonlinear hydraulic turbine model, a PID governor system, and a servomotor [1].



The hydraulic turbine is modeled by the following nonlinear system.



The gate servomotor is modeled by a second-order system.



Ports

Input

wref — Reference speed
scalar

Reference speed, in pu.

Pref — Reference mechanical power
scalar

Reference mechanical power in pu. This input can be left unconnected if you want to use the gate position as input to the feedback loop instead of the power deviation.

we — Machine actual speed
scalar

Machine actual speed, in pu.

Pe0 — Machine actual electrical power
scalar

Machine actual electrical power in pu. This input can be left unconnected if you want to use the gate position as input to the feedback loop instead of the power deviation.

dw — Speed deviation
scalar

Speed deviation, in pu.

Output

Pm — Mechanical power
scalar

Mechanical power Pm for the Synchronous Machine block, in pu.

gate — Gate opening
scalar

Gate opening, in pu.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Servo-motor [Ka() Ta(sec)] — Gain and time constant of first-order system
[10/3 0.07] (default) | vector of two positive elements

Gain Ka and time constant Ta, in seconds (s), of the first-order system representing the servomotor.

Gate opening limits [gmin,gmax(pu) vgmin,vgmax(pu/s)] — Gate opening limits
[0.01 0.97518 -0.1 0.1] (default) | vector of four elements

Limits gmin and gmax (pu) imposed on the gate opening, and vgmin and vgmax (pu/s) imposed on gate speed.

Permanent droop and regulator [Rp() Kp() Ki() Kd() Td(s)] — Permanent droop and regulator
[0.05 1.163 0.105 0 0.01] (default) | vector of five nonnegative elements

Static gain of the governor is equal to the inverse of the permanent droop Rp in the feedback loop. The PID regulator has a proportional gain Kp, an integral gain Ki, and a derivative gain Kd. The high-frequency gain of the PID is limited by a first-order low-pass filter with time constant Td (s).

Hydraulic turbine [β (s) T_w (sec)] — Speed deviation damping coefficient and water starting time
[0 2.67] (default) | vector of two nonnegative elements

Speed deviation damping coefficient β and water starting time T_w (s).

Droop reference (0=power error, 1=gate opening) — Input of feedback loop
0 (default) | 1

Input of the feedback loop: gate position (1) or electrical power deviation (0).

Initial mechanical power (pu) — Initial mechanical power at machine shaft
0.7516 (default) | nonnegative scalar

Initial mechanical power P_{m0} (pu) at the machine shaft. This value is automatically updated by the load flow utility of the Powergui block.

Version History

Introduced before R2006a

References

- [1] IEEE Working Group on Prime Mover and Energy Supply Models for System Dynamic Performance Studies, "Hydraulic Turbine and Turbine Control Models for Dynamic Studies," *IEEE Transactions on Power Systems*, Vol. 7, No. 1, February, 1992, pp. 167-179.

Extended Capabilities

C/C++ Code Generation

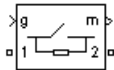
Generate C and C++ code using Simulink® Coder™.

See Also

Excitation System | Steam Turbine and Governor | Synchronous Machine SI Fundamental | Synchronous Machine pu Fundamental | Synchronous Machine pu Standard

Ideal Switch

Implement ideal switch device

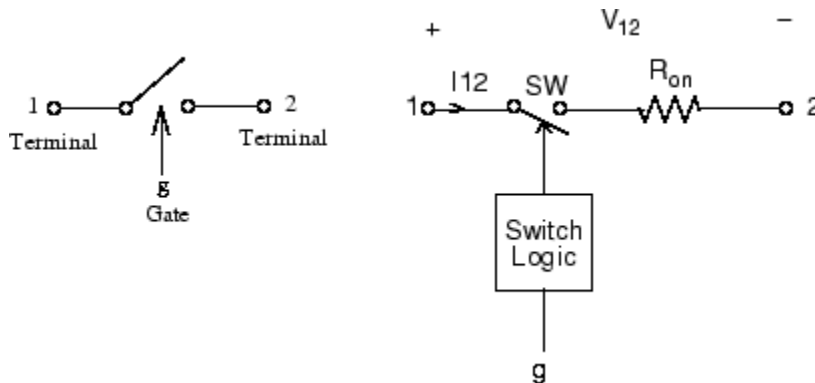


Libraries:

Simscape / Electrical / Specialized Power Systems / Power Electronics

Description

The Ideal Switch block does not correspond to a particular physical device. When used with appropriate switching logic, it can be used to model simplified semiconductor devices such as a GTO or a MOSFET, or even a power circuit breaker with current chopping. The switch is simulated as a resistor R_{on} in series with a switch controlled by a logical gate signal g .

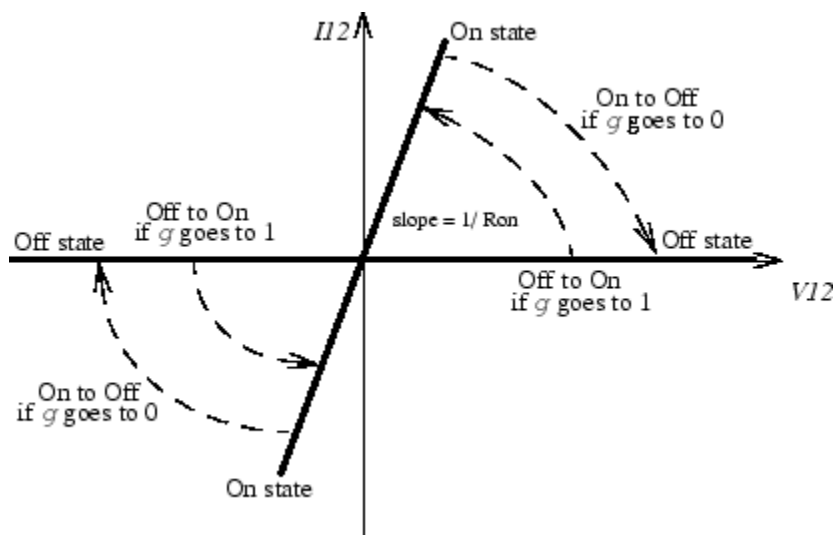


The Ideal Switch block is fully controlled by the gate signal ($g > 0$ or $g = 0$). It has the following characteristics:

- Blocks any forward or reverse applied voltage with 0 current flow when $g = 0$
- Conducts any bidirectional current with quasi-zero voltage drop when $g > 0$
- Switches instantaneously between on and off states when triggered

The Ideal Switch block turns on when a positive signal is present at the gate input ($g > 0$). It turns off when the gate signal equals 0 ($g = 0$).

The Ideal Switch block also contains a series R_s - C_s snubber circuit that can be connected in parallel with the ideal switch (between nodes 1 and 2).



Examples

The `power_switch` example uses the Ideal Switch block to switch an RLC circuit on an AC source (60 Hz). The switch, which is initially closed, is first opened at $t = 50$ ms (3 cycles) and then reclosed at $t = 138$ ms (8.25 cycles). The Ideal Switch block has 0.01 ohms resistance and no snubber is used.

Assumptions and Limitations

The Ideal Switch block is modeled as a current source. It cannot be connected in series with an inductor, a current source, or an open circuit, unless its snubber circuit is in use.

Ports

Input

g — Switch control signal
Simulink signal

Simulink signal to control the opening and closing of the switch.

Output

m — Switch output
Simulink signal

The Simulink output of the block is a vector containing two signals. You can demultiplex these signals by using the Bus Selector block provided in the Simulink library;

Signal	Definition	Units
1	Ideal switch current	A
2	Ideal switch voltage	V

Conserving

1 — Ideal switch terminal 1
specialized electrical

Specialized electrical conserving port associated with the terminal 1 of the ideal switch.

2 — Ideal switch terminal 2
specialized electrical

Specialized electrical conserving port associated with the terminal 2 of the ideal switch.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Internal resistance Ron — Internal resistance of the switch device
0.001 (default) | positive scalar

Internal resistance of the switch device, in ohms (Ω). The **Internal resistance Ron** parameter cannot be set to 0.

Initial state — Initial state of block
0 (default) | nonnegative scalar

Initial state of the Ideal Switch block. The initial status of the Ideal Switch block is taken into account in the steady-state calculation.

Snubber resistance Rs — Snubber resistance
1e5 (default) | nonnegative scalar

Snubber resistance, in ohms (Ω). Default is 1e5. Set the **Snubber resistance Rs** parameter to `inf` to eliminate the snubber from the model.

Snubber capacitance Cs — Snubber capacitance
`inf` (default) | nonnegative scalar

Snubber capacitance in farads (F). Default is `inf`. Set the **Snubber capacitance Cs** parameter to 0 to eliminate the snubber, or to `inf` to get a resistive snubber.

Show measurement port — Option to show measurement port
`on` (default) | `off`

If selected, add a Simulink output to the block returning the ideal switch current and voltage.

Version History

Introduced before R2006a

References

[1] Mohan, N., T.M. Undeland, and W.P. Robbins, *Power Electronics: Converters, Applications, and Design*, John Wiley & Sons, Inc., New York, 1995.

Extended Capabilities

C/C++ Code Generation

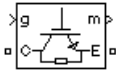
Generate C and C++ code using Simulink® Coder™.

See Also

Breaker

IGBT

Implement insulated gate bipolar transistor (IGBT)

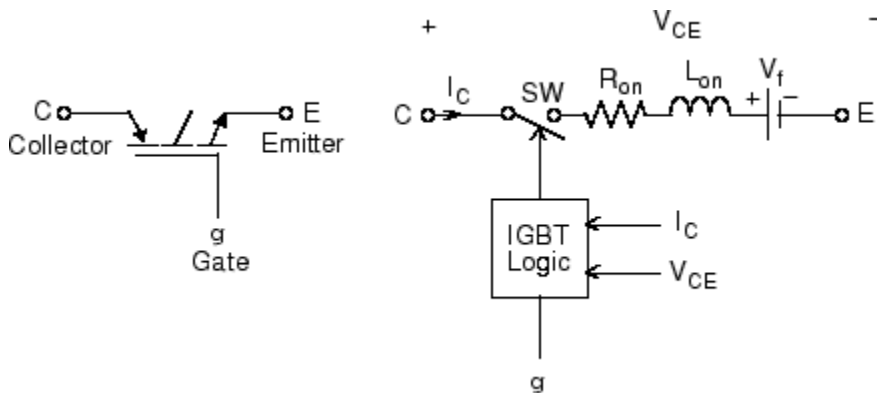


Libraries:

Simscape / Electrical / Specialized Power Systems / Power Electronics

Description

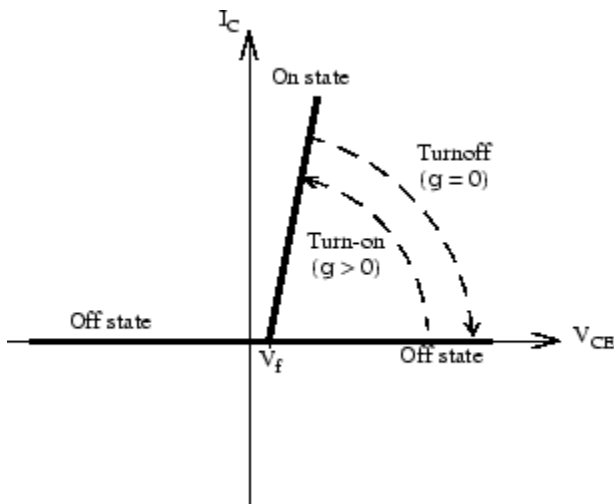
The IGBT block implements a semiconductor device controllable by the gate signal. The IGBT is simulated as a series combination of a resistor R_{on} , inductor L_{on} , and a DC voltage source V_f in series with a switch controlled by a logical signal ($g > 0$ or $g = 0$).



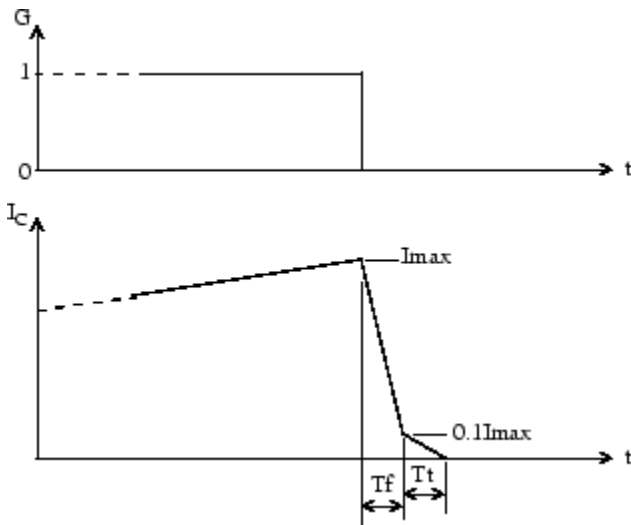
The IGBT turns on when the collector-emitter voltage is positive and greater than V_f and a positive signal is applied at the gate input ($g > 0$). It turns off when the collector-emitter voltage is positive and a 0 signal is applied at the gate input ($g = 0$).

The IGBT device is in the off state when the collector-emitter voltage is negative. Note that many commercial IGBTs do not have the reverse blocking capability. Therefore, they are usually used with an antiparallel diode.

The IGBT block contains a series R_s - C_s snubber circuit, which is connected in parallel with the IGBT device (between terminals C and E).



The turnoff characteristic of the IGBT model is approximated by two segments. When the gate signal falls to 0, the collector current decreases from I_{max} to $0.1 I_{max}$ during the fall time (T_f), and then from $0.1 I_{max}$ to 0 during the tail time (T_t).



Examples

The `power_igbtconv` example illustrates the use of the IGBT block in a boost DC-DC converter. The IGBT is switched on and off at a frequency of 10 kHz to transfer energy from the DC source to the load (RC). The average output voltage (V_R) is a function of the duty cycle (α) of the IGBT switch:

$$V_R = \frac{1}{1 - \alpha} V_{dc}$$

Assumptions and Limitations

- The IGBT block implements a macro model of the real IGBT device. It does not take into account either the geometry of the device or the complex physical processes [1].

- Depending on the value of the inductance L_{on} , the IGBT is modeled either as a current source ($L_{on} > 0$) or as a variable topology circuit ($L_{on} = 0$). The IGBT block cannot be connected in series with an inductor, a current source, or an open circuit, unless its snubber circuit is in use.
- The inductance L_{on} is forced to 0 if you choose to discretize your circuit.

Ports

Input

g — IGBT control signal
Simulink signal

Simulink signal to control the opening and closing of the IGBT.

Output

m — Switch output
Simulink signal

The Simulink output of the block is a vector containing two signals. You can demultiplex these signals by using the Bus Selector block provided in the Simulink library:

Signal	Definition	Units
1	IGBT current	A
2	IGBT voltage	V

Dependencies

To enable this port, select the **Show measurement port** parameter.

Conserving

C — IGBT collector
specialized electrical

Specialized electrical conserving port associated with the IGBT collector.

E — IGBT emitter
specialized electrical

Specialized electrical conserving port associated with the IGBT emitter.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Resistance Ron — Internal resistance of IGBT
0.001 (default) | nonnegative scalar

Internal resistance of the switch device, in ohms (Ω). The **Resistance Ron** parameter cannot be set to 0 when the **Inductance Lon** parameter is set to 0.

Inductance L_{on} — Internal inductance of IGBT 0 (default) | nonnegative scalar

Internal inductance L_{on} , in henries (H). The **Inductance L_{on}** parameter is normally set to 0 except when the **Resistance R_{on}** parameter is set to 0 .

Forward voltage V_f — Forward voltage of IGBT 1 (default) | nonnegative scalar

Forward voltage of the IGBT device, in volts.

Initial current I_c — Initial current flowing in IGBT 0 (default) | nonnegative scalar

Initial current flowing in the IGBT. It is usually set to 0 to start the simulation with the device blocked.

If the **Initial current I_c** parameter is set to a value greater than 0 , the steady-state calculation considers the initial status of the IGBT as closed. Initializing all states of a power electronic converter is a complex task. Therefore, this option is useful only with simple circuits.

Snubber resistance R_s — Snubber resistance $1e5$ (default) | nonnegative scalar

Snubber resistance, in ohms (Ω). Default is $1e5$. Set the **Snubber resistance R_s** parameter to inf to eliminate the snubber from the model.

Snubber capacitance C_s — Snubber capacitance inf (default) | nonnegative scalar

Snubber capacitance in farads (F). Default is inf . Set the **Snubber capacitance C_s** parameter to 0 to eliminate the snubber, or to inf to get a resistive snubber.

Show measurement port — Option to show measurement port on (default) | off

If selected, add a Simulink output to the block returning the IGBT current and voltage.

Version History

Introduced before R2006a

References

- [1] Mohan, N., T.M. Undeland, and W.P. Robbins, *Power Electronics: Converters, Applications, and Design*, John Wiley & Sons, Inc., New York, 1995.

Extended Capabilities

C/C++ Code Generation

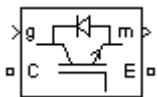
Generate C and C++ code using Simulink® Coder™.

See Also

GTO | Mosfet | Three-Level Bridge | Thyristor

IGBT/Diode

Implements ideal IGBT, GTO, or MOSFET and antiparallel diode



Libraries:
Simscape / Electrical / Specialized Power Systems / Power Electronics

Description

The IGBT/Diode block is a simplified mode of an IGBT (or GTO or MOSFET)/Diode pair where the forward voltages of the forced-commutated device and diode are ignored.

Assumptions and Limitations

- The IGBT/Diode block implements a macro model of the real IGBT and Diode devices. It does not take into account either the geometry of the devices or the complex physical processes [1].
- The IGBT/Diode block cannot be connected in series with an inductor, a current source, or an open circuit, unless its snubber circuit is in use.

Ports

Input

g — Device control signal
Simulink signal

Simulink signal to control the opening and closing of the device.

Output

m — Switch output
Simulink signal

The Simulink output of the block is a vector containing two signals. You can demultiplex these signals by using the Bus Selector block provided in the Simulink library:

Signal	Definition	Units
1	IGBT/Diode current	A
2	IGBT/Diode voltage	V

Dependencies

To enable this port, select the **Show measurement port** parameter.

Conserving

C — IGBT collector
specialized electrical

Specialized electrical conserving port associated with the IGBT/Diode collector.

E — IGBT emitter
specialized electrical

Specialized electrical conserving port associated with the IGBT/Diode emitter.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Internal resistance Ron — Internal resistance of IGBT
1e-3 (default) | nonnegative scalar

Internal resistance of the device, in ohms (Ω).

Snubber resistance Rs — Snubber resistance
1e5 (default) | nonnegative scalar

Snubber resistance, in ohms (Ω). Set the **Snubber resistance Rs** parameter to `inf` to eliminate the snubber from the model.

Snubber capacitance Cs — Snubber capacitance
`inf` (default) | nonnegative scalar

Snubber capacitance in farads (F). Set the **Snubber capacitance Cs** parameter to `0` to eliminate the snubber, or to `inf` to get a resistive snubber.

Show measurement port — Option to show measurement port
`on` (default) | `off`

If selected, add a Simulink output to the block returning the diode IGBT current and voltage.

Version History

Introduced in R2006a

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using Simulink® Coder™.

See Also

IGBT | Diode

Impedance Measurement

Measure impedance of circuit as function of frequency



Libraries:

Simscape / Electrical / Specialized Power Systems / Sensors and Measurements

Description

The Impedance Measurement block measures the impedance between two nodes of a linear circuit as a function of the frequency. It consists of a current source I_z , connected between inputs one and two of the Impedance Measurement blocks, and a voltage measurement V_z , connected across the terminals of the current source. The network impedance is calculated as the transfer function $H(s)$ from the current input to the voltage output of the state-space model.

$$H(s) = \frac{V_z(s)}{I_z(s)}.$$

The impedance (magnitude and phase) as function of frequency is displayed by using the Impedance versus Frequency Measurement tool of the Powergui block.

The measurement takes into account the initial states of the Breaker and Ideal Switch blocks. It also allows impedance measurements with Distributed Parameters Line blocks in your circuit.

The source impedances are taken into account in the impedance calculations. By definition, the impedance of voltage sources is 0 (voltage source blocks are considered as short-circuit) and the impedance of current sources is infinite (current source blocks are considered as open-circuit).

In some applications, you may need to delete current or voltage source blocks manually in order to ignore their impact on impedance calculations.

Assumptions and Limitations

The only nonlinear blocks that are taken into account during the impedance measurement are the Breaker, Three-Phase Breaker, Three-Phase Fault, Ideal Switch, and Distributed Parameters Line blocks. All other nonlinear blocks, such as machines and power electronic devices, are not considered, and they are disconnected during the measurement.

If you plan to connect the Impedance Measurement block in series with an inductance, a current source, or any nonlinear element, you must add a large resistor across the terminals of the block, because the Impedance Measurement block is simulated as a current source block.

Ports

Conserving

Cons 1 — Electric terminal of line
specialized electrical

Specialized electrical conserving port associated with one node of a linear circuit.

Cons 2 — Electric terminal of line
specialized electrical

Specialized electrical conserving port associated with one node of a linear circuit.

Parameters

Multiplication factor — Rescale value for measured impedance
1 (default) | positive scalar

If you use the Impedance Measurement block in a three-phase circuit, you can use the **Multiplication factor** parameter to rescale the measured impedance. For example, measuring the impedance between two phases of a three-phase circuit gives two times the positive-sequence impedance. Therefore you must apply a multiplication factor of 1/2 to the impedance in order to obtain the correct positive-sequence impedance value.

Similarly, to measure the zero-sequence impedance of a balanced three-phase circuit, you can connect the Impedance Measurement block between ground or neutral and the three phases connected together.

In that case, you are measuring one third of the zero-sequence impedance and you must apply a multiplication factor of 3 to obtain the correct zero-sequence value.

Version History

Introduced before R2006a

Extended Capabilities

C/C++ Code Generation

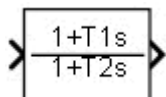
Generate C and C++ code using Simulink® Coder™.

See Also

power_zmeter

Lead-Lag Filter

Implement first-order lead-lag filter



Libraries:

Simscape / Electrical / Specialized Power Systems / Control / Filters

Description

The Lead-Lag Filter block implements the following transfer function:

$$H(s) = \frac{1 + T_1 s}{1 + T_2 s}$$

where

s = Laplace operator

T_1, T_2 = time constants

This type of filter is used mainly for implementing lead-lag compensation in control systems. The key characteristics of the Lead-Lag Filter block are:

- Input accepts a vectorized input of N signals, thus implementing N filters. This feature is particularly useful for designing controllers in three-phase systems ($N=3$).
- The same block is used for continuous or discrete model. Changing the sample time T_s from 0 to a positive value automatically discretizes the filter, and vice versa.
- Filter states can be initialized for specified DC inputs and outputs.

Examples

The `power_LeadLagFilter` example shows two uses of a vectorized Lead-Lag Filter.

The model sample time is parameterized with variable T_s (default value $T_s = 50e-6$). To simulate continuous filters, specify $T_s = 0$ in the MATLAB® Command Window before simulating the model.

Characteristics

Direct Feedthrough	Yes
Sample Time	Specified in the Sample Time parameter Continuous if Sample Time = 0
Scalar Expansion	Yes, of the parameters
States	One state per filter
Dimensionalized	Yes

Ports

Input

In1 — Input signal
scalar | vector

Input signal the block applies the transfer function to.

Output

Out1 — Output signal
scalar | vector

Output after the block has applied the transfer function.

Parameters

Time constant T1 (s) — T_1 filter time constant
5e-3 (default) | positive scalar

Specify the filter time constant(s) T_1 in seconds.

Time constant T2 (s) — T_2 filter time constant
20e-3 (default) | positive scalar

Specify the filter time constant(s) T_2 in seconds.

DC initial input and output — Input and output signal initial DC value
0 (default) | scalar | vector

Specify the DC initial value of input and output signals. If the input signal is vectorized, specify a 1-by-N vector, where each value corresponds to a particular input.

Sample time — Block sample time
0 (default) | positive scalar

Specify the sample time of the block, in seconds.

Version History

Introduced in R2013a

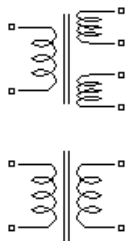
Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

Linear Transformer

Implement two- or three-winding linear transformer

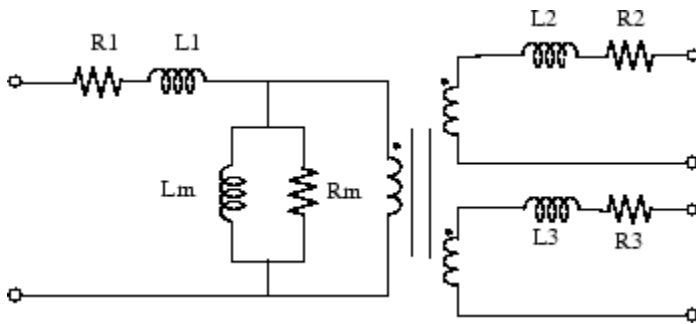


Libraries:

Simscape / Electrical / Specialized Power Systems / Power Grid Elements

Description

The Linear Transformer block model shown consists of three coupled windings wound on the same core.



The model takes into account the winding resistances ($R1$ $R2$ $R3$) and the leakage inductances ($L1$ $L2$ $L3$), as well as the magnetizing characteristics of the core, which is modeled by a linear (Rm Lm) branch.

The Per Unit Conversion

In order to comply with industry, the block allows you to specify the resistance and inductance of the windings in per unit (pu). The values are based on the transformer rated power P_n in VA, nominal frequency f_n in Hz, and nominal voltage V_n in Vrms, of the corresponding winding. For each winding, the per unit resistance and inductance are defined as

$$R(pu) = \frac{R(\Omega)}{R_{base}}$$

$$L(pu) = \frac{L(H)}{L_{base}}$$

The base impedance, base resistance, base reactance, and base inductance used for each winding are

$$Z_{base} = R_{base} = X_{base} = \frac{(V_n)^2}{P_n}$$

$$L_{base} = \frac{X_{base}}{2\pi f n}$$

For the magnetization resistance R_m and inductance L_m , the pu values are based on the transformer rated power and on the nominal voltage of winding 1.

For example, the default parameters of winding 1 specified in the dialog box section give the following bases:

$$R_{base} = \frac{(735e3)^2}{250e6} = 2161\Omega$$

$$L_{base} = \frac{2161}{2\pi 60} = 5.732H$$

Suppose that the winding 1 parameters are $R_1 = 4.32\Omega$ and $L_1 = 0.4586H$; enter the corresponding values in the dialog box:

$$R_1 = \frac{4.32\Omega}{2161\Omega} = 0.002pu$$

$$L_1 = \frac{0.4586H}{5.732H} = 0.08pu$$

To specify a magnetizing current of 0.2% (resistive and inductive) based on nominal current, you must enter per unit values of $1/0.002 = 500$ pu for the resistance and the inductance of the magnetizing branch. Using the base values calculated previously, these per unit values correspond to $R_m = 1.08e6$ ohms and $L_m = 2866$ H.

Modeling an Ideal Transformer

To implement an ideal transformer model, set the winding resistances and inductances to 0, and the magnetization resistance and inductance (R_m L_m) to `inf`.

Limitations

Windings can be left floating (that is, not connected to the rest of the circuit). However, an internal resistor is automatically added between the floating winding and the main circuit. This internal connection does not affect voltage and current measurements.

Due to limitations inherent to graph theory and its application to electric network theory as implemented in Simscape Electrical Specialized Power Systems, the following topologies are unsolvable:

- Loops containing only ideal transformer secondary windings (for example, delta-connected ideal secondary windings of three-phase transformer). To solve this topology issue, you can add a small impedance in series with the loop.
- Loops containing only ideal transformer secondary windings and ideal voltage sources. To solve this topology issue, you can add a small impedance in series with the loop.
- Loops containing only ideal transformer secondary windings and capacitors. To solve this topology issue, you can add a small impedance in series with the loop.
- All topologies where an ideal transformer primary has at least one of its nodes that is connected to elements consisting only of ideal transformer primary windings or current sources (for example,

wye-connected three-phase primary windings with floating common point). To resolve this circuit topology, you connect a small resistance to problematic node.

Examples

The `power_transformer` example shows a typical residential distribution transformer network feeding line-to-neutral and line-to-line loads.

To open this example, at the MATLAB Command Window enter `power_transformer`.

Ports

Conserving

+1 — Primary winding positive polarity
specialized electrical

Specialized electrical conserving port associated with the primary winding positive polarity.

-1 — Primary winding negative polarity
specialized electrical

Specialized electrical conserving port associated with the primary winding negative polarity.

+2 — Secondary winding positive polarity
specialized electrical

Specialized electrical conserving port associated with the secondary winding positive polarity.

-2 — Secondary winding negative polarity
specialized electrical

Specialized electrical conserving port associated with the secondary winding negative polarity.

+3 — Tertiary winding positive polarity
specialized electrical

Specialized electrical conserving port associated with the tertiary winding positive polarity.

Dependencies

To enable this port, select the **Three windings transformer** parameter.

-3 — Tertiary winding negative polarity
specialized electrical

Specialized electrical conserving port associated with the tertiary winding negative polarity.

Dependencies

To enable this port, select the **Three windings transformer** parameter.

Parameters

Units — Units
pu (default) | SI

Specify the units used to enter the parameters of the Linear Transformer block.

Set to **pu** to use per unit.

Set to **SI** to use SI units.

Changing the **Units** parameter from **pu** to **SI** or from **SI** to **pu** automatically converts the parameters displayed in the mask of the block. The per unit conversion is based on the transformer rated power P_n in VA, nominal frequency f_n in Hz, and nominal voltage V_n in V_{rms} , of the windings.

Nominal power and frequency [Pn(VA) fn(Hz)] — Nominal power and frequency
[250e6 60] (default)

The nominal power rating P_n in VA and frequency f_n in Hz, of the transformer.

This parameter does not impact the transformer model when you set the **Units** parameter to **SI**.

Winding 1 parameters [V1(Vrms) R1(pu) L1(pu)] — Winding 1 parameters
[7.35e+05 0.002 0.08] (default)

The nominal voltage V_1 in volts RMS, resistance R_1 in pu, and leakage inductance L_1 in pu, of the primary winding.

The pu values are based on the nominal power P_n and on V_1 .

To implement an ideal winding, set R_1 and L_1 to 0.

Dependencies

To enable this parameter, set the **Units** parameter to **pu**.

Winding 1 parameters [V1(Vrms) R1(ohm) L1(H)] — Winding 1 parameters
[7.35e+05 4.3218 0.45856] (default)

The nominal voltage V_1 in volts RMS, resistance R_1 in ohms, and leakage inductance L_1 in H, of the primary winding.

To implement an ideal winding, set R_1 and L_1 to 0.

Dependencies

To enable this parameter, set the **Units** parameter to **SI**.

Winding 2 parameters [V2(Vrms) R2(pu) L2(pu)] — Winding 2 parameters
[3.15e+05 0.002 0.08] (default)

The nominal voltage, V_2 in volts RMS, resistance R_2 in pu, and leakage inductance L_2 in pu, of the secondary winding.

The pu values are based on the nominal power P_n and on V_2 .

To implement an ideal winding, set R_2 and L_2 to 0.

Dependencies

To enable this parameter, set the **Units** parameter to **pu**.

Winding 2 parameters [V2(Vrms) R2(ohm) L2(H)] — Winding 2 parameters
[3.15e+05 0.7938 0.084225] (default)

The nominal voltage V2 in volts RMS, resistance R2 in ohms, and leakage inductance L2 in H, of the secondary winding.

To implement an ideal winding, set R2 and L2 to 0.

Dependencies

To enable this parameter, set the **Units** parameter to SI.

Three windings transformer — Three windings transformer
on (default) | off

Select this option to implement a linear transformer with three windings.

Clear this option to implement a linear transformer with two windings.

A transformer with the number of selected windings is displayed in the block icon.

Winding 3 parameters [V3(Vrms) R3(pu) L3(pu)] — Winding 3 parameters
[3.15e+05 0.002 0.08] (default)

The nominal voltage V3 in volts RMS, resistance R3 in pu, and leakage inductance L3 in pu, of the tertiary winding.

The pu values are based on the nominal power Pn and on V3.

To implement an ideal winding, set R3 and L3 to 0.

Dependencies

To enable this parameter, set the **Units** parameter to pu and select the **Three windings transformer** parameter.

Winding 3 parameters [V3(Vrms) R3(ohm) L3(H)] — Winding 3 parameters
[3.15e+05 0.7938 0.084225] (default)

The nominal voltage V3 in volts RMS, resistance R3 in ohms, and leakage inductance L3 in H, of the tertiary winding.

To implement an ideal winding, set R3 and L3 to 0.

Dependencies

To enable this parameter, set the **Units** parameter to SI and select the **Three windings transformer** parameter.

Magnetization resistance and inductance [Rm(pu) Lm(pu)] — Magnetization resistance and inductance
[500 500] (default)

The resistance Rm and inductance Lm simulating the core active and reactive losses.

The pu values are based on the nominal power Pn and on V1. For example, to specify 0.2% of active and reactive core losses, at nominal voltage, use Rm = 500 pu and Lm = 500 pu.

To implement an ideal winding, set this parameter to `[inf inf]`.

R_m must have a finite value when the inductance of winding 1 is greater than zero.

Dependencies

To enable this parameter, set the **Units** parameter to pu.

Magnetization resistance and inductance [$R_m(\text{ohm})$ $L_m(\text{H})$] — Magnetization resistance and inductance

`[1.0805e+06 2866]` (default)

The resistance R_m and inductance L_m simulating the core active and reactive losses.

To implement an ideal winding, set this parameter to `[inf inf]`.

R_m must have a finite value when the inductance of winding 1 is greater than zero.

Dependencies

To enable this parameter, set the **Units** parameter to SI.

Measurements — Measurements

None (default) | Winding voltages | Winding currents | Magnetization current | All voltages and currents

Set to **Winding voltages** to measure the voltage across the winding terminals of the Linear Transformer block.

Set to **Winding currents** to measure the current flowing through the windings of the Linear Transformer block.

Set to **Magnetization current** to measure the magnetization current of the Linear Transformer block.

Set to **All voltages and currents** to measure the winding voltages and currents plus the magnetization current.

Place a Multimeter block in your model to display the selected measurements during the simulation.

In the **Available Measurements** list box of the Multimeter block, the measurements are identified by a label followed by the block name.

Measurement	Label
Winding voltages	U_{w1} :
Winding currents	I_{w1} :
Magnetization current	I_{mag} :

Version History

Introduced before R2006a

Extended Capabilities

C/C++ Code Generation

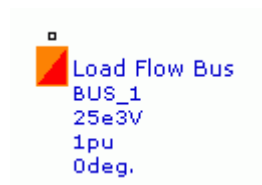
Generate C and C++ code using Simulink® Coder™.

See Also

Mutual Inductance | Saturable Transformer | Three-Phase Transformer (Two Windings) | Three-Phase Transformer (Three Windings) | Multimeter

Load Flow Bus

Identify and parameterize load flow bus



Libraries:

Simscape / Electrical / Specialized Power Systems / Utilities

Description

You use the Load Flow Bus block in models to specify the bus locations and parameters to solve a load flow. The Powergui Load Flow tool solves the load flow. Simscape Electrical Specialized Power Systems allows you to perform two types of load flows:

- Positive-sequence load flow applied to a three-phase system. Positive-sequence voltages as well as active power (P) and reactive power (Q) flows are computed at each three-phase bus.
- Unbalanced load flow applied to a mix of three-phase, two-phase, and single-phase systems. Individual phase voltage and PQ flow are computed for each phase.

To perform a positive-sequence load flow, you must define all of the Load Flow Bus blocks in your model with the **Connectors** parameter set to `single`. For this type of load flow, the load flow blocks are:

- Asynchronous Machine
- Simplified Synchronous Machine
- Synchronous Machine
- Three-Phase Dynamic Load
- Three-Phase Parallel RLC Load
- Three-Phase Programmable Voltage Source
- Three-Phase Series RLC Load
- Three-Phase Source

To perform an unbalanced load flow, you must define all of the Load Flow Bus blocks in your model with the **Connectors** parameter set to one of the following values: ABC, AB, AC, BC, A, B, or C. For an unbalanced load flow, the load flow blocks are:

- AC Voltage Source
- Asynchronous Machine
- Parallel RLC Load
- Series RLC Load
- Synchronous Machine
- Three-Phase Dynamic Load
- Three-Phase Parallel RLC Load

- Three-Phase Series RLC Load
- Three-Phase Source

The Powergui Load Flow tool reports an error if a model contains a mix of Load Flow Bus blocks set to perform a positive-sequence load flow and Load Flow Bus blocks set to perform an unbalanced load flow.

When several load flow blocks are connected together at the same bus, only one Load Flow Bus block is required. You can also connect the Load Flow Bus block at a location where you are interested in monitoring the load flow, even if no load flow blocks are connected at that location.

Implicit Buses (for Positive-Sequence Load Flow Only)

If you omit to connect a Load Flow Bus block to a load flow block, the Load Flow tool will automatically define an implicit (internal) load flow bus for that block. The bus base voltage of this implicit bus is set to the same value as the nominal voltage of the load flow block. When several load flow blocks are connected together, one nominal voltage is arbitrarily selected among the blocks.

Although the Load Flow tool can perform load flow on a model with no Load Flow Bus block in the model (working only with implicit buses), the recommended practice is to connect a Load Flow Bus block everywhere a load flow block exists.

Examples

The “Synchronous Machine” example contains two Load Flow Bus blocks. The B1 block is connected to the 5 MW Three-Phase Parallel RLC Load and to the Synchronous Machine load flow blocks. The B2 block is connected to the 10 MW Three-Phase Parallel RLC Load and to the Three-Phase Source load flow blocks.

The two blocks display the bus voltages and angles resulting from a load flow solution previously saved with the model. The bus voltage at bus B1 displays 1 pu, -23.67 deg. Start the simulation, verify it starts in steady state, then zoom on the initial value of the measured voltage Va. You should read an initial voltage of about -0.4015 pu. At the MATLAB command prompt, compute the angle of the measured voltage:

```
asin(-0.4015)*180/pi
```

```
ans =  
    -23.672
```

You can also take a look at the “Initializing a 5-Bus Network with the Load Flow Tool of Powergui” model. In this model five Load Flow Bus blocks are used. The block labeled B120 is the swing bus.

The “IEEE 13 Node Test Feeder” example contains Load Flow Bus blocks set to perform an unbalanced load flow for a 13-node test feeder IEEE benchmark model.

Ports

Conserving

A — Phase A connector
specialized electrical

Specialized electrical conserving port associated with the phase A connector. If you set the **Configuration** parameter to on both sides, this port appears twice.

Dependencies

To enable this port, set **Connectors** to ABC, AB, AC, or A.

B — Phase B connector
specialized electrical

Specialized electrical conserving port associated with the phase B connector. If you set the **Configuration** parameter to on both sides, this port appears twice.

Dependencies

To enable this port, set **Connectors** to ABC, AB, BC, or B.

C — Phase C connector
specialized electrical

Specialized electrical conserving port associated with the phase C connector. If you set the **Configuration** parameter to on both sides, this port appears twice.

Dependencies

To enable this port, set **Connectors** to ABC, BC, AC, or C.

Parameters

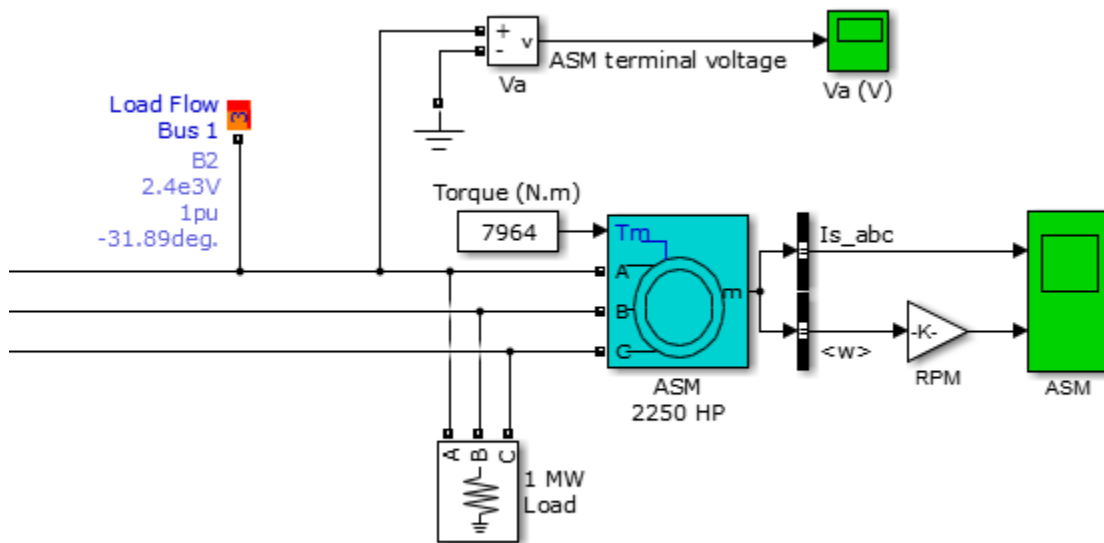
The Load Flow Bus parameters are used for model initialization only. They have no impact on the simulation performance.

Note The bus type (PV, PQ, or swing) is not defined in the mask of the Load Flow Bus block. You can have several machines and voltage source blocks with different generator type parameters connected at the same bus. The Load Flow tool determines the resulting bus type.

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Parameters**Connectors** — Block connectors
single (default) | ABC | AB | AC | BC | A | B | C

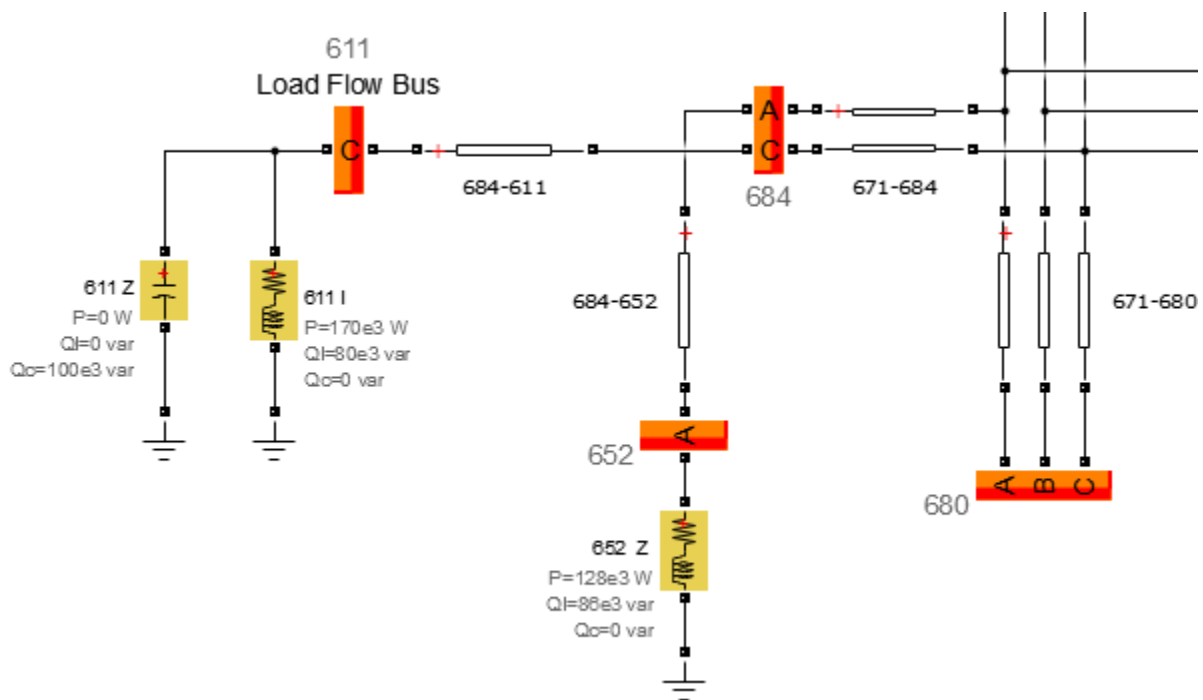
Set to single (default) when you perform a positive-sequence load flow. You connect a Load Flow Bus block to any phase (A, B, or C) of every load flow block in the model. The icon of the block displays the number 3, indicating that the block is defined to perform a positive-sequence load flow.



When you perform an unbalanced load flow, you connect a Load Flow Bus block to all phases of every load flow block in the model. Depending on the number of phases, you need to specify the appropriate Connectors parameter by selecting one of these connector combinations:

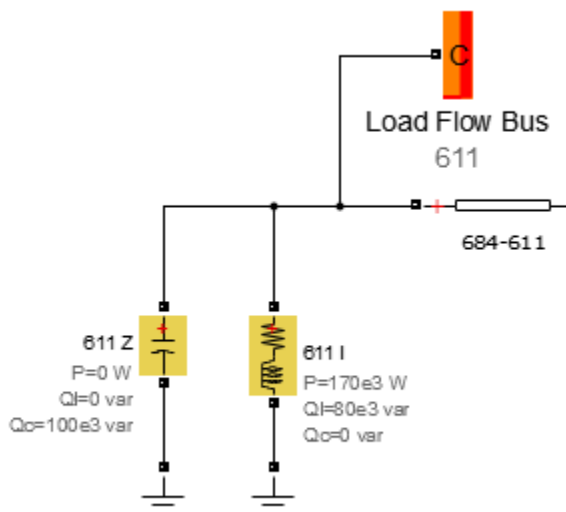
- Three connectors: ABC
- Two connectors: AB, AC, or BC
- A single connector : A, B, or C

The figure shows examples of Load Flow Bus blocks set to define different phases of a given model, to perform an unbalanced load flow.

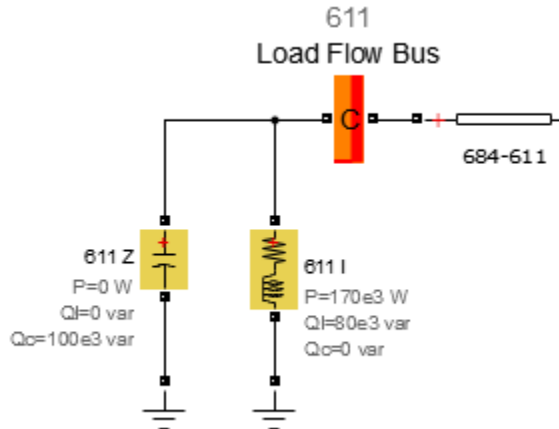


Configuration — Configuration options
on one side (default) | on both sides

When the **Configuration** parameter is set to on one side, the block displays the block phase ports only on one side of the block. You can connect the Load Flow Bus block anywhere in the model to identify a connection line as a load flow bus:



When the **Configuration** parameter is set to on both sides, the block displays the block phase ports on both sides of the block. You can connect the Load Flow Bus block in series with other Simscape Electrical Specialized Power Systems blocks to identify a connection line as a load flow bus.



Both options perform the same action, and you can seamlessly use either method to connect the Load Flow Bus blocks.

Dependencies

To enable this parameter, set **Connectors** to ABC, AB, AC, BC, A, B, or C.

Bus Identification — Bus label

BUS_1 (default) | character vector

Alphanumeric label. The **Bus Identification** label appears below the block as a block annotation.

Base voltage (Vrms phase-phase) — Base voltage (Vrms phase-phase)

25e3

Base voltage is usually the same as the nominal voltage of the load flow blocks connected to the Load Flow Bus block. The **Base voltage** values appears below the block as a block annotation.

Dependencies

To enable this parameter, set **Connectors** to single.

Swing bus or PV bus voltage (pu) — Required bus voltage magnitude

1 (default)

Required bus voltage magnitude, in pu. Depending on the bus type of the load flow blocks connected at that bus, this voltage corresponds to the swing bus voltage or the PV bus voltage.

Dependencies

To enable this parameter, set **Connectors** to single.

Swing bus voltage angle (degrees) — Swing bus voltage angle

0 (default)

Swing bus voltage angle is used as a reference to compute voltage angles of all other buses in the model.

Dependencies

To enable this parameter, set **Connectors** to single.

Base voltage (Vrms phase-ground) — Base voltage (Vrms phase-ground)
25e3 (default)

Base voltage is usually the same as the nominal voltage of the load flow blocks connected to the Load Flow Bus block. The **Base voltage** values appears below the block as a block annotation.

Dependencies

To enable this parameter, set **Connectors** to single.

Swing bus or PV bus voltage, per phase (pu) — Required bus voltage magnitude for every phase
1 (default)

Specify the required bus voltage magnitude, in pu, for every phase specified by the **Connectors** parameter. Depending on the bus type of the load flow blocks connected at that bus, this voltage corresponds to the swing bus voltage or the PV bus voltage.

Dependencies

To enable this parameter, set **Connectors** to ABC, AB, AC, BC, A, B, or C.

Swing bus voltage angle, per phase (degrees) — Required bus voltage angle for every phase
0 (default)

Required bus voltage angle, in pu, for every phase specified by the **Connectors** parameter. The swing bus voltage angles are used as a reference to compute voltage angles of all other buses in the model.

Dependencies

To enable this parameter, set **Connectors** to ABC, AB, AC, BC, A, B, or C.

Load Flow

Bus voltage section

Phase A — Resulting bus voltage of phase A
1 (default)

Displays the resulting bus voltage of phase A, in pu, once the load flow is solved. This parameter appears dimmed because it is updated by the Load Flow tool. The parameter value appears below the block as a block annotation.

The value is set to NaN by the Load Flow tool when the load flow solution is not computed for this phase (that is, when the **Connectors** parameter is set to BC, B, or C).

Phase B — Resulting bus voltage of phase B
1 (default)

Displays the resulting bus voltage of phase B, in pu, once the load flow is solved. This parameter appears dimmed because it is updated by the Load Flow tool. The parameter value appears below the block as a block annotation.

The value is set to NaN by the Load flow tool when the load flow solution is not computed for this phase (that is, when the **Connectors** parameter is set to AC, A, or C).

Dependencies

To enable this parameter, set **Connectors** to ABC, AB, AC, BC, A, B, or C.

Phase C — Resulting bus voltage of phase C

1 (default)

Displays the resulting bus voltage of phase C, in pu, once the load flow is solved. This parameter appears dimmed because it is updated by the Load Flow tool. The parameter value appears below the block as a block annotation.

The value is set to NaN by the Load flow tool when the load flow solution is not computed for this phase (that is, when the **Connectors** parameter is set to AB, A, or B).

Dependencies

To enable this parameter, set **Connectors** to ABC, AB, AC, BC, A, B, or C.

Bus angle section**Phase A — Resulting bus voltage angle of phase A**

0 (default)

Displays the resulting bus voltage angle of phase A, in pu, once the load flow is solved. This parameter appears dimmed because it is updated by the Load Flow tool. The parameter value appears below the block as a block annotation.

The value is set to NaN by the Load Flow tool when the load flow solution is not computed for this phase (that is, when the **Connectors** parameter is set to BC, B, or C).

Phase B — Resulting bus voltage angle of phase B

0 (default)

Displays the resulting bus voltage angle of phase B, in pu, once the load flow is solved. This parameter appears dimmed because it is updated by the Load Flow tool. The parameter value appears below the block as a block annotation.

The value is set to NaN by the Load flow tool when the load flow solution is not computed for this phase (that is, when the **Connectors** parameter is set to AC, A, or C).

Dependencies

To enable this parameter, set **Connectors** to ABC, AB, AC, BC, A, B, or C.

Phase C — Resulting bus voltage angle of phase C

0 (default)

Displays the resulting bus voltage angle of phase C, in pu, once the load flow is solved. This parameter appears dimmed because it is updated by the Load Flow tool. The parameter value appears below the block as a block annotation.

The value is set to NaN by the Load flow tool when the load flow solution is not computed for this phase (that is, when the **Connectors** parameter is set to AB, A, or B).

Dependencies

To enable this parameter, set **Connectors** to ABC, AB, AC, BC, A, B, or C.

Version History

Introduced in R2011a

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

`power_loadflow`

Machine Measurement Demux

Split measurement signal of machine models into separate signals



Library

Fundamental Blocks/Machines

Description

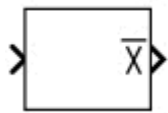
Note This block is now obsolete. We strongly recommend using the Bus Selector block from Simulink library to demux the measurement signals of the machine blocks.

Version History

Introduced before R2006a

Mean

Compute mean value of signal



Libraries:
Simscape / Electrical / Specialized Power Systems / Sensors and Measurements

Description

The Mean block computes the mean value of the input signal. The mean value is computed over a running average window of one cycle of the specified fundamental frequency:

$$Mean(f(t)) = \frac{1}{T} \int_{(t-T)}^t f(t) \cdot dt$$

$f(t)$:input signal, $T = 1/\text{fundamental frequency}$

As this block uses a running average window, one cycle of simulation must complete before the output gives the correct value. For the first cycle of simulation, the output is held constant to the specified initial input value.

Characteristics

Sample Time	Specified in the Sample Time parameter Continuous if Sample Time = 0
Scalar Expansion	Yes, of the parameters
Dimensionalized	Yes

Examples

Determine Mean Value of DC Signal with 12th Harmonic Ripple

The power_Mean model uses the Mean block to determine the mean value of a DC signal containing a 12th harmonic ripple.

The model sample time is parameterized by the Ts variable set to a default value of 50e-6 s. Set Ts to 0 in the command window to simulate the model in continuous mode.

Ports

Input

Port_1 — Input signal
Simulink signal

The input signal for which the block computes the mean value, specified as a Simulink signal.

Output

X — Mean value of input signal
Simulink signal

The mean value of the input signal, returned as a Simulink signal.

Parameters

Fundamental frequency (Hz) — Fundamental frequency
60 (default) | positive scalar

Specify the fundamental frequency, in hertz. This value determines the period ($1/F$) of the running average window.

Initial input (DC component) — Input value for first simulation cycle
0 (default) | scalar

Specify the input value for the first cycle of simulation.

Sample time — Sample time of block
0 (default) | nonnegative scalar

Specify the sample time of the block, in seconds. Set to 0 to implement a continuous block.

Version History

Introduced in R2013a

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

Mean (Phasor)

Compute mean value of input phasor over a running window of one cycle of specified frequency



Library

Simscape / Electrical / Specialized Power Systems / Sensors and Measurements

Description

The Mean (Phasor) block filters an AC sinusoidal component that is added to the DC component of a phasor signal. Filtering is performed by computing the average values of real and imaginary parts of the input phasor, using a running window of one cycle of the specified fundamental frequency.

Parameters

Fundamental frequency (Hz)

Specify the fundamental frequency, in hertz, to determine the period ($1/F$) of the running average window. Default is 60.

Initial input (DC component)

Specify the initial DC component of the input signal. If the input is a vectorized signal, specify a 1-by-N vector, where each value corresponds to a particular input. Default is 0.

Sample time

Specify the sample time of the block, in seconds. Set to 0 to implement a continuous block. Default is 0.

Characteristics

Direct Feedthrough	Yes
Sample Time	Specified in the Sample Time parameter Continuous if Sample Time = 0
Scalar Expansion	Yes, of the parameters
States	Two states per filter
Dimensionalized	Yes

Examples

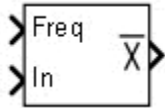
The power_PhaseMeasurements example shows the use of the Mean (Phasor) block.

Version History

Introduced in R2013a

Mean (Variable Frequency)

Compute mean value of signal



Library

Simscape / Electrical / Specialized Power Systems / Sensors and Measurements

Description

The Mean (Variable Frequency) block computes the mean value of the signal connected to the second input of the block. The mean value is computed over a running average window of one cycle of the frequency of the signal:

$$\text{Mean}(f(t)) = \frac{1}{T} \int_{(t-T)}^t f(t) \cdot dt$$

$f(t)$: input signal, $T = 1/\text{frequency}$

This block uses a running average window. Therefore, one cycle of simulation must complete before the block outputs the computed mean value. For the first cycle of simulation, the output is held constant to the specified initial value.

Parameters

Initial frequency (Hz)

Specify the frequency of the first cycle of simulation. Default is 60.

Minimum frequency (Hz)

The minimum frequency value determines the buffer size of the Variable Time Delay block used inside the block to compute the mean value. Default is 45.

Initial input (DC component)

Specify the initial value of the input during the first cycle of simulation. Default is 0.

Sample time

Specify the sample time of the block, in seconds. Set to 0 to implement a continuous block. Default is 0.

Inputs and Outputs

Freq

The frequency of the signal.

- In
Connects to the signal to be analyzed.
- X
The mean value of the signal.

Characteristics

Sample Time	Specified in the Sample Time parameter Continuous if Sample Time = 0
Scalar Expansion	Yes, of the parameters
Dimensionalized	Yes

Examples

The `power_MeanVariableFrequency` model compares the Mean block to the Mean (Variable Frequency) block for three identical input signals. It shows that, even if the frequency of the input signals varies during the simulation, the Mean (Variable Frequency) block outputs correct values.

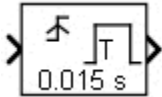
The model sample time is parameterized by the `Ts` variable with a default value of 50e-6 s. Set `Ts` to 0 in the command window to simulate the model in continuous mode.

Version History

Introduced in R2013a

Monostable

Implement monostable flip-flop (one-shot multivibrator)



Library

Simscape / Electrical / Specialized Power Systems / Control

Description

The Monostable block outputs a true pulse when a change in the logical input is detected.

The Monostable block can be programmed to detect a rising edge (when the input goes from false to true), a falling edge (when the input goes from true to false), or either edge (any change in input is detected). The Monostable block ignores any edge occurring when the pulse is true.

Parameters

Edge detection

Select the detection criteria triggering the output pulse. Choices are **Rising** (default), **Falling**, or **Either**.

Pulse duration (s)

Specify the duration of the output true state. When the **Sample time** parameter is greater than 0, the **Pulse duration** value must be a multiple of the sample time. Default is 0.015.

Initial condition of previous input

Specify the initial value of the previous input (0 or 1). The Monostable block can detect an edge at $t = 0$ if the input is different from the specified value. Default is 0.

Sample time

Specify the sample time of the block, in seconds. Set to 0 to implement a continuous block. Default is 0.

Inputs and Outputs

The input signal must be a Boolean signal. The output is Boolean.

Characteristics

Direct Feedthrough	Yes
Sample Time	Specified in the Sample Time parameter
Scalar Expansion	Yes, of parameters

Dimensionalized	Yes
Zero-Crossing Detection	Yes (in continuous-time domain)

Examples

The `power_Logic` example shows the operation of the monostable flip-flop.

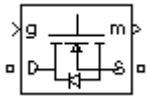
The model sample time is parameterized with the variable `Ts` (default value $Ts = 5e-3$). To simulate a continuous Monostable block, specify $Ts = 0$ in the MATLAB Command Window.

Version History

Introduced in R2013a

Mosfet

Implement MOSFET model

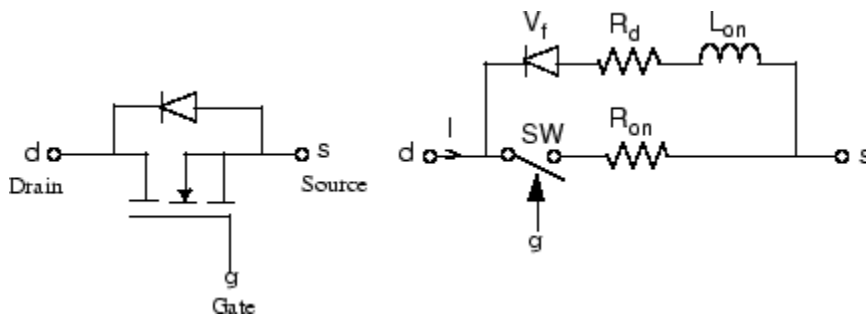


Libraries:

Simscape / Electrical / Specialized Power Systems / Power Electronics

Description

The metal-oxide-semiconductor field-effect transistor (MOSFET) is a semiconductor device controllable by the gate signal ($g > 0$). The MOSFET device is connected in parallel with an internal diode that turns on when the MOSFET device is reverse biased ($V_{ds} < 0$) and no gate signal is applied ($g=0$). The model is simulated by an ideal switch controlled by a logical signal ($g > 0$ or $g=0$), with a diode connected in parallel.



The MOSFET device turns on when a positive signal is applied at the gate input ($g > 0$) whether the drain-source voltage is positive or negative. If no signal is applied at the gate input ($g=0$), only the internal diode conducts when voltage exceeds its forward voltage V_f .

With a positive or negative current flowing through the device, the MOSFET turns off when the gate input becomes 0. If the current I is negative and flowing in the internal diode (no gate signal or $g = 0$), the switch turns off when the current I becomes 0.

The on state voltage V_{ds} varies:

- $V_{ds} = R_{on} \cdot I$ when a positive signal is applied at the gate input.
- $V_{ds} = R_d \cdot I - V_f + L_{on} \cdot dI/dt$ when the antiparallel diode is conducting (no gate signal).

The L_{on} diode inductance is available only with the continuous model. For most applications, L_{on} should be set to zero for both continuous and discrete models.

The MOSFET block also contains a series R_s - C_s snubber circuit that can be connected in parallel with the MOSFET (between nodes d and s).

Assumptions and Limitations

The MOSFET block implements a macro model of the real MOSFET device. It does not take into account either the geometry of the device or the complex physical processes [1].

Depending on the value of the inductance **Lon**, the MOSFET is modeled either as a current source ($L_{on} > 0$) or as a variable topology circuit ($L_{on} = 0$). The MOSFET block cannot be connected in series with an inductor, a current source, or an open circuit, unless its snubber circuit is in use.

The inductance **Lon** is forced to 0 if you choose to discretize your circuit.

Ports

Input

g — Control signal
Simulink signal

Simulink signal to control the opening and closing of the MOSFET.

Output

m — Measurement vector
Simulink signal

Simulink output of the block is a vector containing 2 signals. You can demultiplex these signals by using the Bus Selector block provided in the Simulink library.

Signal	Definition	Units
1	MOSFET current	A
2	MOSFET voltage	V

Conserving

D — Drain
specialized electrical

Specialized electrical conserving port associated with the drain.

S — Source
specialized electrical

Specialized electrical conserving port associated with the source.

Parameters

FET resistance Ron (Ohms) — Internal resistance
0.1 (default)

The internal resistance **Ron**, in ohms (Ω). Default is 0.1. The **Resistance Ron** parameter cannot be set to 0 when the **Inductance Lon** parameter is set to 0.

Internal diode inductance Lon (H) — Internal inductance
0 (default)

The internal inductance **Lon**, in henries (H). Default is 0. The **Inductance Lon** parameter is normally set to 0 except when the **Resistance Ron** parameter is set to 0.

Internal diode resistance Rd (Ohms) — Internal resistance of internal diode
0.01 (default)

The internal resistance of the internal diode, in ohms (Ω). Default is 0.01.

Internal diode forward voltage Vf (V) — Forward voltage of internal diode
0 (default)

The forward voltage of the internal diode, in volts (V). Default is 0.

Initial current Ic (A) — MOSFET initial current
0 (default)

You can specify an initial current flowing in the MOSFET device. It is usually set to 0 in order to start the simulation with the device blocked. Default is 0.

If the **Initial current IC** parameter is set to a value greater than 0, the steady-state calculation considers the initial status of the MOSFET as closed. Initializing all states of a power electronic converter is a complex task. Therefore, this option is useful only with simple circuits.

Snubber resistance Rs (Ohms) — Snubber resistance
1e5 (default)

The snubber resistance, in ohms (Ω). Default is 1e5. Set the **Snubber resistance Rs** parameter to `inf` to eliminate the snubber from the model.

Snubber capacitance Cs (F) — Snubber capacitance
`inf` (default)

The snubber capacitance, in farads (F). Default is `inf`. Set the **Snubber capacitance Cs** parameter to 0 to eliminate the snubber, or to `inf` to get a resistive snubber.

Show measurement port — Measurement port visibility
`on` (default) | `off`

If selected, add a Simulink output to the block returning the MOSFET current and voltage. Default is selected.

Version History

Introduced before R2006a

References

[1] Mohan, N., T.M. Undeland, and W.P. Robbins, *Power Electronics: Converters, Applications, and Design*, John Wiley & Sons, Inc., New York, 1995.

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

Diode | GTO | Ideal Switch | Three-Level Bridge | Thyristor | Universal Bridge

Multiband Power System Stabilizer

Implement multiband power system stabilizer



Libraries:

Simscape / Electrical / Specialized Power Systems / Electrical Machines / Synchronous Machine Control

Description

Note This block requires that you have a Control System Toolbox license. Otherwise, trying to simulate a model containing this block produces an error.

The disturbances occurring in a power system induce electromechanical oscillations of the electrical generators. These oscillations, also called power swings, must be effectively damped to maintain the system's stability. Electromechanical oscillations can be classified in four main categories:

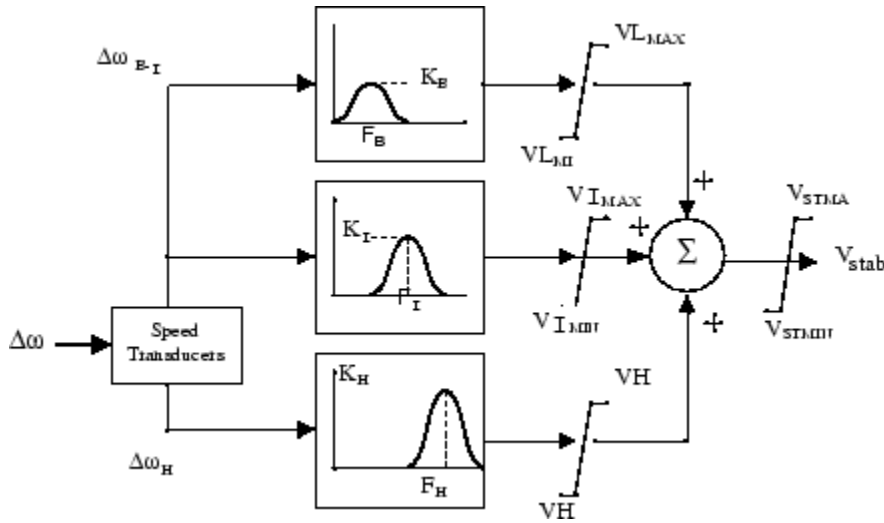
- Local oscillations: between a unit and the rest of the generating station and between the latter and the rest of the power system. Their frequencies typically range from 0.8 Hz to 4.0 Hz.
- Interplant oscillations: between two electrically close generation plants. Frequencies can vary from 1 Hz to 2 Hz.
- Interarea oscillations: between two major groups of generation plants. Frequencies are typically in a range of 0.2 Hz to 0.8 Hz.
- Global oscillation: characterized by a common in-phase oscillation of all generators as found on an isolated system. The frequency of such a global mode is typically under 0.2 Hz.

The need for effective damping of such a wide range, almost two decades, of electromechanical oscillations motivated the concept of the multiband power system stabilizer (MB-PSS).

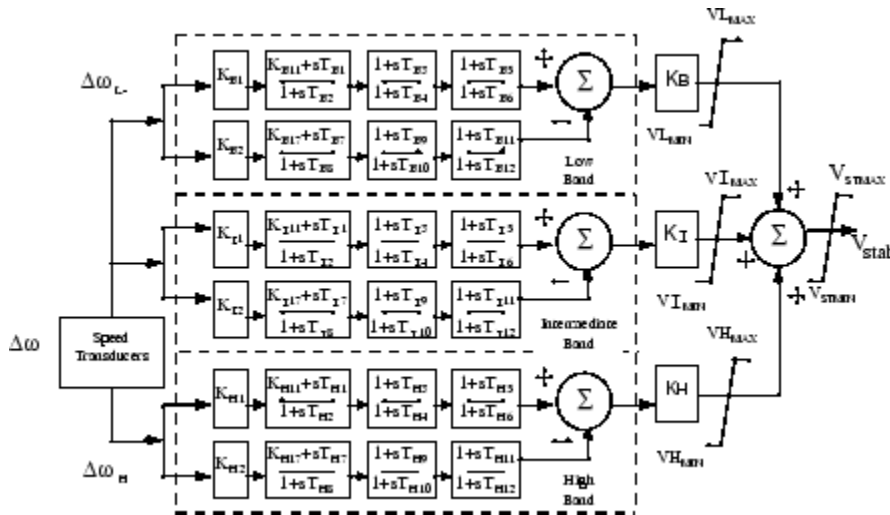
As its name reveals, the MB-PSS structure is based on multiple working bands. Three separate bands are used, respectively dedicated to the low-, intermediate-, and high-frequency modes of oscillations: the low band is typically associated with the power system global mode, the intermediate with the interarea modes, and the high with the local modes.

Each of the three bands is made of a differential bandpass filter, a gain, and a limiter (see the figure called “Conceptual Representation” on page 1-255). The outputs of the three bands are summed and passed through a final limiter producing the stabilizer output V_{stab} . This signal then modulates the set point of the generator voltage regulator so as to improve the damping of the electromechanical oscillations.

To ensure robust damping, the MB-PSS should include a moderate phase advance at all frequencies of interest to compensate for the inherent lag between the field excitation and the electrical torque induced by the MB-PSS action.



Conceptual Representation



Internal Specifications

The MB-PSS is represented by the IEEE St. 421.5 PSS 4B type model [2], illustrated in the figure called "Internal Specifications" on page 1-255, with built-in speed transducers whose parameters are fixed according to manufacturer's specifications.

Generally, only a few of the lead-lag blocks in this figure should be used in a given PSS application. Two different approaches are available to configure the settings in order to facilitate the tuning process:

1 Simplified settings:

Only the first lead-lag block of each frequency band is used to tune the Multiband Power System Stabilizer block. The differential filters are assumed to be symmetrical bandpass filters respectively tuned at the center frequency F_L , F_I , and F_H . The peak magnitude of the frequency responses (see the figure called "Conceptual Representation" on page 1-255) can be adjusted

independently through the three gains K_L , K_I , and K_H . Only six parameters are therefore required for a simplified tuning of the MB-PSS.

2 Detailed settings:

The designer is free to use all the flexibility built into the MB-PSS structure to achieve nontrivial controller schemes and to tackle even the most constrained problem (for example, multi-unit plant including an intermachine mode, in addition to a local mode and multiple interarea modes). In this case, all the time constants and gains appearing in the figure called “Internal Specifications” on page 1-255 have to be specified in the dialog box.

Ports

Input

dw — Synchronous machine speed deviation or acceleration power
scalar

Synchronous machine speed deviation dw signal (in pu).

Output

Vstab — Stabilization voltage
scalar

Stabilization voltage, in per unit. Connect this output port to the **Vstab** input port of the Excitation System block that you use to control the terminal voltage of the synchronous machine.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Mode of operation — Mode of operation
Simplified settings (default) | Detailed settings

Specify the mode of operation of the stabilizer.

Global gain — Gain of multiband power system stabilizer
1.0 (default)

Overall gain K of the multiband power system stabilizer.

Dependencies

To enable this parameter, set **Mode of operation** to Simplified settings.

Low frequency band: [FL KL] — Center frequency and peak gain of low-frequency bandpass filter
[0.2 20] (default) | vector of two scalars

Center frequency, in hertz, and peak gain of the low-frequency bandpass filter.

Dependencies

To enable this parameter, set **Mode of operation** to Simplified settings.

Intermediate frequency band: [FI KI] — Center frequency and peak gain of intermediate-frequency bandpass filter

[0.9 25] (default) | vector of two scalars

Center frequency, in hertz, and peak gain of the intermediate- frequency bandpass filter.

Dependencies

To enable this parameter, set **Mode of operation** to Simplified settings.

High frequency band: [FH KH] — Center frequency and peak gain of high-frequency bandpass filter

[12.0 145] (default) | vector of two scalars

Center frequency, in hertz, and peak gain of the high-frequency bandpass filter.

Dependencies

To enable this parameter, set **Mode of operation** to Simplified settings.

Low frequency gains: [KL1 KL2 KL] — Low frequency gains

[66 66 9.4] (default) | vector of three scalars

Gains of the positive and negative branches of the differential filter in the low-frequency band and the overall gain K_L of the low-frequency band, in pu.

Dependencies

To enable this parameter, set **Mode of operation** to Detailed settings.

Low frequency time constants — Low frequency time constants

[1.667 2 0 0 0 0 2 2.4 0 0 0 0 1 1] (default) | vector of 14 nonnegative scalars

Time constants, in seconds, of the lead-lag blocks in the positive and negative branches of the low-frequency filter. You must specify these twelve time constants and two gains:

[T_{B1} T_{B2} T_{B3} T_{B4} T_{B5} T_{B6} T_{B7} T_{B8} T_{B9} T_{B10} T_{B11} T_{B12} K_{B11} K_{B17}]

Set K_{B11} to 0 in order to make the first block of the positive filter branch a washout block. Set K_{B11} to 1 in order to make the block a lead-lag block.

Set K_{B17} to 0 in order to make the first block of the negative filter branch a washout block. Set K_{B17} to 1 in order to make the block a lead-lag block.

Dependencies

To enable this parameter, set **Mode of operation** to Detailed settings.

Intermediate frequency gains: [KI1 KI2 KI] — Intermediate frequency gains

[66 66 47.6] (default) | vector of three scalars

Gains of the positive and negative branches of the differential filter in the intermediate-frequency band and the overall gain K_I of the intermediate-frequency band, in pu.

Dependencies

To enable this parameter, set **Mode of operation** to Detailed settings.

Intermediate frequency time constants — Intermediate frequency time constants

[1 1 0.25 0.3 0 0 1 1 0.3 0.36 0 0 0 0] (default) | vector of 14 nonnegative scalars

Time constants, in seconds, of the lead-lag blocks in the positive and negative branches of the intermediate-frequency filter. You must specify these twelve time constants and two gains:

[T_{I1} T_{I2} T_{I3} T_{I4} T_{I5} T_{I6} T_{I7} T_{I8} T_{I9} T_{I10} T_{I11} T_{I12} K_{I11} K_{I17}]

Set K_{I11} to 0 in order to make the first block of the positive filter branch a washout block. Set K_{I11} to 1 in order to make the block a lead-lag block.

Set K_{I17} to 0 in order to make the first block of the negative filter branch a washout block. Set K_{I17} to 1 in order to make the block a lead-lag block.

Dependencies

To enable this parameter, set **Mode of operation** to Detailed settings.

High frequency gains: [KH1 KH2 KH] — High frequency gains

[66 66 233] (default) | vector of three scalars

Gains of the positive and negative branches of the differential filter in the high-frequency band and the overall gain K_I of the high-frequency band, in pu.

Dependencies

To enable this parameter, set **Mode of operation** to Detailed settings.

High frequency time constants — High frequency time constants

[0.01 0.012 0 0 0 0 0.012 0.0144 0 0 0 0 1 1] (default) | vector of 14 nonnegative scalars

Time constants, in seconds, of the lead-lag blocks in the positive and negative branches of the high-frequency filter. You must specify these twelve time constants and two gains:

[T_{H1} T_{H2} T_{H3} T_{H4} T_{H5} T_{H6} T_{H7} T_{H8} T_{H9} T_{H10} T_{H11} T_{H12} K_{H11} K_{H17}]

Set K_{H11} to 0 in order to make the first block of the positive filter branch a washout block. Set K_{H11} to 1 in order to make the block a lead-lag block.

Set K_{H17} to 0 in order to make the first block of the negative filter branch a washout block. Set K_{H17} to 1 in order to make the block a lead-lag block.

Dependencies

To enable this parameter, set **Mode of operation** to Detailed settings.

Signal limits [VLmax VImax VHmax VSmax] — Signal limits

[.075 .15 .15 .15] (default) | vector of four scalars

Limits imposed on the output of the low-, intermediate-, and high-frequency bands and the limit VS_{max} imposed on the output of the stabilizer, all in pu.

Plot frequency response — Plot frequency response

off (default) | on

Select this option to display a plot of the frequency response of the stabilizer when you click the **Apply** button.

Magnitude in dB — Plot magnitude of frequency response

on (default) | off

Select this option to plot the magnitude of the frequency response to plot, in dB.

Dependencies

To enable this parameter, select the **Plot frequency response** parameter.

Frequency range — Frequency range

logspace(-2,2,500) (default)

Frequency range used to plot the frequency response of the stabilizer.

Dependencies

To enable this parameter, select the **Plot frequency response** parameter.

Version History

Introduced before R2006a

References

- [1] Grondin, R., I. Kamwa, L. Soulieres, J. Potvin, and R. Champagne, "An approach to PSS design for transient stability improvement through supplementary damping of the common low frequency," *IEEE Transactions on Power Systems*, 8(3), August 1993, pp. 954-963.
- [2] IEEE recommended practice for excitation system models for power system stability studies: *IEEE St. 421.5-2002*(Section 9).

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

Generic Power System Stabilizer

Multimeter

Measure voltages and currents specified in dialog boxes of Simscape Electrical Specialized Power Systems blocks

**Libraries:**

Simscape / Electrical / Specialized Power Systems / Sensors and Measurements

Description

This block measures the voltages and currents specified in the **Measurements** parameter of Simscape Electrical Specialized Power Systems blocks in your model. Choosing voltages or currents through the Multimeter block is equivalent to connecting an internal voltage or current measurement block inside your blocks.

Polarity of Measured Voltages and Currents

The polarity of the measured voltages and current is determined by the + and - terminals of the measured Simscape Electrical Specialized Power Systems blocks. The following table identifies the positive terminals of the Simscape Electrical Specialized Power Systems blocks that do not have the + sign to identify block polarity.

Block	Positive terminal(s)
Mutual Inductance	1 2 3
Three-Phase Mutual Inductance Z1-Z0	ABC
Three-Phase Series RLC Branch	ABC
Three-Phase Parallel RLC Branch	ABC
Three-Phase Breaker	ABC
Breaker	1

For three-phase transformers, the voltage polarities and positive current directions are indicated by the signal labels used in the Multimeter block. For example, U_{an_w2} = phase A-to-neutral voltage of the Y connected winding #2 and I_{ab_w1} = winding current flowing from A to B in the delta-connected winding #1.

The Multimeter block is not designed to be used in a model that contains more than one powergui block.

Examples

The `power_compensated` example uses a Multimeter block to measure the voltage across the secondary winding of a Saturable Transformer block and the currents flowing through two Series RLC Load blocks.

The output of the Multimeter block is connected to a Scope block in order to display the measurements during the simulation. In addition, you can select the **Plot Output Measurements at simulation stop** parameter to display a plot of selected measurements when simulation stops.

Open the Multimeter block dialog box and select the signals you want to observe, as described in the Parameters section. Notice the labels used to define the available measurements in the Multimeter block. For a description of these labels, see the reference section of the Saturable Transformer block and Series RLC Load block.

Start the simulation. After 0.4 seconds, the simulation stops and a MATLAB figure window opens to display the selected measurements in the Multimeter block.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Available Measurements — Available measurements in block

Available measurements in the Multimeter block. Click **Select** to select measurements from the list. When you select a measurement, the measurement moves from the **Available Measurements** list to the **Output Measurements** list. Click **Refresh** to refresh the list of available measurements.

The measurements in the list are identified by the name of the block where the measurement is done. The type of measurement (voltage measurement, current measurement, or flux) is defined by a label preceding the block name. For more information, see the **Measurements** parameter description in the Parameters section of blocks in your model.

Output Measurements — Measurements output by block

Measurements output by the block. You can reorder the measurements by using the **Up**, **Down**, and **Remove** buttons. The **+/-** button allows you to reverse the polarity of a selected measurement.

Plot Output Measurements at simulation — Option to plot output measurements at simulation
off (default) | on

Select to display a plot of selected measurements in a MATLAB figure window. The plot is generated when the simulation stops.

Output type — Format of output signals
Complex (default) | Real-ImagMagnitude-Angle | Magnitude

Specifies the format of the output signals when the block is used in a phasor simulation.

Set to **Complex** (default) to output the selected measurements as complex values. The outputs are complex signals.

Set to **Real - Imag** to output the real and imaginary parts of the measurements. For each selected measurement, the multimeter outputs the real and imaginary parts.

Set to **Magnitude-Angle** to output the magnitude and angle of the selected measurements. For each selected measurement, the multimeter outputs the magnitude and angle values.

Set to **Magnitude** to output the magnitude of the selected measurements.

Dependencies

To enable this parameter, in the Powergui block, set the **Solver type** parameter to **Phasor**.

Version History

Introduced before R2006a

Extended Capabilities

C/C++ Code Generation

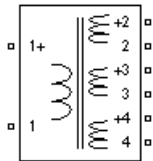
Generate C and C++ code using Simulink® Coder™.

See Also

[Current Measurement](#) | [Voltage Measurement](#)

Multi-Winding Transformer

Implement multi-winding transformer with taps



Libraries:

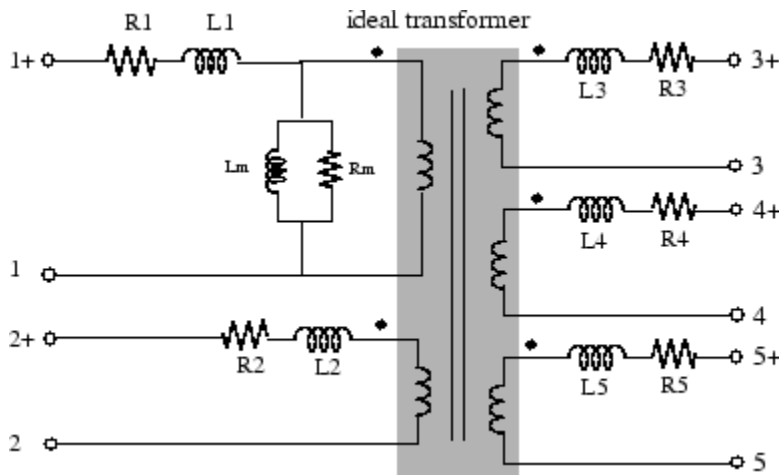
Simscape / Electrical / Specialized Power Systems / Power Grid Elements

Description

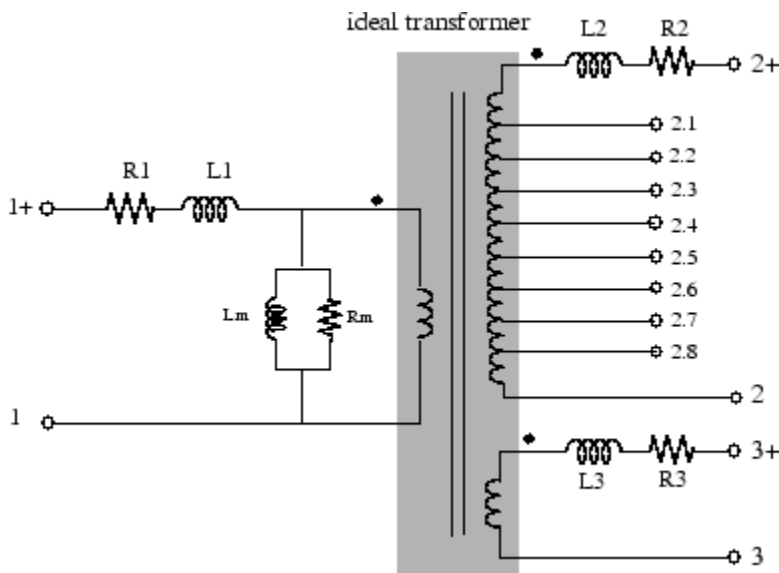
The Multi-Winding Transformer block implements a transformer where the number of windings can be specified for both the primary (left side windings) and the secondary (right side windings).

The equivalent circuit of the Multi-Winding Transformer block is similar to the one of the Linear Transformer blocks and the saturation characteristic of the core can be specified or not. See the Saturable Transformer block reference pages for more details on how the saturation and the hysteresis characteristic are implemented.

The equivalent circuit of a Multi-Windings Transformer block with two primary windings and three secondary windings is shown in the next figure.



You can add equally spaced taps to the first primary winding (the upper-left winding) or to the first secondary winding (the upper-right winding). The equivalent circuit of a Multi-Winding Transformer block with one primary winding and eight taps on the first of the two secondary windings is shown in the next figure.



The winding terminals are identified by the corresponding winding number. The first winding is the first one on the primary side (upper-left side) and the last winding is the last one on the secondary side (bottom-right side). The polarities of the windings are defined by a plus sign.

The tap terminals are identified by their winding number followed by a dot character and the tap number. Taps are equally spaced so that voltage appearing at no load between two consecutive taps is equal to the total voltage of the winding divided by (number of taps + 1). The total winding resistance and leakage inductance of a tapped winding is equally distributed along the taps.

The “OLTC Regulating Transformer Phasor Model” example uses three Multi-Winding Transformer blocks to implement a three-phase On Load Tap Changer (OLTC) transformer.

Ports

Conserving

1+ — Primary winding positive polarity

specialized electrical

Specialized electrical conserving port associated with the primary winding positive polarity.

1 — Primary winding negative polarity

specialized electrical

Specialized electrical conserving port associated with the primary winding negative polarity.

2+ — Secondary winding positive polarity

specialized electrical

Specialized electrical conserving port associated with the secondary winding positive polarity.

2 — Secondary winding negative polarity

specialized electrical

Specialized electrical conserving port associated with the secondary winding negative polarity.

n+... — n^{th} winding positive polarity
specialized electrical

Specialized electrical conserving port associated with the n^{th} winding positive polarity.

Dependencies

To enable this parameter, set the **Number of windings on the left side** or **Number of windings on the right side** parameters to a value larger than 1.

n... — n^{th} winding negative polarity
specialized electrical

Specialized electrical conserving port associated with the n^{th} winding negative polarity.

Dependencies

To enable this parameter, set the **Number of windings on the left side** or **Number of windings on the right side** parameters to a value larger than 1.

m.1... — Tap on m^{th} winding
specialized electrical

Specialized electrical conserving port associated with a tap on the n^{th} winding. The label of these ports depend on the value of the **Tapped winding** and **Number of taps (equally spaced)** parameters.

Dependencies

To enable this parameter, set the **Number of windings on the left side** or **Number of windings on the right side** parameters to a value larger than 1.

Parameters

Configuration Tab

Number of windings on left side — Number of windings on transformer primary side
1 (default) | positive integer

Specifies the number of windings on the primary side (left side) of the transformer.

Number of windings on right side — Number of windings on transformer secondary side
3 (default) | positive integer

Specifies the number of windings on the secondary side (right side) of the transformer.

Tapped winding — Specify which winding to tap
no taps (default) | taps on upper left winding | taps on upper right winding

Select **no taps** if you don't want to add taps to the transformer. Select **taps on upper left winding** to add taps to the first winding on the primary side of the transformer. Select **taps on upper right winding** to add taps to the secondary winding on the right side of the transformer. The number of taps is specified by the **Number of taps (equally spaced)** parameter.

Number of taps (equally spaced) — Number of taps on tapped winding
2 (default) | positive integer

If the **Tapped winding** parameter is set to `taps on upper left winding`, you specify the number of taps to add to the first winding on the left side.

If the **Tapped winding** parameter is set to `taps on upper right winding`, you specify the number of taps to add to the first winding on the right side.

Dependencies

To enable this parameter, set **Tapped winding** to a value other than `no taps`.

Saturable core — Option to model saturable transformer

`off` (default) | `on`

If selected, implements a saturable transformer. See also the **Saturation characteristic** parameter on the Parameters tab.

Simulate hysteresis — Whether to model hysteresis

`off` (default) | `on`

Select to model hysteresis saturation characteristic instead of a single-valued saturation curve.

Hysteresis Mat file — Specify file for hysteresis

`'hysteresis.mat'` (default) | `string`

Specify a `.mat` file containing the data to be used for the hysteresis model. When you open the **Hysteresis Design Tool** of the Powergui, the default hysteresis loop and parameters saved in the `hysteresis.mat` file are displayed. Use the **Load** button of the Hysteresis Design tool to load another `.mat` file. Use the **Save** button of the Hysteresis Design tool to save your model in a new `.mat` file.

Dependencies

To enable this parameter, select **Simulate hysteresis**.

Measurements — Specify what to measure

`None` (default) | `Winding voltages` | `Winding currents` | `Flux and excitation current (Im + IRm)` | `Flux and magnetization current (Im)` | `All measurement (V, I, Flux)`

Select `Winding voltages` to measure the voltage across the winding terminals of the Saturable Transformer block.

Select `Winding currents` to measure the current flowing through the windings of the Saturable Transformer block.

Select `Flux and excitation current (Im + IRm)` to measure the flux linkage, in volt seconds (Vs), and the total excitation current including iron losses modeled by `Rm`.

Select `Flux and magnetization current (Im)` to measure the flux linkage, in volt seconds (Vs), and the magnetization current, in amperes (A), not including iron losses modeled by `Rm`.

Select `All measurement (V, I, Flux)` to measure the winding voltages, currents, magnetization currents, and the flux linkage.

Place a Multimeter block in your model to display the selected measurements during the simulation.

In the **Available Measurements** list box of the Multimeter block, the measurements are identified by a label followed by the block name.

Measurement	Label
Winding voltages	Uw1:
Winding currents	Iw1:
Excitation current	Iexc:
Magnetization current	Imag:
Flux linkage	Flux:

Dependencies

Parameters Tab

Units — Units

pu (default) | SI

Specify the units used to enter the parameters of the block.

Set to pu to use per unit.

Set to SI to use SI units.

Changing the **Units** parameter from pu to SI or from SI to pu automatically converts the parameters displayed in the mask of the block. The per unit conversion is based on the transformer rated power Pn in VA, nominal frequency fn in Hz, and nominal voltage Vn in Vrms, of the windings.

Nominal power and frequency [Pn(VA) fn(Hz)] — Nominal power and frequency [75e3 60] (default)

The nominal power rating Pn in VA and frequency fn in Hz, of the transformer.

This parameter does not impact the transformer model when you set the **Units** parameter to SI.

Winding nominal voltages [U1 U2 ... Un] (Vrms) — Vector of winding voltage [14400 120 120 120] (default) | vector

Specify a vector containing the nominal RMS voltages, in Vrms, of the windings on the left side, followed by the nominal RMS voltages of the windings on the right side. You don't have to specify the individual tap nominal voltages.

Dependencies

Winding resistances [R1 R2 ... Rn] (Ohm) — Vector of winding resistance values [0.005 0.005 0.005 0.005] (default) | [13.824 0.00096 0.00096 0.00096]

Specify a vector containing the resistance values of the windings on the left side, followed by the resistance values of the windings on the right side. You don't have to specify the individual tap resistances. Default is [0.005 0.005 0.005 0.005] when the **Units** parameter is pu and [13.824 0.00096 0.00096 0.00096] when the **Units** parameter is SI.

Dependencies

Winding leakage inductances [L1 L2 ... Ln] (H) — Vector of winding inductance values [0.02 0.02 0.02 0.02] (default) | [0.14668 1.0186e-05 1.0186e-05 1.0186e-05]

Specify a vector containing the leakage inductance values of the windings on the left side, followed by the leakage inductance values of windings on the right side. You don't have to specify the

individual tap leakage inductances. Default is [0.02 0.02 0.02 0.02] when the **Units** parameter is pu and [0.14668 1.0186e-05 1.0186e-05 1.0186e-05] when the **Units** parameter is SI.

Dependencies

Magnetization resistance Rm (Ohm) — Magnetization resistance

50 (default) | 1.3824e+05

The magnetization resistance Rm, in ohms or in pu. Default is 50 when the **Units** parameter is pu and 1.3824e+05 when the **Units** parameter is SI.

Dependencies

Magnetization inductance Lm (H) — Magnetization inductance for nonsaturable core

50 (default) | 366.69

The magnetization inductance Lm, in Henry or in pu, for a nonsaturable core. Default is 50 when the **Units** parameter is pu and 366.69 when the **Units** parameter is SI.

Dependencies

To enable this parameter, clear the **Saturable core** parameter.

Saturation characteristic [i1(A) , phi1(V.s) ; i2 , phi2 ; ...] — Saturation characteristic for saturable core

[0,0 ; 0.0024,1.2 ; 1.0,1.52] (default) | [0 0;0.017678 64.823;7.3657 82.109]

The saturation characteristic for the saturable core. Specify a series of current/ flux pairs (in pu) starting with the pair (0,0). Default is [0,0 ; 0.0024,1.2 ; 1.0,1.52] when the **Units** parameter is pu and [0 0;0.017678 64.823;7.3657 82.109] when the **Units** parameter is SI.

Dependencies

To enable this parameter, select **Saturable core**.

Specify initial flux — Option to define the initial flux

off (default) | on

Select to define the initial flux with the **Initial flux phi0 (pu)** parameter.

When you clear this parameter, the block automatically computes the initial flux required to start the simulation in steady state. The computed value is saved in the **Initial flux phi0 (pu)** parameter and overwrites any previous value.

Dependencies

To enable this parameter, select **Saturable core**.

Initial flux phi0 (V.s) — Initial flux of transformer

0 (default) | positive scalar

Initial flux of the transformer.

When you clear **Specify initial flux** parameter, the block automatically computes the initial flux required to start the simulation in steady state. The computed value is saved in the **Initial flux phi0 (pu)** parameter and overwrites any previous value.

Dependencies

To enable this parameter, select **Saturable core** and **Specify initial flux**.

Advanced Tab

Break Algebraic loop in discrete saturation model — Option to break algebraic loop

off (default) | on

When selected, a delay is inserted at the output of the saturation model computing magnetization current as a function of flux linkage (the integral of input voltage computed by a trapezoidal method). This delay eliminates the algebraic loop resulting from trapezoidal discretization methods and speeds up the simulation of the model. However, the delay introduces a one simulation step time delay in the model and can cause numerical oscillations if the sample time is too large. The algebraic loop is required in most cases to get an accurate solution.

When cleared (default), the discretization method of the saturation model is specified by the **Discrete solver model** parameter.

Dependencies

To enable this parameter, add a powergui block to your model, set the **Simulation type** parameter of the powergui block to **Discrete**, and clear the **Automatically handle discrete solver** parameter of the powergui block. Also, in the Multi-Winding Transformer block, select **Saturable core** parameter.

Discrete solver model — Method to resolve algebraic loop

Trapezoidal iterative (default) | Trapezoidal robust | Backward Euler robust

Select one of these methods to resolve the algebraic loop.

- **Trapezoidal iterative**—Although this method produces correct results, it is not recommended because Simulink tends to slow down and may fail to converge (simulation stops), especially when the number of saturable transformers is increased. Also, because of the Simulink algebraic loop constraint, this method cannot be used in real time. In R2018b and previous releases, you used this method when the **Break Algebraic loop in discrete saturation model** parameter was cleared.
- **Trapezoidal robust**—This method is slightly more accurate than the **Backward Euler robust** method. However, it may produce slightly damped numerical oscillations on transformer voltages when the transformer is at no load.
- **Backward Euler robust**—This method provides good accuracy and prevents oscillations when the transformer is at no load.

The maximum number of iterations for the robust methods is specified in the **Preferences** tab of the powergui block, in the **Solver details for nonlinear elements** section. For real time applications, you may need to limit the number of iterations. Usually, limiting the number of iterations to 2 produces acceptable results. The two robust solvers are the recommended methods for discretizing the saturation model of the transformer.

For more information on what method to use in your application, see “Simulating Discretized Electrical Systems”.

Dependencies

To enable this parameter, add a powergui block to your model, set the **Simulation type** parameter of the powergui block to **Discrete**, and clear the **Automatically handle discrete solver** parameter of

the powergui block. Also, in the Multi-Winding Transformer block, select **Saturable core** and clear the **Break Algebraic loop in discrete saturation model** parameter.

Version History

Introduced before R2006a

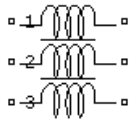
Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

Mutual Inductance

Implement inductances with mutual coupling



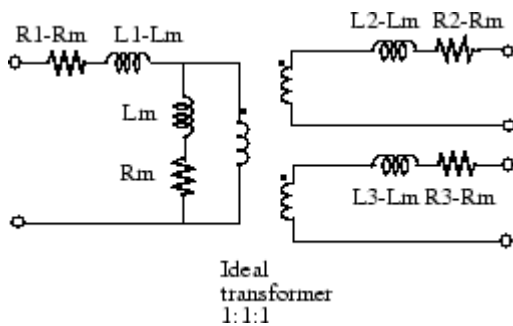
Libraries:

Simscape / Electrical / Specialized Power Systems / Passives

Description

The Mutual Inductance block can be used to model two- or three-windings inductances with equal mutual coupling, or to model a generalized multi-windings mutual inductance with balanced or unbalanced mutual coupling.

If you choose to model two- or three-windings inductances with equal mutual coupling, you specify the self-resistance and inductance of each winding plus the mutual resistance and inductance. The electrical model for this block in this case is given below:



If you choose to model a general mutual inductance, specify the number of self-windings (not just limited to 2 or 3 windings) plus the resistance and inductance matrices that define the mutual coupling relationship between the windings (balanced or not).

Equations

The resistance and inductance matrices are defined as

$$R = \begin{bmatrix} R1 & Rm \\ Rm & R2 \end{bmatrix}$$

and

$$L = \begin{bmatrix} L1 & Lm \\ Lm & L2 \end{bmatrix},$$

where:

- R is the resistance.
- $R1$ is the self-resistance of resistor $R1$.

- R_2 is the self-resistance of resistor R_2 .
- R_m is the mutual resistance, such that $R_m < R_1$ and $R_m < R_2$.
- L is the inductance.
- L_1 is the self-inductance of inductor L_1 .
- L_2 is the self-inductance of inductor L_2 .
- L_m is the mutual inductance, such that $L_m \leq \sqrt{L_1 \cdot L_2}$

Examples

The `power_mutual` example uses three coupled windings to inject a third harmonic voltage into a circuit fed at 60 Hz.

Assumptions and Limitations

- If you choose to model two or three windings inductances with equal mutual coupling, the following restrictions apply:

$$R_1, R_2, \dots, R_N \neq R_m, L_1, L_2, \dots, L_N \neq L_m. \quad (1-6)$$

- Negative values are allowed for the self- and mutual inductances as long as the self-inductances are different from the mutual inductance.
- Windings can be left floating (not connected by an impedance to the rest of the circuit). However an internal resistor between the floating winding and the main circuit is automatically added. This internal connection does not affect voltage and current measurements.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Type of mutual inductance — Type of mutual inductance

Two or three windings with equal mutual terms (default) | Generalized mutual inductance

Select `Two or Three windings with equal mutual terms` to implement a three-phase mutual inductance with equal mutual coupling between the windings. This is the default.

Select `Generalized mutual inductance` to implement a multi windings mutual inductance with mutual coupling defined by an inductance and a resistance matrix.

Dependencies

Winding 1 self impedance [R1(Ohm) L1(H)] — Self-resistance and inductance for winding 1
[1.1 1.1e-03] (default)

Self-resistance and inductance in ohms (Ω) and henries (H) for winding 1.

Dependencies

To enable this parameter, set **Type of mutual inductance** to `Two or three windings with equal mutual terms`.

Winding 2 self impedance [R2(Ohm) L2(H)] — Self-resistance and inductance for winding 2
[1.1 1.1e-03] (default)

Self-resistance and inductance in ohms (Ω) and henries (H) for winding 2.

Dependencies

To enable this parameter, set **Type of mutual inductance** to Two or three windings with equal mutual terms.

Three windings Mutual Inductance — Option to implement three-coupled windings
off (default) | on

If selected, implements three coupled windings; otherwise, it implements two coupled windings.

Dependencies

To enable this parameter, set **Type of mutual inductance** to Two or three windings with equal mutual terms.

Winding 3 self impedance [R3(Ohm) L3(H)] — Self-resistance and inductance for winding 3
[1.1 1.1e-03] (default)

Self-resistance and inductance in ohms (Ω) and henries (H) for winding 3.

Dependencies

To enable this parameter, set **Type of mutual inductance** to Two or three windings with equal mutual terms and select the **Three windings Mutual Inductance** parameter.

Mutual impedance [Rm(Ohm) Lm(H)] — Mutual resistance and inductance between windings
[1.0 1.0e-03] (default)

The mutual resistance and inductance between windings, in ohms (Ω) and henries (H). The mutual resistance and inductance corresponds to the magnetizing resistance and inductance on the standard transformer circuit diagram. If the mutual resistance and reactance are set to [0 0], the block implements three separate inductances with no mutual coupling.

The mutual inductance can be expressed as a relationship between two self inductances as

$$L_m = k \cdot \sqrt{L_1 \cdot L_2}, \quad (1-7)$$

where k is the coupling coefficient ($-1 \leq k \leq 1$).

Dependencies

To enable this parameter, set **Type of mutual inductance** to Two or three windings with equal mutual terms.

Number of windings (N) — Number of self inductances
3 (default)

Number of self inductances.

Dependencies

To enable this parameter, set **Type of mutual inductance** to Generalized mutual inductance.

Inductance matrix L (H) [N-by-N] — Inductance matrix
[1.0 0.9 0.9 ; 0.9 1.0 0.9; 0.9 0.9 1.0] * 1e-3 (default)

Inductance matrix, in Henry, that define the mutual coupling relationship between the self windings. It must be an N-by-N symmetrical matrix.

Dependencies

To enable this parameter, set **Type of mutual inductance** to Generalized mutual inductance.

Resistance matrix R (Ohm) [N-by-N] — Resistance matrix
[1.0 0.9 0.9 ; 0.9 1.0 0.9; 0.9 0.9 1.0] (default)

Resistance matrix, in ohms, that define the mutual coupling relationship between the self windings. It must be an N-by-N symmetrical matrix.

Dependencies

To enable this parameter, set **Type of mutual inductance** to Generalized mutual inductance.

Measurements — Type of measurements
None (default) | Winding voltages | Winding currents | Winding voltages and currents

Select **Winding voltages** to measure the voltage across the winding terminals.

Select **Winding currents** to measure the current flowing through the windings.

Select **Winding voltages and currents** to measure the winding voltages and currents.

Default is None.

Place a Multimeter block in your model to display the selected measurements during the simulation.

In the **Available Measurements** list box of the Multimeter block, the measurements are identified by a label followed by the block name.

Measurement	Label
Winding voltages	Uw1 :
Winding currents	Iw1 :

Version History
Introduced before R2006a

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using Simulink® Coder™.

See Also
Linear Transformer | Saturable Transformer | Three-Phase Mutual Inductance Z1-Z0

Neutral

Implement common node in circuit



Library

Simscape / Electrical / Specialized Power Systems / Utilities

Description

The Neutral block implements a common node with a specific node number. You can use this block to create a floating neutral or to interconnect two points without drawing a connection line.

Parameters

Node number

Specify a number of the neutral node. If the **Node number** parameter is set to 0, the Neutral block makes a connection to ground. The node number is displayed next to the block icon. Default is 10.

Examples

The `power_neutral` example uses three Neutral blocks. One Neutral block is used to refer a Voltage Measurement block to the ground (node 0).

See Also

Ground

Version History

Introduced before R2006a

Nonlinear Inductor

Discrete nonlinear inductor



Libraries:

Simscape / Electrical / Specialized Power Systems / Passives

Description

The Nonlinear Inductor block represents a time-varying inductor. It implements a discrete variable inductor as a current source. The inductance is specified by a monotonically increasing flux linkage-current characteristic.

When you use a Nonlinear Inductor block in your model, set the powergui block **Simulation type** to **Discrete** and select the **Automatically handle Discrete solver and Advanced tab solver settings of blocks** parameter in the **Preferences** tab. The robust discrete solver is used to discretize the electrical model. Simulink signals an error if the robust discrete solver is not used.

Equations

The block uses the following equations for the relationship between the voltage, v , across the device and the current through the inductor, i , and the flux linkage:

$$v = \frac{d(\phi)}{dt}$$

$$\phi = L * i = \text{flux linkage}.$$

Ports

Conserving

+ — Positive terminal
specialized electrical

Specialized electrical conserving port associated with the inductor positive voltage.

- — Negative terminal
specialized electrical

Specialized electrical conserving port associated with the inductor negative voltage.

Parameters

Discrete Solver — Solver type

Backward Euler (default) | Trapezoidal

Robust integration method used by the block. The discrete solver method is automatically set to Trapezoidal when, in the powergui block, in the **Preferences** settings, you select **Automatically handle discrete solver and Advanced tab solver settings of blocks**.

The Trapezoidal robust solver is slightly more accurate than the Backward Euler robust solver, especially when the model is simulated at larger sample times. The Trapezoidal robust solver may produce slight damped numerical oscillations on machine voltage in no-load conditions, while the Backward Euler robust solver prevents oscillations and maintains good accuracy.

Flux linkage - current characteristics [(V.s);(A)] — Inductance characteristic, L
 $\begin{bmatrix} 0, 0 & ; & 1, 1 \end{bmatrix}$ (default) | 2-by- n matrix

Resistance specified by a monotonically increasing flux linkage-current characteristic, specified as a 2-by- n matrix where n is greater than or equal to 2.

Initial flux linkage (V.s) — Initial flux linkage
 0 (default) | scalar

Initial flux linkage.

Measurements — Measurement type
 None (default) | Flux, Imag

Measurement type. Select Flux, Imag to measure the flux and magnetization current of the nonlinear inductor. Place a Multimeter block in your model to display the selected measurements during the simulation.

Version History

Introduced in R2019b

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

Nonlinear Resistor | Variable Capacitor | Variable Inductor | Variable Resistor | Variable-Ratio Transformer

Nonlinear Resistor

Discrete nonlinear resistor



Libraries:

Simscape / Electrical / Specialized Power Systems / Passives

Description

The Nonlinear Resistor block represents a time-varying resistor. It implements a discrete variable resistor as a current source or a voltage source. The resistance is specified by a monotonically increasing or decreasing voltage-current characteristic.

Equations

When the nonlinear resistor is implemented as a voltage source, the block uses the following equation for the relationship between the voltage, v , across the device and the current through the resistor, i , for the resistance characteristic specified by R :

$$v = R * i.$$

When the nonlinear resistor is implemented as a current source, the block uses the following equation for the relationship:

$$i = \frac{1}{R} * v.$$

When you use a Nonlinear Resistor block in your model, set the powergui block **Simulation type** to **Discrete** and select the **Automatically handle Discrete solver and Advanced tab solver settings of blocks** parameter in the **Preferences** tab. The robust discrete solver is used to discretize the electrical model. Simulink signals an error if the robust discrete solver is not used.

Ports

Conserving

+ — Positive terminal
specialized electrical

Specialized electrical conserving port associated with the resistor positive voltage.

- — Negative terminal
specialized electrical

Specialized electrical conserving port associated with the resistor negative voltage.

Parameters

Type — Source type
Current source (default) | Voltage source

Whether the nonlinear resistor is implemented as a current source or a voltage source.

Voltage-Current characteristics [(V);(A)] — Resistance characteristic, R

[0,0 ; 1,1] (default) | 2-by- n matrix

Resistance specified by a monotonically increasing or decreasing voltage-current characteristic, specified as a 2-by- n matrix where n is greater than or equal to 2.

Version History

Introduced in R2019b

Extended Capabilities

C/C++ Code Generation

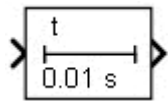
Generate C and C++ code using Simulink® Coder™.

See Also

Nonlinear Inductor | Variable Capacitor | Variable Inductor | Variable Resistor | Variable-Ratio Transformer

On/Off Delay

Implement switch-on or switch-off delay



Libraries:
Simscape / Electrical / Control / General Control

Description

The On/Off Delay block implements the following logic when the **Select type** parameter is set to **On delay**:

- When the input becomes true, the output becomes true after a preset time delay. The output remains true as long as the input is true.
- When the input is false or becomes false, the output becomes false with no delay.
- When the **Initial condition of previous input** parameter is set to 1, and the input is true at $t = 0$, the output is true with no delay.

The On/Off Delay block implements the following logic when the **Select type** parameter is set to **Off delay**:

- When the input becomes false, the output becomes false after a preset time delay. The output remains false as long as the input is false.
- When the input is true or becomes true, the output is true with no delay.
- When the **Initial condition of previous input** parameter is set to 0, and the input is false at $t = 0$, the output is false with no delay.

Characteristics

Direct Feedthrough	Yes
Sample Time	Specified in the Sample Time parameter
Scalar Expansion	Yes, of parameters
Dimensionalized	Yes
Zero-Crossing Detection	No

Examples

The `power_Logic` example shows the operation of the On/Off Delay block.

The model sample time is parameterized with the variable T_s (default value $T_s = 5e-3$). To simulate a continuous On/Off Delay block, specify $T_s = 0$ in the MATLAB Command Window.

Ports

Input

In — Delay input
boolean

Input of Delay block.

Output

Out — Delay output
boolean

Output of Delay block. The value of the output depends on the value of the **Select type** and **Initial condition of previous input** parameters.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Select type — Type of delay
On delay (default) | Off delay

Select between On delay (default for On Delay blocks) or Off delay (default for Off Delay blocks).

Time delay (s) — Duration of delay
0.01 (default) | nonnegative scalar

Duration of the delay, in seconds. When the **Sample time** parameter is greater than 0, the time delay value must be a multiple of the sample time.

Initial condition of previous input — Proportional, integral, and derivative gains of internal PID controller
0 (default) | 1 | nonnegative scalar

Initial value of the previous input. Default is 0 for On Delay blocks and 1 for Off Delay blocks.

Sample time — Block sample time
0 (default) | nonnegative scalar

Sample time of the block, in seconds. Set to 0 to implement a continuous block.

Version History

Introduced in R2013a

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

Discrete Variable Time Delay

Overmodulation

Add third harmonic or triplen harmonic zero-sequence signal to three-phase signal



Library

Simscape / Electrical / Specialized Power Systems / Power Electronics / Power Electronics Control

Description

The Overmodulation block increases the linear region of a three-phase PWM generator by adding a third harmonic or triplen harmonic zero-sequence signal V_0 to the three-phase original reference signal U_{ref} . This zero-sequence signal does not appear in the line-to-line voltages.

A modulation index of up to 1.1547 (exact value = $2/\sqrt{3}$) can be used without pulse dropping.

The Overmodulation block implements three overmodulation techniques:

- The Third Harmonic overmodulation technique. In this technique the third-harmonic signal V_0 subtracted from the original signal is calculated as

$$V_0 = \frac{|U|}{6} \times \sin[3 \cdot (wt + \angle U)]$$

- The Flat Top overmodulation technique. In this technique the portion of the three-phase input signal exceeding values ± 1 is computed. The three resulting signals are then summed and removed from the original signal U_{ref} . The resulting modified signal U_{ref}^* is therefore a flat-top three-phase signal that contains zero-sequence triplen-harmonics. The block outputs a value between -1 and 1 .
- The Min-Max overmodulation technique. In this technique the minimum and maximum values of the three components of input signal U_{ref} are summed and divided by two, and then subtracted from the input signal. The resulting modified signal U_{ref}^* also contains zero-sequence triplen-harmonics. The block outputs a value between -1 and 1 .

Parameters

Overmodulation type

Select the overmodulation technique you want to apply to the Uref signal: Third harmonic (default), Flat top, or Min-Max.

Inputs and Outputs

Uref

The three-phase reference signal of three-phase PWM generator.

Uref*

The overmodulated three-phase signal of three-phase PWM generator.

Characteristics

Sample Time	Inherited
Scalar Expansion	No
Dimensionalized	Yes

Examples

The `power_OverModulation` example compares the three overmodulation techniques implemented in the Overmodulation block. Choose the overmodulation technique (type 1, 2, or 3 on the first input of the Multiport Switch) and run the simulation. Observe the resulting waveforms in Scope 1.

The model sample time is parameterized with variable `Ts` (default value of `5e-6`). To run a continuous simulation, at the MATLAB command prompt, enter

```
Ts = 0
```

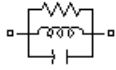
and set the **Simulation type** parameter of the Powergui block to `Continuous` before starting the simulation.

Version History

Introduced in R2013a

Parallel RLC Branch

Implement parallel RLC branch



Library

Simscape / Electrical / Specialized Power Systems / Passives

Description

The Parallel RLC Branch block implements a single resistor, inductor, and capacitor or a parallel combination of these. Use the **Branch type** parameter to select elements you want to include in the branch.

Negative values are allowed for resistance, inductance, and capacitance.

Parameters

Branch type

Select the elements you want to include in the branch. The **R** letter defines the resistor, the **L** letter defines the inductor, and the **C** letter defines the capacitor. Select **Open circuit** to define an open circuit ($R=\infty$, $L=\infty$, $C=0$). Only existing elements are displayed in the block icon. Default is RLC.

Resistance

The branch resistance, in ohms (Ω). Default is 1. The **Resistance** parameter is not visible if the resistor element is not specified in the **Branch type** parameter.

Inductance L

The branch inductance, in henries (H). Default is $1e-3$. The **Inductance** parameter is not visible if the inductor element is not specified in the **Branch type** parameter.

Set the initial inductor current

If selected, the initial inductor current is defined by the **Inductor initial current** parameter. If cleared, the software calculates the initial inductor current in order to start the simulation steady-state. Default is cleared.

The **Set the initial inductor current** parameter is not visible and has no effect on the block if the inductor element is not specified in the **Branch type** parameter.

Inductor initial current (A)

The initial inductor current used at the start of the simulation. Default is 0. This parameter is not visible and has no effect on the block if the inductor is not modeled and if the **Set the initial inductor current** parameter is not selected.

Capacitance C

The branch capacitance, in farads (F). Default is $1e-6$. The **Capacitance** parameter is not visible if the capacitance element is not specified in the **Branch type** parameter.

Set the initial capacitor voltage

If selected, the initial capacitor voltage is defined by the **Capacitor initial voltage** parameter. If cleared, the software calculates the initial capacitor voltage in order to start the simulation in steady-state. Default is cleared.

The **Set the initial capacitor voltage** parameter is not visible and has no effect on the block if the capacitor element is not specified in the **Branch type** parameter.

Capacitor initial voltage (V)

The initial capacitor voltage used at the start of the simulation. The **Capacitor initial voltage** parameter is not visible and has no effect on the block if the capacitor is not modeled and if the **Set the initial capacitor voltage** parameter is not selected.

Measurements

Select **Branch voltage** to measure the voltage across the Parallel RLC Branch block terminals.

Select **Branch current** to measure the total current (sum of R, L, C currents) flowing through the Parallel RLC Branch block.

Select **Branch voltage** and **current** to measure the voltage and the current of the Parallel RLC Branch block.

Default is None.

Place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurements** list box of the Multimeter block, the measurement is identified by a label followed by the block name.

Measurement	Label
Branch voltage	Ub :
Branch current	Ib :

Examples

The `power_parallelbranch` example is used to obtain the frequency response of an eleventh-harmonic filter (tuned frequency at 660 Hz) connected on a 60 Hz power system:

The network impedance in the Laplace domain is

$$Z(s) = \frac{V(s)}{I(s)} = \frac{RLCs^2 + Ls + R}{LCs^2 + RCs}.$$

To obtain the frequency response of the impedance you have to get the state-space model (A B C D matrices) of the system.

This system is a one input (Is) and one output (Vs) system.

Note If you have Control System Toolbox software installed, you can get the transfer function $Z(s)$ from the state-space matrices and the `bode` function.

```
[A,B,C,D] = power_analyze('power_parallelbranch');
freq = logspace(1,4,500);
```



```
w = 2*pi*freq;  
[Zmag,Zphase] = bode(A,B,C,D,1,w);  
subplot(2,1,1)  
loglog(freq,Zmag)  
grid  
title('11th harmonic filter')  
xlabel('Frequency, Hz')  
ylabel('Impedance Z')  
subplot(2,1,2)  
semilogx(freq,Zphase)  
xlabel('Frequency, Hz')  
ylabel('phase Z')  
grid
```

You can also use the Impedance Measurement block and the Powergui block to plot the impedance as a function of frequency.

See Also

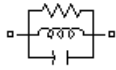
Multimeter, Parallel RLC Load, powergui, Series RLC Branch, Series RLC Load

Version History

Introduced before R2006a

Parallel RLC Load

Implement linear parallel RLC load



Library

Simscape / Electrical / Specialized Power Systems / Passives

Description

The Parallel RLC Load block implements a linear load as a parallel combination of RLC elements. At the specified frequency, the load exhibits a constant impedance. The active and reactive powers absorbed by the load are proportional to the square of the applied voltage.

Only elements associated with nonzero powers are displayed in the block icon.

Parameters

Parameters Tab

Nominal voltage Vn

The nominal voltage of the load, in volts RMS (Vrms). Default is 1000.

Nominal frequency fn

The nominal frequency, in hertz (Hz). Default is 60.

Active power P

The active power of the load, in watts. Default is 10e3.

Inductive reactive power QL

The inductive reactive power QL, in vars. Specify a positive value, or 0. Default is 100.

Capacitive reactive power QC

The capacitive reactive power QC, in vars. Specify a positive value, or 0. Default is 100.

Set the initial capacitor voltage

If selected, the initial capacitor voltage is defined by the **Capacitor initial voltage** parameter. If cleared, the software calculates the initial capacitor voltage in order to start the simulation in steady-state. Default is cleared.

The **Set the initial capacitor voltage** parameter have no effect on the block if the capacitive reactive power is equal to zero.

Capacitor initial voltage (V)

The initial capacitor voltage used at the start of the simulation. The **Capacitor initial voltage** parameter have no effect on the block if the capacitive reactive power is equal to zero and if the **Set the initial capacitor voltage** parameter is cleared. Default is 0.

Set the initial inductor current

If selected, the initial inductor current is defined by the **Inductor initial current** parameter. If cleared, the software calculates the initial inductor current in order to start the simulation steady-state. Default is cleared.

The **Set the initial inductor current** parameter have no effect on the block if the inductive reactive power is equal to zero.

Inductor initial current (A)

The initial inductor current used at the start of the simulation. The **Inductor initial current** parameter have no effect on the block if the inductive reactive power is equal to zero and if the **Set the initial inductor current** parameter is cleared. Default is 0.

Measurements

Select Branch voltage to measure the voltage across the Parallel RLC Load block terminals.

Select Branch current to measure the current flowing through the Parallel RLC Load block.

Select Branch voltage and current to measure the voltage and the current of the Parallel RLC Load block.

Default is None.

Place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurements** list box of the Multimeter block, the measurement is identified by a label followed by the block name.

Measurement	Label
Branch voltage	Ub :
Branch current	Ib :

Load Flow Tab

The Load Flow tool of the powergui block uses the parameters on this tab. These load flow parameters affect only model initialization. They do not affect simulation.

Load type

Specify the load type. The default value is constant Z.

If you select constant Z, the load impedance is determined from the nominal phase-to-phase voltage V_n , active power P, and reactive power (QL-QC) specified on the **Parameters** tab of the block dialog box. During the load flow solution, the impedance is kept constant. The effective P and Q therefore vary proportionally to the square of the bus voltage computed by the Load Flow tool.

If you select constant PQ, the active power P and reactive power Q are kept constant and equal to the values specified on the **Parameters** tab of the block dialog box. When you apply the load flow solution to the model (by clicking **Apply** in the Load Flow tool), the **Nominal phase-to-phase voltage V_n** parameter on the **Parameters** tab is automatically adjusted to the phase-to-phase bus voltage computed by the Load Flow tool.

If you select constant I, the load current is kept constant at its nominal value determined from the base voltage specified in the dialog box of the associated Load Flow Bus block and from active

power P and reactive power QL and QC specified on the **Parameters** tab of the block dialog box. The effective P and Q therefore vary proportionally to the bus voltage computed by the Load Flow tool. When you apply the load flow solution to the model (by clicking **Apply** in the Load Flow tool), the pu value of the nominal voltage Vn on the **Parameters** tab is automatically adjusted to the square root of the load voltage (in pu) computed by the Load Flow tool.

Examples

The `power_parallelload` example uses a Parallel RLC Load block to implement a load.

See Also

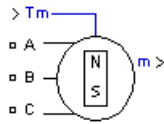
Multimeter, Parallel RLC Branch, Series RLC Branch, Series RLC Load

Version History

Introduced before R2006a

Permanent Magnet Synchronous Machine

Three-phase permanent magnet synchronous machine with sinusoidal or trapezoidal back electromotive force, or five-phase permanent magnet synchronous machine with sinusoidal back electromotive force



Libraries:

Simscape / Electrical / Specialized Power Systems / Electrical Machines

Description

The Permanent Magnet Synchronous Machine block implements a three-phase or a five-phase permanent magnet synchronous machine. The stator windings are connected in wye to an internal neutral point.

The three-phase machine can have a sinusoidal or trapezoidal back EMF waveform. The rotor can be round or salient-pole for the sinusoidal machine. The rotor is round when the machine is trapezoidal. Preset models are available for the sinusoidal back EMF machine.

The five-phase machine has a sinusoidal back EMF waveform and round rotor.

The Permanent Magnet Synchronous Machine block operates in either generator or motor mode. The mode of operation is dictated by the sign of the mechanical torque (positive for motor mode, negative for generator mode). The electrical and mechanical parts of the machine are each represented by a second-order state-space model.

The sinusoidal model assumes that the flux established by the permanent magnets in the stator is sinusoidal, which implies that the electromotive forces are sinusoidal.

The trapezoidal model assumes that the winding distribution and flux established by the permanent magnets produce three trapezoidal back EMF waveforms.

Three-Phase Sinusoidal Model Electrical System

These equations are expressed in the rotor reference frame (qd frame). All quantities in the rotor reference frame are referred to the stator.

$$\frac{d}{dt}i_d = \frac{1}{L_d}v_d - \frac{R}{L_d}i_d + \frac{L_q}{L_d}p\omega_m i_q$$

$$\frac{d}{dt}i_q = \frac{1}{L_q}v_q - \frac{R}{L_q}i_q - \frac{L_d}{L_q}p\omega_m i_d - \frac{\lambda p\omega_m}{L_q}$$

$$T_e = 1.5p[\lambda i_q + (L_d - L_q)i_d i_q]$$

L_q, L_d	q -axis and d -axis inductances
R	Resistance of the stator windings

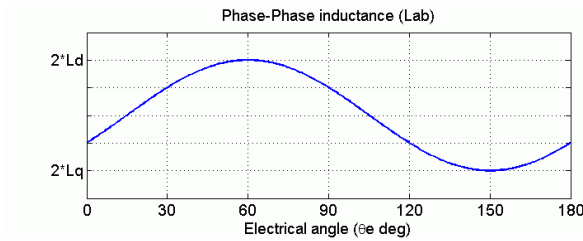
i_q, i_d	q -axis and d -axis currents
v_q, v_d	q -axis and d -axis voltages
ω_m	Angular velocity of the rotor
λ	Amplitude of the flux induced by the permanent magnets of the rotor in the stator phases
p	Number of pole pairs
T_e	Electromagnetic torque

The L_q and L_d inductances represent the relation between the phase inductance and the rotor position due to the saliency of the rotor. For example, the inductance measured between phase A and B (when phase C is left open) is given by:

$$L_{ab} = L_d + L_q + (L_q - L_d)\cos\left(2\theta_e + \frac{\pi}{3}\right),$$

where Θ_e represents the electrical angle.

The next figure shows the variation of the phase-to-phase inductance in function of the electrical angle of the rotor.



- For a round rotor, there is no variation in the phase inductance:

$$L_d = L_q = \frac{L_{ab}}{2}.$$

- For a salient round rotor, the dq inductances are given by:

$$L_d = \frac{\max(L_{ab})}{2}$$

$$L_q = \frac{\min(L_{ab})}{2}$$

Five-Phase Sinusoidal Model Electrical System

These equations are expressed in the rotor reference frame using an extended Park transformation ($q1d1$ and $q2d2$ frame). All quantities in the rotor reference frame are referred to the stator.

$$\frac{d}{dt}i_{d1} = \frac{1}{L}v_{d1} - \frac{R}{L}i_{d1} + \frac{L_q}{L}p\omega_m i_{q1}$$

$$\frac{d}{dt}i_{q1} = \frac{1}{L}v_{q1} - \frac{R}{L}i_{q1} - \frac{L_d}{L}p\omega_m i_{d1} - \frac{\lambda p\omega_m}{L}$$

$$\frac{d}{dt}i_{d2} = \frac{1}{L}v_{d2} - \frac{R}{L}i_{d2}$$

$$\frac{d}{dt}i_{q2} = \frac{1}{L}v_{q2} - \frac{R}{L}i_{q2}$$

$$T_e = 2.5p\lambda i_{q1}$$

L	Armature inductance
R	Resistance of the stator windings
i_{q1}, i_{d1}	$q1$ -axis and $d1$ -axis currents
v_{q1}, v_{d1}	$q1$ -axis and $d1$ -axis voltages
i_{q2}, i_{d2}	$q2$ -axis and $d2$ -axis currents
v_{q2}, v_{d2}	$q2$ -axis and $d2$ -axis voltages
ω_m	Angular velocity of the rotor
λ	Amplitude of the flux induced by the permanent magnets of the rotor in the stator phases
p	Number of pole pairs
T_e	Electromagnetic torque

Three-Phase Trapezoidal Model Electrical System

These equations are expressed in the phase reference frame (abc frame). Note that the phase inductance L_s is assumed to be constant and does not vary with the rotor position.

$$\frac{d}{dt}i_a = \frac{1}{3L_s}(2v_{ab} + v_{bc} - 3R_s i_a + \lambda p \omega_m (-2\Phi'_a + \Phi'_b + \Phi'_c))$$

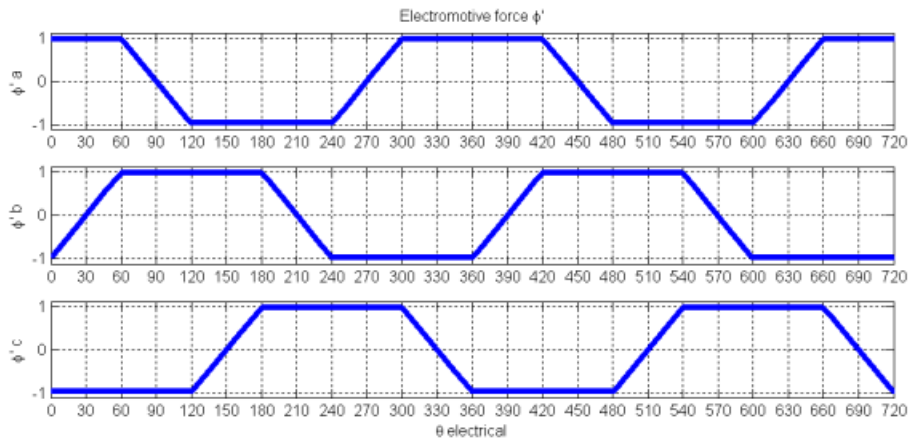
$$\frac{d}{dt}i_b = \frac{1}{3L_s}(-v_{ab} + v_{bc} - 3R_s i_b + \lambda p \omega_m (\Phi'_a - 2\Phi'_b + \Phi'_c))$$

$$\frac{d}{dt}i_c = -\left(\frac{d}{dt}i_a + \frac{d}{dt}i_b\right)$$

$$T_e = p\lambda(\Phi'_a \cdot i_a + \Phi'_b \cdot i_b + \Phi'_c \cdot i_c)$$

L_s	Inductance of the stator windings
R	Resistance of the stator windings
i_a, i_b, i_c	a, b and c phase currents
$\Phi'_a, \Phi'_b, \Phi'_c$	a, b and c phase electromotive forces, in per-unit value to the amplitude of the flux λ
v_{ab}, v_{bc}	ab and bc phase to phase voltages
ω_m	Angular velocity of the rotor
λ	Amplitude of the flux induced by the permanent magnets of the rotor in the stator phases
p	Number of pole pairs
T_e	Electromagnetic torque

The electromotive force Φ' is represented by:



Mechanical System

$$\frac{d}{dt}\omega_m = \frac{1}{J}(T_e - T_f - F\omega_m - T_m)$$

$$\frac{d\theta}{dt} = \omega_m$$

J	Combined inertia of rotor and load
F	Combined viscous friction of rotor and load
θ	Rotor angular position
T_m	Shaft mechanical torque
T_f	Shaft static friction torque
ω_m	Angular velocity of the rotor (mechanical speed)

Limitations and Assumptions

When you use Permanent Magnet Synchronous Machine blocks in discrete systems, you might have to use a small parasitic resistive load, connected at the machine terminals, to avoid numerical oscillations. Large sample times require larger loads. The minimum resistive load is proportional to the sample time. Remember that with a 25 μ s time step on a 60 Hz system, the minimum load is approximately 2.5% of the machine nominal power. For example, a 200 MVA permanent magnet synchronous machine in a power system discretized with a 50 μ s sample time requires approximately 5% of resistive load, or 10 MW. If the sample time is reduced to 20 μ s, a resistive load of 4 MW is sufficient.

The Permanent Magnet Synchronous Machine block assumes a linear magnetic circuit with no saturation of the stator and rotor iron. This assumption can be made because of the large air gap usually found in permanent magnet synchronous machines.

Ports

Input

Tm — Mechanical torque
scalar

Mechanical torque at the machine shaft. This input port is normally positive because the Permanent Magnet Synchronous Machine block is usually used as a motor. If you choose to use the block in generator mode, you can apply a negative torque input.

Dependencies

To enable this port, set **Mechanical input** to Torque T_m .

w — Machine speed
scalar

Machine speed, in rad/s.

Dependencies

To enable this port, set **Mechanical input** to Speed w .

Output

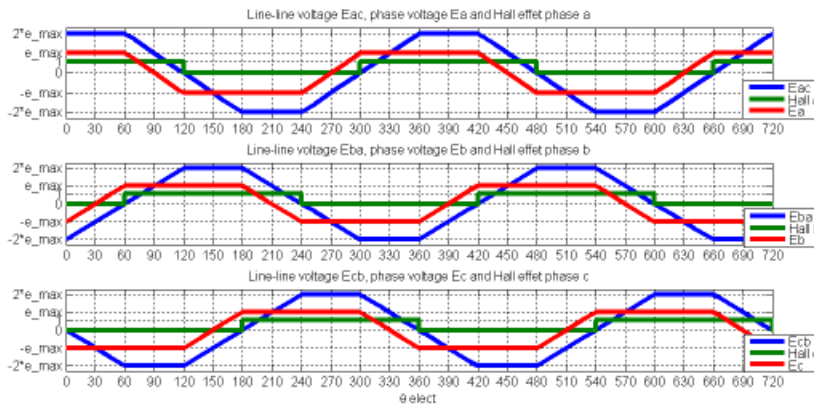
m — Machine internal measurements
13-element vector | 16-element vector

Vector containing measurement signals. The block returns a 13-element vector when **Number of phases** is set to 3, and a 16-element vector when **Number of phases** is set to 5. The available signals depend on the model you selected. You can demultiplex these signals by using the Bus Selector block provided in the Simulink library. The signals include:

Name	Definition	Units	Model
ias	Stator current is_a	A	All
ibs	Stator current is_b	A	All
ics	Stator current is_c	A	All
ids	Stator current is_d	A	Five-Phase Sinusoidal
ies	Stator current is_e	A	Five-Phase Sinusoidal
iqs	Stator current is_q	A	Three-Phase Sinusoidal
ids	Stator current is_d	A	Three-Phase Sinusoidal
iqs1	Stator current is_q1	A	Five-Phase Sinusoidal
ids1	Stator current is_d1	A	Five-Phase Sinusoidal
iqs2	Stator current is_q2	A	Five-Phase Sinusoidal
ids2	Stator current is_d2	A	Five-Phase Sinusoidal
vqs	Stator voltage Vs_q	V	Three-Phase Sinusoidal
vds	Stator voltage Vs_d	V	Three-Phase Sinusoidal
vqs1	Stator voltage Vs_q1	V	Five-Phase Sinusoidal
vds1	Stator voltage Vs_d1	V	Five-Phase Sinusoidal
vqs2	Stator voltage Vs_q2	V	Five-Phase Sinusoidal
vds2	Stator voltage Vs_d2	V	Five-Phase Sinusoidal
ea	Phase back EMF e_a	V	Three-Phase Trapezoidal

Name	Definition	Units	Model
eb	Phase back EMF e_b	V	Three-Phase Trapezoidal
ec	Phase back EMF e_c	V	Three-Phase Trapezoidal
ha	Hall effect signal h_a^*	logical (0 or 1)	Three-Phase, Sinusoidal and Trapezoidal
hb	Hall effect signal h_b^*	logical (0 or 1)	Three-Phase, Sinusoidal and Trapezoidal
hc	Hall effect signal h_c^*	logical (0 or 1)	Three-Phase, Sinusoidal and Trapezoidal
w	Rotor speed ω_m	rad/s	All
theta	Rotor angle θ_{tam}	rad	All
Te	Electromagnetic torque T_e	N.m	All

The Hall effect signal provides a logical indication of the back EMF positioning. This signal is very useful to directly control the power switches. There is a change of state at each zero crossing of the phase-to-phase voltage. These signals must be decoded before being applied to the switches.



Conserving

S — Machine rotational shaft

mechanical rotational

Mechanical rotational port, that represents the rotational shaft of the machine.

Dependencies

To enable this port, set **Mechanical input** to Mechanical rotational port.

A — Phase A electric terminal

specialized electrical

Specialized electrical conserving port associated with the electrical terminal of phase A.

B — Phase B electric terminal
specialized electrical

Specialized electrical conserving port associated with the electrical terminal of phase B.

C — Phase C electric terminal
specialized electrical

Specialized electrical conserving port associated with the electrical terminal of phase C.

Parameters

Configuration

Number of phases — Number of phases for machine model
3 (default) | 5

Select between a three-phase machine model or a five-phase machine model.

Back EMF waveform — Electromotive force
Sinusoidal (default) | Trapezoidal

Select between Sinusoidal and Trapezoidal electromotive force.

Dependencies

To enable this parameter, set **Number of phases** to 3.

Rotor type — Rotor type
Round (default) | Salient-pole

Select between Salient-pole and Round rotors.

Dependencies

To enable this parameter, set **Number of phases** to 3 and set **Back EMF waveform** to Sinusoidal.

Mechanical input — Type of input
Torque T_m (default) | Speed w | Mechanical rotational port

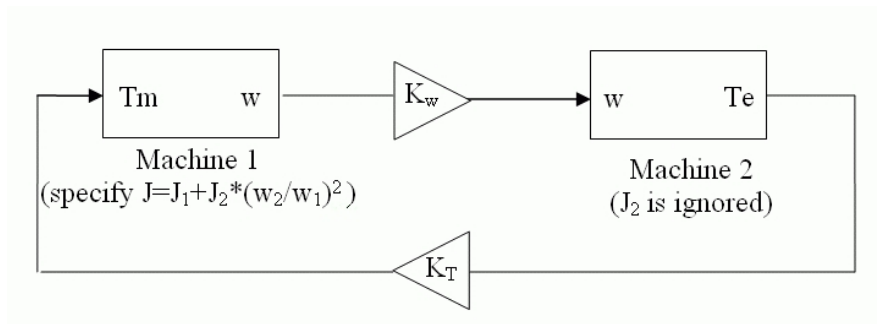
Select whether input is supplied by torque applied to the shaft, rotor speed, or a machine shaft represented by a Simscape rotational mechanical port.

Select Torque T_m to specify a torque input in N.m and expose the T_m port. The machine speed is determined by the machine Inertia J and by the difference between the applied mechanical torque T_m and the internal electromagnetic torque T_e . The sign convention for the mechanical torque is when the speed is positive. A positive torque signal indicates motor mode and a negative signal indicates generator mode.

Select Speed w to specify a speed input in rad/s and expose the w port. The machine speed is imposed and the mechanical part of the model (Inertia J) is ignored. Using the speed as the mechanical input allows modeling a mechanical coupling between two machines.

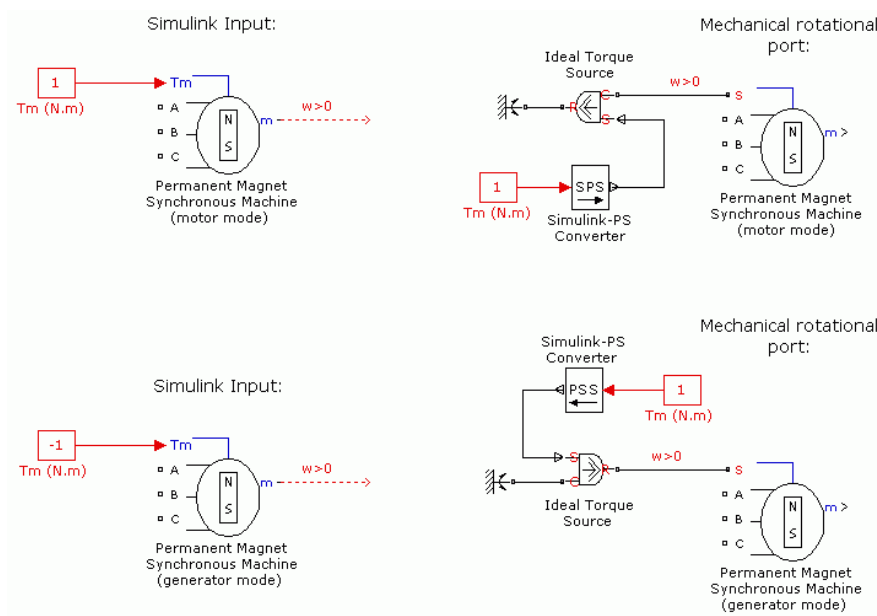
The next figure indicates how to model a stiff shaft interconnection in a motor-generator set when friction torque is ignored in machine 2. The speed output of machine 1 (the motor) is connected to

the speed input of machine 2 (the generator), while the machine 2 electromagnetic torque output, T_e , is applied to the mechanical torque input of machine 1, T_m . The K_w factor takes into account the speed units of both machines (pu or rad/s) and the gear box ratio w_2/w_1 . The K_T factor takes into account the torque units of both machines (pu or N.m) and machine ratings. Also, because the inertia J_2 is ignored in machine 2, J_2 is added to the machine 1 inertia, J_1 .



Select **Mechanical rotational port** to expose a Simscape mechanical rotational port that allows you to connect the machine shaft to other Simscape blocks with mechanical rotational ports.

The next figure indicates how to connect an Ideal Torque Source block from the Simscape library to the machine shaft to represent the machine in motor mode or in generator mode.



Preset model — Electrical and mechanical parameters

No (default) | 01: 0.8 Nm 300 Vdc 3000 RPM - 0.8 Nm | 02: 1.7 Nm 300 Vdc 3750 RPM - 1.7 Nm | ...

Predetermined electrical and mechanical parameters for various permanent magnet synchronous motor ratings of torque (N.m), DC bus voltage (V), rated speed (rpm), and continuous stall torque (N.m).

Select one of the preset models to load the corresponding electrical and mechanical parameters in the entries of the dialog box. Select **No** (default) if you do not want to use a preset model, or if you want to modify some of the parameters of a preset model.

The available preset models are:

- No
- 01: 0.8 Nm 300 Vdc 3000 RPM - 0.8 Nm
- 02: 1.7 Nm 300 Vdc 3750 RPM - 1.7 Nm
- 03: 2.8 Nm 300 Vdc 4250 RPM - 3.2 Nm
- 04: 6 Nm 300 Vdc 4500 RPM - 6 Nm
- 05: 8 Nm 300 Vdc 2000 RPM - 10 Nm
- 06: 10 Nm 300 Vdc 2300 RPM - 14.2 Nm
- 07: 20 Nm 300 Vdc 2200 RPM - 33.9 Nm
- 08: 24 Nm 300 Vdc 2300 RPM - 41.4 Nm
- 09: 7.14 Nm 560 Vdc 5000 RPM - 8.3 Nm
- 10: 7.71 Nm 560 Vdc 5000 RPM - 10.2 Nm
- 11: 26.13 Nm 560 Vdc 3000 RPM - 27.3 Nm
- 12: 35.17 Nm 560 Vdc 3000 RPM - 37.4 Nm
- 13: 42.09 Nm 560 Vdc 3000 RPM - 45.6 Nm
- 14: 67.27 Nm 560 Vdc 1700 RPM - 70.2 Nm
- 15: 87.75 Nm 560 Vdc 3000 RPM - 97.96 Nm
- 16: 111 Nm 560 Vdc 3000 RPM - 126 Nm

When you select a preset model, the electrical and mechanical parameters in the **Parameters** tab of the dialog box are disabled. To start from a given preset model and then modify machine parameters:

- 1 Select the preset model for which you want to initialize the parameters.
- 2 Change **Preset model** to No. This does not change the machine parameters in the **Parameters** tab.
- 3 Modify the machine parameters, then click **Apply**.

Dependencies

To enable this parameter, set **Number of phases** to 5, or set **Number of phases** to 3 and **Back EMF waveform** to Sinusoidal.

Use signal names to identify bus labels — Identify with signal names or signal definition
off (default) | on

When selected, the measurement output uses the signal names to identify the bus labels. Select this option for applications that require bus signal labels to have only alphanumeric characters.

When cleared, the measurement output uses the signal definition to identify the bus labels. The labels contain nonalphanumeric characters that are incompatible with some Simulink applications.

Parameters

Stator phase resistance Rs (Ohm) — Stator phase resistance

0.0485 (default) | positive scalar

Stator phase resistance R_s (Ω).

Stator phase inductance Ls (H) — Stator-phase-to-neutral inductance

8.5e-3 (default) | positive scalar

Stator-phase-to-neutral inductance L_s (H) of the trapezoidal model.

Dependencies

To enable this parameter, set **Number of phases** to 3 and **Back EMF waveform** to Trapezoidal.

Inductances [Ld(H) Lq(H)] — Phase to neutral inductances

[8.5e-3, 8.5e-3] (default) | two-element vector

Phase-to-neutral L_d (H) and L_q (H) inductances in the d -axis and q -axis of the sinusoidal model with a salient-pole rotor.

Dependencies

To enable this parameter, set **Number of phases** to 3, **Back EMF waveform** to Sinusoidal, and **Rotor type** to Salient-pole.

Armature inductance (H) — Armature inductance of sinusoidal model

0.000395 (default) | positive scalar

Armature inductance of the sinusoidal model with a round rotor. L_d is equal to L_q .

Dependencies

To enable this parameter, set **Number of phases** to 3, **Back EMF waveform** to Sinusoidal, and **Rotor type** to Round.

Specify — Machine constant

Flux linkage established by magnets (V.s) (default) | Voltage Constant ($V_{peak} L - L / \text{krpm}$) | Torque Constant ($N.m / A_{peak}$)

Machine constant for block parameterization. Once you select a constant, you can enter its value in the appropriate parameter field, while the other two parameters are disabled.

Flux linkage — Constant flux

0.1194 (default) | positive scalar

Constant flux λ (Wb) per pole pairs induced in the stator windings by the magnets.

Dependencies

To enable this parameter, set **Specify** to Flux linkage established by magnets (V.s).

Voltage constant — Peak line-to-line voltage

86.6271 (default) | positive scalar

Peak line-to-line voltage per 1000 rpm. This voltage represents the peak open circuit voltage when the machine is driven as a generator at 1000 rpm.

Dependencies

To enable this parameter, set **Specify** to Voltage Constant ($V_{\text{peak}} \text{ L-L} / \text{krpm}$).

Torque constant — Torque per ampere constant

0.7164 (default) | positive scalar

Torque per ampere constant. This constant assumes that the machine is driven by an inverter that provides a perfect synchronization between the current and the back EMF.

- Sinusoidal model: A sine wave current is assumed.
- Trapezoidal model: A square wave current is assumed.

Dependencies

To enable this parameter, set **Specify** to Torque Constant ($\text{N.m} / A_{\text{peak}}$).

Back EMF flat area (degrees) — Width of flat top

120 (default) | positive scalar

Width of the flat top for a half period of the electromotive force Φ' (degrees) for a trapezoidal machine.

Dependencies

To enable this parameter, set **Number of phases** to 3 and **Back EMF waveform** to Trapezoidal.

Inertia, viscous damping, pole pairs, static friction [$J(\text{kg.m}^2)$ $F(\text{N.m.s})$ $p()$ $T_f(\text{N.m})$] —

Mechanical parameters

[0.0027 0.0004924 4 0] (default) | three-element vector | four-element vector

Combined machine and load inertia coefficient J (kg.m^2), combined viscous friction coefficient F (N.m.s), pole pairs p , and shaft static friction T_f (N.m). If the fourth value of the vector (static friction) is not specified, the block considers this value to be 0.

Dependencies

To enable this parameter, set **Mechanical input** to Torque T_m or Mechanical rotational port.

Pole pairs p () — Number of pole pairs

4 (default) | positive scalar

Number of pole pairs, p .

Dependencies

To enable this parameter, set **Mechanical input** to Speed w .

Initial conditions [$w_m(\text{rad/s})$ $\text{thetam}(\text{deg})$ $i_a, i_b(\text{A})$] — Initial conditions for three-phase machine

[0, 0, 0, 0] (default) | four-element vector

Mechanical speed (rad/s), mechanical angle Θ_m (degrees), and instantaneous stator current (A) for a three-phase machine [w_m , Θ_m , i_a , i_b].

Because the stator is wye-connected and the neutral point is isolated, the current i_c in the three-phase machine is given by $i_c = -i_a - i_b$.

Dependencies

To enable this parameter, set **Number of phases** to 3.

Initial conditions [$w_m(\text{rad/s})$ $\theta_{m}(\text{deg})$ $i_a, i_b, i_c, i_d(\text{A})$] — Initial conditions for five-phase machine

[0 0 0 0 0] (default) | six-element vector

Mechanical speed (rad/s), mechanical angle θ_m (degrees), and instantaneous stator current (A) for a five-phase machine [$w_m, \theta_m, i_a, i_b, i_c, i_d$].

Because the stator is wye-connected and the neutral point is isolated, the current i_e in the five-phase machine is given by $i_e = -i_a - i_b - i_c - i_d$.

Dependencies

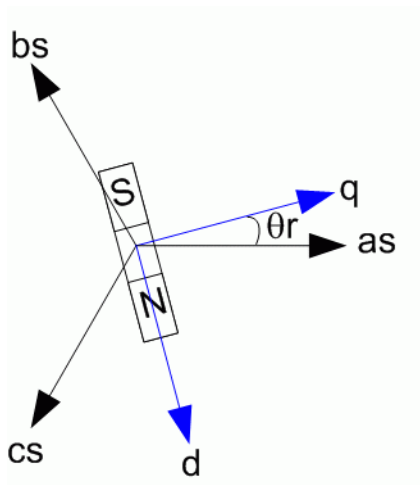
To enable this parameter, set **Number of phases** to 5.

Rotor flux position when theta = 0 — Reference position of rotor flux

90 degrees behind phase A axis (modified Park) (default) | Aligned with phase A axis (original Park)

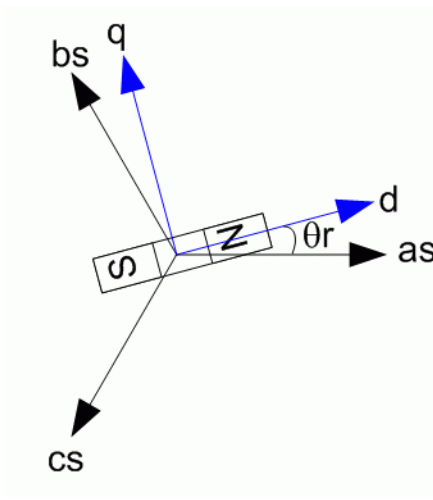
Reference position of the rotor flux relative to the phase A axis.

Select 90 degrees behind phase A axis (modified Park) to choose the reference position of the rotor represented by:



The modified Park transformation [4 on page 1-304] is more convenient for vector control because the maximum phase induction occurs at $\theta = 0$.

Select **Aligned with phase A axis (original Park)** to choose the reference position of the rotor represented by:



Advanced

To enable the **Advanced** tab, set the **Simulation type** parameter of the powergui block to **Discrete** and, on the **Preferences** tab, clear the **Automatically handle discrete solver and Advanced tab solver settings of blocks** parameter.

Discrete solver model — Integration method

Trapezoidal non iterative (default) | Trapezoidal robust | Backward Euler robust

Integration method used by the block.

When you select the **Automatically handle discrete solver and Advanced tab solver settings of blocks** parameter in the powergui block, the discrete solver model is automatically set to Trapezoidal robust.

Trapezoidal non iterative requires you to add a non-negligible shunt load at the machine terminals to maintain simulation stability, and the simulation may fail to converge and stop when the number of machines increases in the model.

Trapezoidal robust and Backward Euler robust allow you to eliminate the need to use parasitic loads. To eliminate the topological errors of the machines connected to an inductive circuit (for example, a circuit breaker connected in series with the machine), the machine models a negligible internal load of 0.01% of nominal power.

Trapezoidal robust is slightly more accurate than Backward Euler robust, especially when the model is simulated at larger sample times. Trapezoidal robust may produce slight damped numerical oscillations on machine voltage in no-load conditions, while Backward Euler robust prevents oscillations and maintains accuracy.

For more information on what method to use in your application, see “Simulating Discretized Electrical Systems”.

Sample time (-1 for inherited) — Sample time used by block

-1 (default) | scalar

Sample time used by the block. To inherit the sample time specified in the powergui block, set this parameter to -1 (default).

Version History

Introduced before R2006a

References

- [1] Grenier, D., L.-A. Dessaint, O. Akhrif, Y. Bonnassieux, and B. LePioufle. "Experimental Nonlinear Torque Control of a Permanent Magnet Synchronous Motor Using Saliency." *IEEE Transactions on Industrial Electronics*, Vol. 44, No. 5, October 1997, pp. 680-687.
- [2] Toliyat, H.A. "Analysis and Simulation of Multi-Phase Variable Speed Induction Motor Drives Under Asymmetrical Connections." *Applied Power Electronics Conference and Exposition*, Vol. 2, March 1996, pp. 586-592.
- [3] Beaudart, F., F. Labrique, E. Matagne, D. Telteux, and P. Alexandre. "Control under normal and fault tolerant operation of multiphase SMPM synchronous machines with mechanically and magnetically decoupled phases." *International Conference on Power Engineering, Energy and Electrical Drives*, March 2009, pp. 461-466.
- [4] Krause, P.C., O. Wasynczuk, and S.D. Sudhoff. *Analysis of Electric Machinery and Drive Systems*. IEEE Press, 2002.

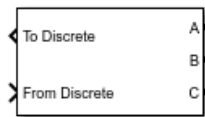
Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

Phasor-to-Discrete Interface

Connect a Discrete Phasor network to Discrete network



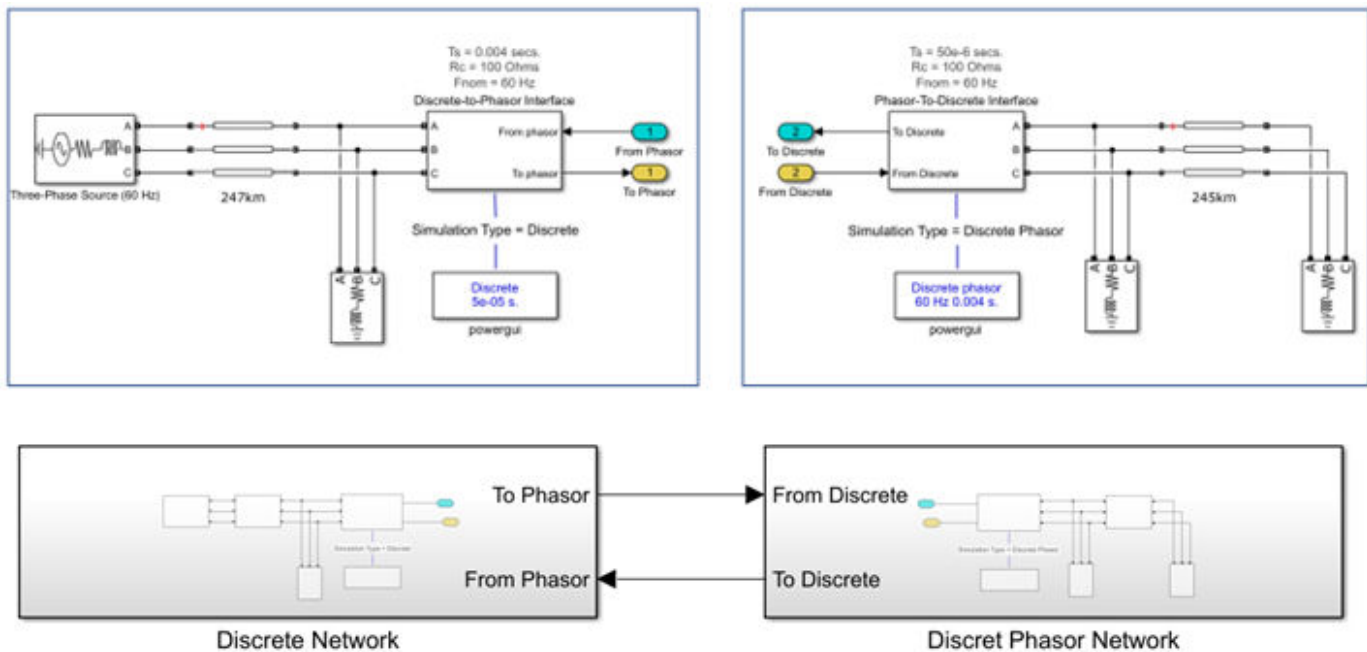
Libraries:

Simscape / Electrical / Specialized Power Systems / Utilities

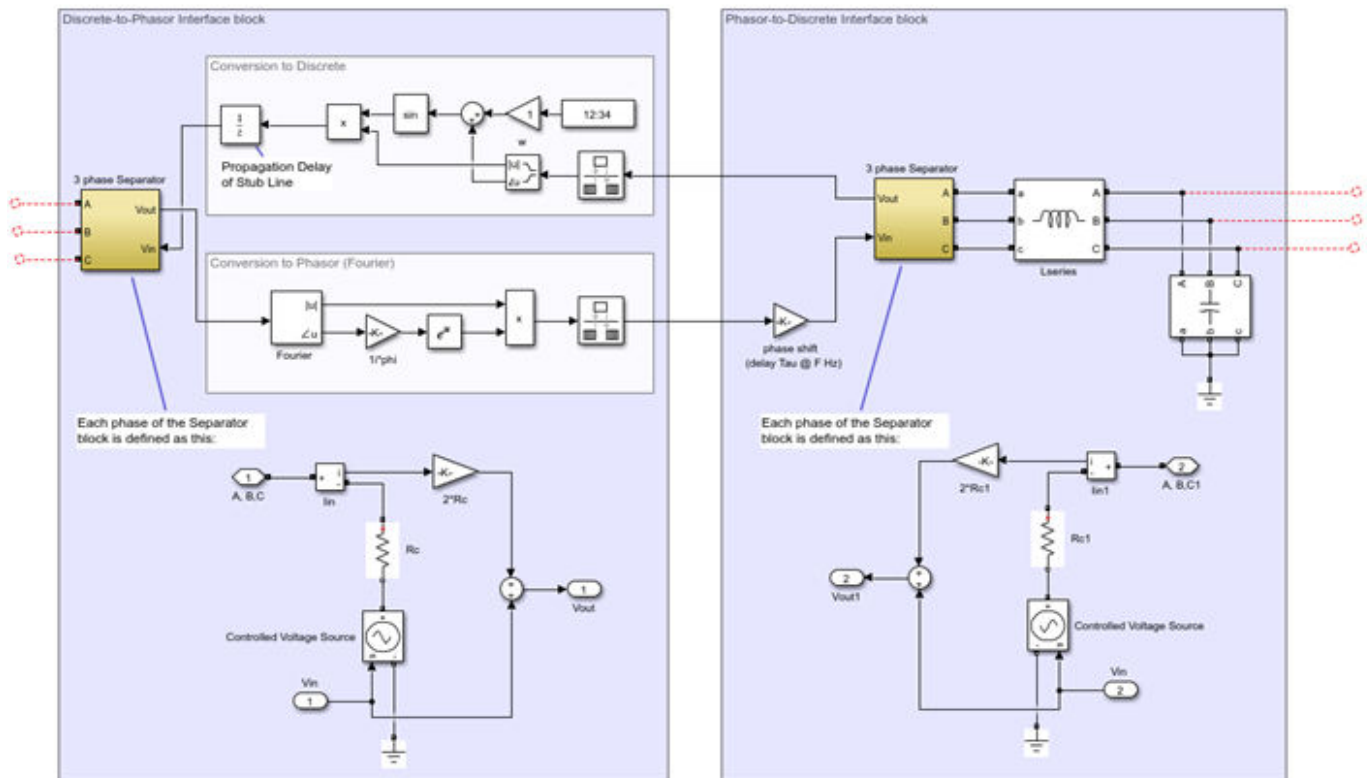
Description

Use the Phasor-to-Discrete Interface block with a Discrete-to-Phasor Interface block to connect a network simulated in Discrete Phasor mode to a network simulated in Discrete mode. The Phasor-to-Discrete Interface block must be used in a subsystem or Model Reference that has a Powergui block with Simulation type set to Discrete phasor. The electrical network that this subsystem contains is called the Discrete Phasor Network.

An associated Discrete-to-Phasor Interface block must be used in another subsystem or Model Reference that has a Powergui block with **Simulation type** set to **Discrete**. The electrical network contained that this subsystem contains is called the Discrete Network.



The Phasor-to-Discrete Interface block and the Discrete-to-Phasor Interface block form a decoupling device system. Each phase terminal of the Phasor-to-Discrete Interface block consists of a controlled Voltage Source block in series with a Current Measurement block and a resistor, connected to a series inductance and shunt capacitor, as shown in the right section of this model:



The Phasor-to-Discrete Interface block send the voltages measured at the A, B, and C terminals of the block to the Discrete-to-Phasor Interface block. It also phase shift the phasor voltages received by the Discrete-to-Phasor Interface block. Because the Phasor-to-Discrete Interface block and the Discrete-to-Phasor Interface block act together to model a single decoupling system, the values specified for the resistance R_c , nominal frequency F , and sample time T_s parameters must be the same in the two blocks.

Ports

Conserving

A — Phase A terminal
specialized electrical

Electrical conserving port associated with the phase A terminal of the block.

B — Phase B terminal
specialized electrical

Electrical conserving port associated with the phase B terminal of the block.

C — Phase C terminal
specialized electrical

Electrical conserving port associated with the phase C terminal of the block.

Simulink

From Discrete — Three-phase voltage

Vector of 3 elements

Signal representing the three-phase voltage coming from the Discrete-to-Phasor Interface block. Connect this input to an Inport block of the subsystem or Model Reference containing the block.

To Discrete — Three-phase voltage

Vector of 3 elements

Signal that measures the three-phase voltage on the Discrete network. Connect this signal to an Outport block of the subsystem or Model Reference containing the block.

Parameters

Rc (Ohms) — Decoupling resistance

100 (default) | positive scalar

Resistance of the decoupling equivalent model, in ohms. This value must be the same as the value of the **Rc (Ohms)** parameter of the associated Discrete-to-Phasor Interface block.

Nominal frequency (Hz) — Network nominal frequency

60 (default) | positive scalar

Nominal frequency of the system, in hertz. The block uses this value to phase-shift the voltage measurement coming from the Discrete-to-Phasor block. The value must be same as the value of the **Nominal frequency** parameter of the associated Discrete-to-Phasor Interface block.

Sample time of phasor network (s) — Discrete network sample time

1e-4 (default) | positive scalar

Sample time, in seconds, of the Discrete phasor network. This value must be the same as the value of the **Sample time** parameter of the Powergui block of the Discrete network.

Version History

Introduced in R2023a

Extended Capabilities

C/C++ Code Generation

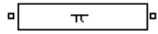
Generate C and C++ code using Simulink® Coder™.

See Also

Discrete-to-Phasor Interface

PI Section Line

Implement transmission line or cable with lumped parameters



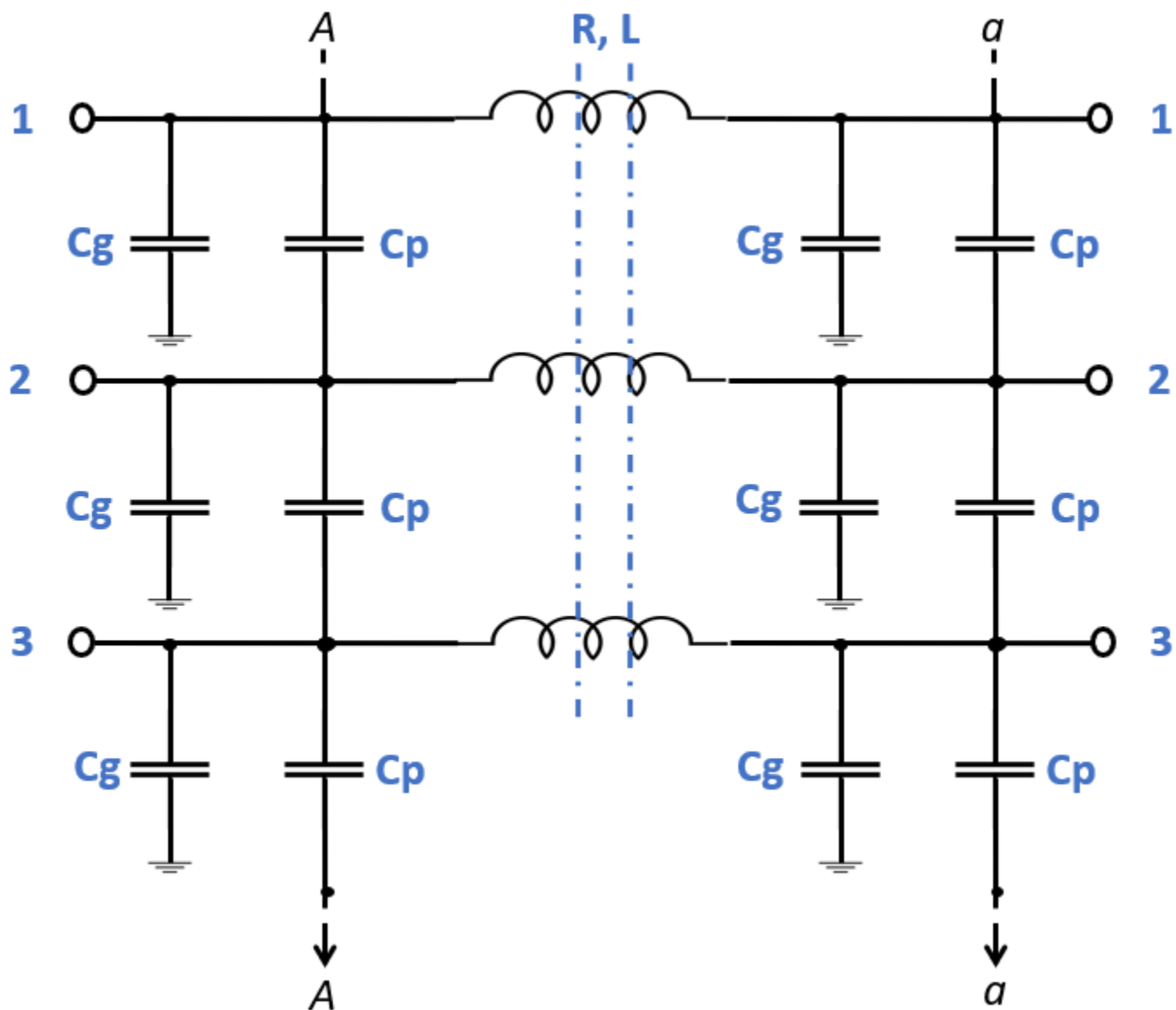
Libraries:

Simscape / Electrical / Specialized Power Systems / Power Grid Elements

Description

The PI Section Line block implements an N -phase transmission line or cable with parameters lumped in PI sections.

For a transmission line, the resistance, inductance, and capacitance are uniformly distributed along the line. An approximate model of the distributed parameter line is obtained by cascading several identical PI sections. The following figure shows one PI section of a three-phase transmission line.



When the number of phases is greater than 1, the series resistance and inductance are implemented by a mutual inductance device defined by the R and L matrices.

When the number of phases is 1, the series resistance and inductance are modeled by a single resistance R and inductance L . The phase-to-phase capacitors, Cp , are not modeled.

The shunt capacitance Cg of each phase and the phase-to-phase capacitance Cp are defined as:

$$Cg = \text{sum}(C(:,p)) / 2; \quad (1-8)$$

$$Cp = -C(p,k) \quad (1-9)$$

where C is the capacitance matrix and p and k are the number of the phases to which Cp is connected.

Unlike the Distributed Parameters Line block, which has an infinite number of states, the PI section linear model has a finite number of states that permit you to compute a linear state-space model. The number of sections to be used depends on the frequency range to be represented.

An approximation of the maximum frequency range represented by the PI line model is given by the following equation:

$$f_{\max} = \frac{N_{bpi} \cdot v}{8 \cdot ltot}$$

where

N_{bpi}	Number of PI sections
v	Propagation speed (km/s) = $\frac{1}{\sqrt{lc}}$; l in H/km, c in F/km
$ltot$	Line length (km)

For example, for a 100 km aerial line having a propagation speed of 300,000 km/s, the maximum frequency range represented with a single PI section is approximately 375 Hz. For studying interactions between a power system and a control system, this simple model could be sufficient. However for switching surge studies involving high-frequency transients in the kHz range, much shorter PI sections should be used. In fact, you can obtain the most accurate results by using a distributed parameters line model.

Note The powergui block provides the **Power Line Parameters** app and the Power Cable Parameters app, which calculates resistance, inductance, and capacitance per unit of length based on the line or cable geometry and the conductor characteristics.

Hyperbolic Correction of RLC Elements

For short line sections (approximately $lsec < 50$ km) the RLC elements for each line section are simply given by:

$$R = r \cdot lsec$$

$$L = l \cdot lsec$$

$$C = c \cdot lsec$$

where

r	Resistance per unit length (Ω/km)
l	Inductance per unit length (H/km)
c	Capacitance per unit length (F/km)
f	Frequency (Hz)
$lsec$	Line section length = $ltot / N$ (km)

However, for long line sections, the RLC elements given by the above equations must be corrected in order to get an exact line model at a specified frequency. The RLC elements are then computed using hyperbolic functions as explained below.

$$\omega = 2\pi f$$

Per unit length series impedance at frequency f is

$$z = r + j\omega l$$

Per unit length shunt admittance at frequency f is

$$y = j\omega c$$

Characteristic impedance is

$$Z_c = \sqrt{z/y}$$

Propagation constant is

$$\gamma = \sqrt{z \cdot y}$$

$$Z = R + j\omega L = Z_c \cdot \sinh(\gamma \cdot l_{\text{sec}})$$

$$R = \text{real}(Z)$$

$$L = \text{imag}(Z)/\omega$$

$$Y = \frac{2}{Z_c} \cdot \tanh\left(\gamma \cdot \frac{l_{\text{sec}}}{2}\right)$$

$$C = \text{imag}(Y)/\omega$$

Hyperbolic corrections result in RLC values slightly different from the non-corrected values. R and L are decreased while C is increased. These corrections become more important as line section length is increasing. For example, let us consider a 735 kV line with the following positive-sequence and zero-sequence parameters (these are the default parameters of the Three-Phase PI Section Line block and Distributed Parameters Line block):

Positive sequence	$r = 0.01273 \, \Omega/\text{km}$ $l = 0.9337 \times 10^{-3} \text{H}/\text{km}$ $c = 12.74 \times 10^{-9} \text{F}/\text{km}$
Zero sequence	$r = 0.3864 \, \Omega/\text{km}$ $l = 4.1264 \times 10^{-3} \text{H}/\text{km}$ $c = 7.751 \times 10^{-9} \text{F}/\text{km}$

For a 350 km line section, noncorrected RLC positive-sequence values are:

$$R = 0.01273 \times 350 = 4.455 \, \Omega$$

$$L = 0.9337 \times 10^{-3} \times 350 = 0.3268 \, \text{H}$$

$$C = 12.74 \times 10^{-9} \times 350 = 4.459 \times 10^{-6} \, \text{F}$$

Hyperbolic correction at 60 Hz yields:

$$R = 4.153 \, \Omega$$

$$L = 0.3156 \, \text{H}$$

$$C = 4.538 \times 10^{-6} \, \text{F}$$

For these particular parameters and long line section (350 km), corrections for positive-sequence RLC elements are relatively important (respectively -6.8% , -3.4% , and $+1.8\%$). For zero-sequence

parameters, you can verify that even higher RLC corrections must be applied (respectively -18% , -8.5% , and $+4.9\%$).

The PI Section Line block always uses the hyperbolic correction, regardless of the line section length.

Examples

The `power_piline` example shows the line energization voltages and currents of a single-phase PI section line.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Number of phases [N] — Number of phases

1 (default) | scalar greater than or equal to 1

Number of phases of the PI-section line. The number of ports of this block is equal to the value of this parameter.

Number of pi sections — Number of PI sections

1 (default) | scalar greater than or equal to 1

Number of PI sections.

Line length (km) — Line section length

100 (default) | positive scalar

Line length in kilometers (km).

Frequency used for rlc specification (Hz) — Frequency used for specification of per unit length rlc line parameters

60 (default) | positive scalar

Frequency f , in hertz (Hz), at which per unit length r , l , c parameters are specified. Hyperbolic correction is applied on RLC elements of each line section using this frequency.

Resistance per unit length (Ohms/km) [NxN matrix] — Resistance per unit length

0.01273 (default) | matrix of N-by-N elements

Resistance per unit length, as an N -by- N matrix in ohms/km (Ω/km).

Inductance per unit length (H/km) [NxN matrix] — Inductance per unit length

0.9337e-3 (default) | matrix of N-by-N elements

Inductance per unit length of the line, as an N -by- N matrix in henries/km (H/km). The terms in the matrix diagonal cannot be zero, because it would result in an invalid propagation speed computation.

Capacitance per unit length (F/km) [NxN matrix] — Capacitance per unit length

12.74e-9 (default) | matrix of N-by-N elements

Capacitance per unit length of the line, as an N -by- N matrix in farads/km (F/km). The terms in the matrix diagonal cannot be zero, because it would result in an invalid propagation speed computation.

Specify conductance — Option to specify conductance
off (default) | on

Option to specify the shunt conductance per unit length.

Conductance per unit length (mho/km) [NxN matrix] — Shunt conductance per unit length
0 (default) | matrix of N-by-N elements

Shunt conductance per unit length as an N-by-N matrix in mho/km.

Shunt conductance is normally not considered for transmission line since leakage currents flowing across insulators and air is significantly smaller than nominal current. However, if you are using the Pi Section Line block to model a cable arrangement, the shunt conductance can be non negligible.

Dependencies

To enable this parameter, select the **Specify conductance** parameter.

Measurements — Block measurements

None (default) | Input and output voltages | Input and output currents | All pi-section voltages and currents | All voltages and currents

When the **Number of phases [N]** parameter is greater than 1, the only measurement available in the drop-down menu is Phase-to-ground voltages.

When the **Number of phases [N]** is 1, you can select these options:

Select **Input and output voltages** to measure the sending end (input port) and receiving end (output port) voltages of the line model.

Select **Input and output currents** to measure the sending end and receiving end currents of the line model.

Select **All pi-section voltages and currents** to measure voltages and currents at the start and end of each pi-section.

Select **All voltages and currents** to measure the sending end and receiving end voltages and currents of the line model.

Default is None.

Place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurements** list box of the Multimeter block, the measurement is identified by a label followed by the block name.

Measurement	Label
Sending end voltage (block input)	Us:
Receiving end voltage (block output)	Ur:
Sending end current (input current)	Is:
Receiving end current (output current)	Ir:
Phase-to-ground (block inputs and outputs)	Us phase 1, 2, 3, ...N
	Ur phase 1, 2, 3, ...N

Version History

Introduced before R2006a

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

Distributed Parameters Line | Three-Phase PI Section Line | Power Line Parameters | Power Cable Parameters

PLL

Determine frequency and fundamental component of signal phase angle



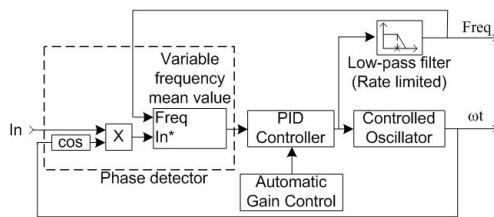
Libraries:

Simscape / Electrical / Specialized Power Systems / Control

Description

The PLL block models a Phase Lock Loop (PLL) closed-loop control system, which tracks the frequency and phase of a sinusoidal signal by using an internal frequency oscillator. The control system adjusts the internal oscillator frequency to keep the phases difference to 0.

The figure shows the internal diagram of the PLL.



The input signal is mixed with an internal oscillator signal. The DC component of the mixed signal (proportional to the phase difference between these two signals) is extracted with a variable frequency mean value. A Proportional-Integral-Derivative (PID) controller with an optional automatic gain control (AGC) keeps the phase difference to 0 by acting on a controlled oscillator. The PID output, corresponding to the angular velocity, is filtered and converted to the frequency, in hertz, which is used by the mean value.

Characteristics

Sample Time	Specified in the Sample Time parameter. Continuous when Sample Time = 0.
Scalar Expansion	No
Dimensionalized	No
Zero-Crossing Detection	Yes

Examples

The power_PLL example shows the use of the PLL (3ph) and PLL blocks.

The PLL block is fed by a sinusoidal signal of 60 Hz, increasing to 61 Hz from 0.5 s to 1.5 s. Notice that the frequency reaches the new frequency in a short response time.

The PLL (3ph) block is fed by three-phase sinusoidal signals increasing from 60 Hz to 61 Hz between 0.5 and 1.5 seconds. The PLL (3ph) frequency reaches the new frequency faster than the PLL due to the additional phase information.

The model sample time is parameterized with the variable T_s (with a default value of 0). To discretize the PLL block, at the MATLAB command prompt, enter

```
 $T_s = 50e-6$ 
```

Ports

Input

In — Normalized input signal
scalar

Normalized input signal, in pu.

Output

freq — Measured frequency
scalar

Measured frequency, in hertz.

wt — Angle synchronized on zero crossing of fundamental of input signal
scalar in the range $[0, 2\pi]$

Angle (rad) varying between 0 and 2π , synchronized on zero crossings (rising) of the fundamental of the input signal.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Minimum frequency (Hz) — Minimum expected frequency of input signal
45 (default) | positive scalar

Minimum expected frequency of the input signal. This parameter sets the buffer size of the Mean (Variable Frequency) block used inside the block to compute the mean value.

Initial inputs [Phase (degrees), Frequency (Hz)] — Initial phase and frequency of input signal
 $[0, 60]$ (default) | vector of two elements

Initial phase and frequency of the input signal.

Regulator gains [Kp, Ki, Kd] — Proportional, integral, and derivative gains of internal PID controller
 $[180, 3200, 1]$ (default) | vector of three elements

Proportional, integral, and derivative gains of the internal PID controller. Use the gains to tune the PLL response time, overshoot, and steady-state error performances.

Time constant for derivative action (s) — Time constant for first-order filter
 $1e-4$ (default) | nonnegative scalar

Time constant for the first-order filter of the PID derivative block.

Maximum rate of change of frequency (Hz/s) — Maximum positive and negative slope of signal frequency

12 (default) | positive scalar

Maximum positive and negative slope of the signal frequency.

Filter cut-off frequency for frequency measurement (Hz) — Second-order lowpass filter cut-off frequency

25 (default) | scalar

Second-order lowpass filter cut-off frequency.

Sample time — Block sample time

0 (default) | nonnegative scalar

Sample time of the block, in seconds. Set to 0 to implement a continuous block.

Enable automatic gain control — Automatic gain control option

on (default) | off

When you select this parameter, the PLL block optimizes its performances by scaling the PID regulator signal according to the input signal magnitude. Select this option when the input signal is not normalized.

Version History

Introduced in R2013a

Extended Capabilities

C/C++ Code Generation

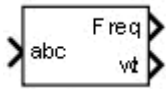
Generate C and C++ code using Simulink® Coder™.

See Also

PLL (3ph) | PMU (PLL-Based, Positive-Sequence)

PLL (3ph)

Determine frequency and fundamental component of three-phase signal phase angle



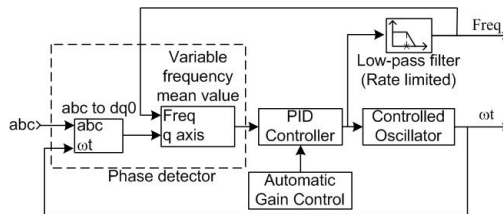
Libraries:

Simscape / Electrical / Specialized Power Systems / Control

Description

The PLL (3ph) block models a Phase Lock Loop (PLL) closed-loop control system, which tracks the frequency and phase of a sinusoidal three-phase signal by using an internal frequency oscillator. The control system adjusts the internal oscillator frequency to keep the phases difference to 0.

The figure shows the internal diagram of the PLL.



The three-phase input signal is converted to a dq0 rotating frame (Park transform) using the angular speed of an internal oscillator. The quadrature axis of the signal, proportional to the phase difference between the abc signal and the internal oscillator rotating frame, is filtered with a Mean (Variable Frequency) block. A Proportional-Integral-Derivative (PID) controller, with an optional automatic gain control (AGC), keeps the phase difference to 0 by acting on a controlled oscillator. The PID output, corresponding to the angular velocity, is filtered and converted to the frequency, in hertz, which is used by the mean value.

Characteristics

Sample Time	Specified in the Sample Time parameter. Continuous when Sample Time = 0.
Scalar Expansion	No
Dimensionalized	No
Zero-Crossing Detection	Yes

Examples

The power_PLL example shows the use of the PLL (3ph) and PLL blocks.

The PLL block is fed by a sinusoidal signal of 60 Hz, increasing to 61 Hz from 0.5 s to 1.5 s. Notice that the frequency reaches the new frequency in a short response time.

The PLL (3ph) block is fed by three-phase sinusoidal signals increasing from 60 Hz to 61 Hz between 0.5 and 1.5 seconds. The PLL (3ph) frequency reaches the new frequency faster than the PLL due to the additional phase information.

The model sample time is parameterized with the variable T_s (with a default value of 0). To discretize the PLL block, at the MATLAB command prompt, enter

```
 $T_s = 50e-6$ 
```

Ports

Input

abc — Normalized three-phase signals
vector

Normalized three-phase signals, in pu.

Output

freq — Measured frequency
scalar

Measured frequency, in hertz.

wt — Angle synchronized on zero crossing of fundamental of phase A
scalar in the range $[0, 2\pi]$

Angle (rad) varying between 0 and 2π , synchronized on zero crossings of the fundamental (positive-sequence) of phase A.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Minimum frequency (Hz) — Minimum expected frequency of input signal
45 (default) | positive scalar

Minimum expected frequency of the input signal. This parameter sets the buffer size of the Mean (Variable Frequency) block used inside the block to compute the mean value.

Initial inputs [Phase (degrees), Frequency (Hz)] — Initial phase and frequency of input signal
[0, 60] (default) | vector of two elements

Initial phase and frequency of the input signal.

Regulator gains [Kp, Ki, Kd] — Proportional, integral, and derivative gains of internal PID controller
[180, 3200, 1] (default) | vector of three elements

Proportional, integral, and derivative gains of the internal PID controller. Use the gains to tune the PLL response time, overshoot, and steady-state error performances.

Time constant for derivative action (s) — Time constant for first-order filter
 $1e-4$ (default) | nonnegative scalar

Time constant for the first-order filter of the PID derivative block.

Maximum rate of change of frequency (Hz/s) — Maximum positive and negative slope of signal frequency

12 (default) | positive scalar

Maximum positive and negative slope of the signal frequency.

Filter cut-off frequency for frequency measurement (Hz) — Second-order lowpass filter cut-off frequency

25 (default) | scalar

Second-order lowpass filter cut-off frequency.

Sample time — Block sample time

0 (default) | nonnegative scalar

Sample time of the block, in seconds. Set to 0 to implement a continuous block.

Enable automatic gain control — Automatic gain control option

on (default) | off

When you select this parameter, the PLL block optimizes its performances by scaling the PID regulator signal according to the input signal magnitude. Select this option when the input signal is not normalized.

Version History

Introduced in R2013a

Extended Capabilities

C/C++ Code Generation

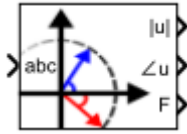
Generate C and C++ code using Simulink® Coder™.

See Also

PLL | PMU (PLL-Based, Positive-Sequence)

PMU (PLL-Based, Positive-Sequence)

Implements a phasor measurement unit using a phase-locked loop

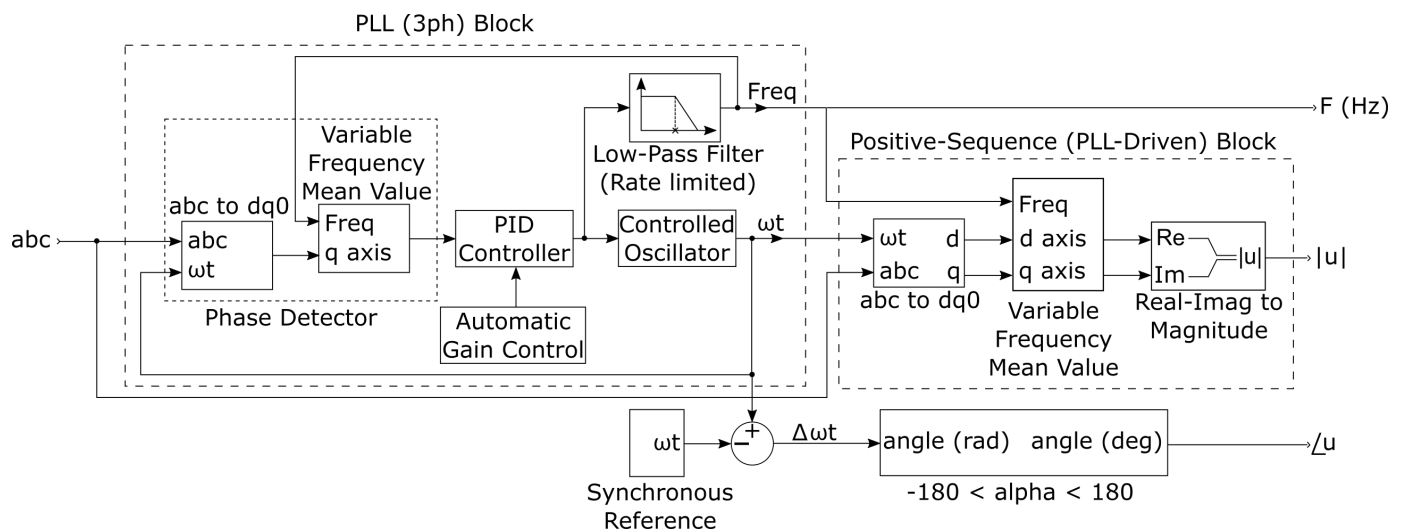


Libraries:

Simscape / Electrical / Specialized Power Systems / Sensors and Measurements

Description

The PMU (PLL-Based, Positive-Sequence) block implements a phasor measurement unit (PMU) using a phase-locked loop (PLL), which computes the positive-sequence component of the input abc signal over a running window of one cycle of fundamental frequency given by input abc . The signal can be a set of three balanced or unbalanced signals which may contain harmonics. The PMU (PLL-Based, Positive-Sequence) block is inspired by the IEEE Std C37.118.1-2011.



The PLL (3ph) block tracks the frequency and phase of a sinusoidal three-phase signal (abc) by using an internal frequency oscillator. The control system adjusts the internal oscillator frequency to keep the phase difference at 0 .

The Positive-Sequence (PLL-Driven) block computes the positive-sequence components (magnitude and phase) of a sinusoidal three-phase input signal (abc) over a running window of one cycle of the fundamental frequency tracked by the PLL (3ph) closed-loop control system. The reference frame required for the computation is given by the angle (rad, varying between 0 and 2π), synchronized on zero crossings of the fundamental (positive-sequence) of phase A. The angle is also tracked by the PLL (3ph) closed-loop control system.

Because the block uses a running average window to perform the Fourier analysis, one cycle of simulation must complete before the outputs give the correct magnitude and angle. For example, the block response to a step change in the positive-sequence component of a three-phase signal is a one-cycle ramp. For the first cycle of simulation, the output is held constant at the values specified by the initial input parameters.

The three outputs of the PMU (PLL-Based, Positive-Sequence) block return the magnitude (same units as the input signal), the phase (in degrees relative to the PLL phase), and the frequency of the positive-sequence component of the *abc* input at the fundamental frequency, respectively.

The sample time (T_s) of the block, in seconds, is a function of the nominal frequency f_n and the sampling rate N_{sr} .

Finally, the reporting rate (R_t), that determines the length of the interval over which an event will be reported, is related to the sample time using a reporting rate factor k , as follows:

$$R_t = k \times T_s$$

Limitations

- Under subsynchronous conditions, the phasor estimation may present erroneous results.
- Time synchronization from the common time source of a global positioning systems (GPS) radio clock is implicit in the model.

Ports

Input

abc — Signal
vector

Three-phase sinusoidal voltage or current.

Data Types: `single` | `double`

Output

|u| — Magnitude
scalar

Magnitude of the positive-sequence component of the *abc* input at the fundamental frequency in the same units as the input signal.

Data Types: `single` | `double`

∠u — Phase
scalar

The phase, in degrees, relative to the PLL phase of the positive-sequence component of the *abc* input at the fundamental frequency.

Data Types: `single` | `double`

F — Frequency, Hz
scalar

Frequency, in Hz, relative to the PLL phase of the positive-sequence component of the *abc* input at the fundamental frequency.

Data Types: `single` | `double` | `int8` | `int16` | `int32` | `uint8` | `uint16` | `uint32` | `Boolean` | `fixed point` | `enumerated` | `bus`

Parameters

Nominal frequency f_n (Hz) — Frequency

60 (default) | 50

Nominal frequency, f_n , in Hz, of the three-phase input.

Sampling rate N_{sr} (point/cycle) — Sampling rate

64 (default) | 4824

Sampling rate, N_{sr} , in point/cycle.

Reporting rate factor $[k]$ — Reporting rate

1 (default)

Sampling rate of the PMU. This parameter multiplies the sample time to calculate the reporting rate. The reporting rate factor must be equal to or greater than 1.

Version History

Introduced in R2017b

References

- [1] *IEEE Standard for Synchrophasor Measurements for Power Systems*. IEEE Std C37.118.1-2011 (Revision of IEEE Std C37.118-2005), pp. 1-61, 2011.
- [2] P. Kundur, N. J. Balu, and M. G. Lauby, *Power system stability and control*. Vol. 7. New York: McGraw-Hill, 1994.

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

PLL (3ph) | Positive-Sequence (PLL-Driven)

Positive-Sequence (PLL-Driven)

Compute positive-sequence component of three-phase signal at fundamental frequency



Library

Simscape / Electrical / Specialized Power Systems / Sensors and Measurements

Description

The Positive-Sequence (PLL-Driven) block computes the positive-sequence component (magnitude and phase) of the input 3 signal over a running window of one cycle of the fundamental frequency given by input 1. The reference frame required for the computation is given by the input 2. The first two inputs are normally connected to a PLL block.

The positive-sequence component of a three-phase signal (V_1 or current I_1) is computed as

$$V_1 = \frac{1}{3}(V_a + a \cdot V_b + a^2 \cdot V_c)$$

V_a, V_b, V_c = three voltage phasors at the fundamental frequency

$$a = e^{j2\pi/3} = 1 \angle 120^\circ \text{ complex operator}$$

A Fourier analysis over a sliding window of one cycle of the fundamental frequency (input 1) is first applied to the three input signals. The block evaluates the phasor values at the fundamental. Then the transformation is applied to obtain the positive sequence.

As the block uses a running average window to perform the Fourier analysis, one cycle of simulation must complete before the outputs give the correct magnitude and angle. For example, the block response to a step change of V_1 is a one-cycle ramp. For the first cycle of simulation, the output is held constant to the values specified by the initial input parameters.

Parameters

Initial frequency (Hz)

Specify the frequency of the first cycle of simulation. Default is 60.

Minimum frequency (Hz)

The minimum frequency value determines the buffer size of the Variable Time Delay block used inside the block to compute the phasors. Default is 45.

Initial input (positive component) [Mag, Phase-relative-to-PLL (degrees)]

Specify the initial positive-sequence magnitude and phase (relative to the PLL phase), in degrees, of the input signals. Default is [1, 0].

Sample time

Specify the sample time of the block, in seconds. Set to 0 to implement a continuous block. Default is 0.

Inputs and Outputs**Freq**

Fundamental frequency (Hz) required by the computation. This input is normally connected to the Freq output of a PLL block.

wt

Angle of the reference frame (rad/s) required for the computation. This input is normally connected to the wt output of a PLL block.

abc

The vectorized signal of the three [a b c] sinusoidal signals. Typical input signals are voltages or currents measured by Current Measurement or Voltage Measurement blocks.

|u|Magnitude

Outputs the magnitude of the positive-sequence component in the same unit as the input signal.

∠uPhase

Outputs the phase of the positive-sequence component, in degrees, relative to the reference frame wt (input 2).

Characteristics

Sample Time	Specified in the Sample Time parameter Continuous if Sample Time = 0
Scalar Expansion	No
Dimensionalized	No

Examples

The `power_PositiveSequencePLLDri`ven model computes the positive-sequence component of a three-phase signal containing harmonics. It shows that the block outputs accurate magnitude and phase even if the fundamental frequency of the input signal varies during the simulation.

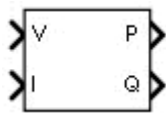
The model sample time is parameterized by the `Ts` variable set to a default value of 50e-6 s. Set `Ts` to 0 in the command window to simulate the model in continuous mode.

Version History

Introduced in R2013a

Power

Compute active and reactive powers of voltage-current pair at fundamental frequency



Libraries:
Simscape / Electrical / Specialized Power Systems / Sensors and Measurements

Description

The Power block computes the active power (P), in watts, and the reactive power (Q), in vars, of a voltage-current pair at fundamental frequency. To perform this computation, the block first determines the fundamental values (magnitude and phase) of the two input signals V & I. The following P and Q are then calculated:

$$P = \frac{|V|}{\sqrt{2}} \times \frac{|I|}{\sqrt{2}} \times \cos(\varphi)$$
$$Q = \frac{|V|}{\sqrt{2}} \times \frac{|I|}{\sqrt{2}} \times \sin(\varphi)$$
$$\varphi = \angle V - \angle I$$

As this block uses a running average window, one cycle of simulation must complete before the outputs give the correct value. For the first cycle of simulation, the outputs are held constant using the values specified by the **Voltage initial input** and **Current initial input** parameters.

Examples

The power_Power model shows the accuracy of the Power block in evaluating the active and reactive powers of a voltage-current set.

The model sample time is parameterized by the Ts variable set to a default value of 50e-6 s. Set Ts to 0 in the command window to simulate the model in continuous mode.

Characteristics

Sample Time	Specified in the Sample Time parameter Continuous if Sample Time = 0
Scalar Expansion	Yes, of the parameters
Dimensionalized	Yes

Ports

Input

V — Voltage signal
scalar

The voltage signal.

I — Current signal
scalar

The current signal.

Output

P — Active power
scalar

The active power of voltage-current pair signal, in watts.

Q — Reactive power
scalar

The reactive power of voltage-current pair signal, in vars.

Parameters

Fundamental frequency (Hz) — Input signal fundamental frequency
60 (default) | positive scalar

Specify the fundamental frequency, in hertz, of the input signals.

Voltage initial input [Mag, Phase (degrees)] — Initial voltage magnitude and phase
[1, 0] (default) | two-element vector

Specify the voltage initial magnitude and phase to set a constant output for the first cycle of simulation.

Current initial input [Mag, Phase (degrees)] — Initial current magnitude and phase
[1, 0] (default) | two-element vector

Specify the current initial magnitude and phase to set a constant output for the first cycle of simulation.

Sample time — Block sample time
0 (default) | nonnegative scalar

Specify the sample time of the block, in seconds. Set to 0 to implement a continuous block.

Version History

Introduced in R2013a

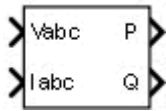
Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

Power (3ph, Instantaneous)

Compute three-phase instantaneous active and reactive powers



Libraries:

Simscape / Electrical / Specialized Power Systems / Sensors and Measurements

Description

The Power (3ph, Instantaneous) block computes the three-phase instantaneous active power P (in watts) and reactive power Q (in vars) associated with a periodic set of three-phase voltages and currents. These formulas are used to perform the calculations:

$$P = V_a \cdot I_a + V_b \cdot I_b + V_c \cdot I_c$$

$$Q = \frac{1}{\sqrt{3}}[(V_b - V_c) \cdot I_a + (V_c - V_a) \cdot I_b + (V_a - V_b) \cdot I_c]$$

With these formulas, a current flowing into an RL circuit produces a positive P and a positive Q .

The computed instantaneous reactive power is accurate only for balanced and harmonic-free three-phase voltages and currents.

Examples

The `power_ThreePhasePower` model compares the outputs of the block with the Power (Positive-Sequence) block and the Power (dq0, Instantaneous) block. It shows that the Power (3ph, Instantaneous) block gives accurate results when the voltages and currents are balanced and harmonic free. When the voltage supply becomes unbalanced, the P output exhibits a ripple, and an error appears on the Q output.

The model sample time is parameterized by the T_s variable set to a default value of 50e-6 s. Set T_s to 0 in the command window and change the **Simulation type** parameter of the Powergui block to Continuous to simulate the model in continuous mode.

Ports

Input

Vabc — Three-phase voltage signal

Three-phase voltage signal.

Iabc — Three-phase current signal

Three-phase current signal.

Output

P — Three-phase instantaneous active power

Three-phase instantaneous active power P , in watts.

Q — Three-phase instantaneous reactive power

Three-phase instantaneous reactive power Q , in vars.

Version History

Introduced in R2013a

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

Power (3ph, Phasor) | Power (Phasor) | Power (PLL-Driven, Positive-Sequence) | Power (Positive-Sequence)

Power (3ph, Phasor)

Compute three-phase active and reactive powers using three-phase voltage and current phasors



Library

Simscape / Electrical / Specialized Power Systems / Sensors and Measurements

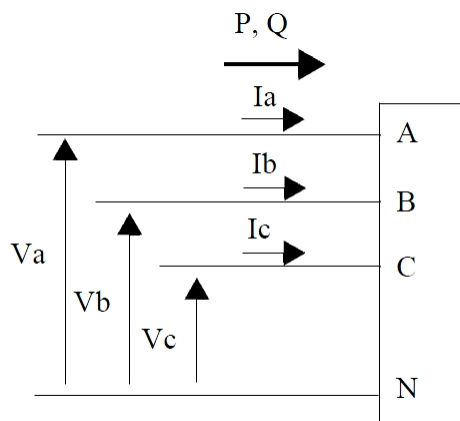
Description

The Power (3ph, Phasor) block outputs total active power P and reactive power Q of a three-phase voltage-current phasor signal pair. P and Q are computed as follows:

$$P + jQ = \frac{1}{2}(V_a \times I_a^* + V_b \times I_b^* + V_c \times I_c^*)$$

I_a^* , I_b^* , I_c^* are the complex conjugates of I_a , I_b , and I_c .

The figure shows sign conventions:



Inputs and Outputs

Vabc

The three-phase phasor (three complex signals) voltage signal, in volts peak.

Iabc

The three-phase phasor (three complex signals) current signal, in amperes peak.

P

Returns the total active power, in watts.

Q

Returns the total active power, in vars.

Characteristics

Direct Feedthrough	Yes
Dimensionalized	No

Examples

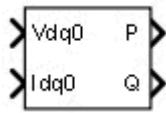
The `power_PhasorPowerMeasurements` example shows the use of the Power (3ph, Phasor) block.

Version History

Introduced in R2013a

Power (dq0, Instantaneous)

Compute three-phase instantaneous active and reactive powers



Libraries:

Simscape / Electrical / Specialized Power Systems / Sensors and Measurements

Description

The Power (dq0, Instantaneous) block computes the three-phase instantaneous active and reactive powers from a periodic set of three-phase voltages and currents expressed in the dq0 reference frame. These formulas perform the calculations:

$$P = \frac{3}{2} \times (V_d \cdot I_d + V_q \cdot I_q + 2 \cdot V_0 \cdot I_0)$$

$$Q = \frac{3}{2} \times (V_q \cdot I_d - V_d \cdot I_q)$$

The inputs Vdq0 and Idq0 (direct axis, quadrature axis, and zero-sequence components) of the three-phase voltage and current are normally connected to the outputs of the abc-to-dq0 Transformation block.

With these formulas, a current flowing into an RL circuit produces a positive P and a positive Q.

The computed instantaneous reactive power is accurate only for balanced and harmonic-free three-phase voltages and currents.

Examples

The power_ThreePhasePower model compares the outputs of the block with the Power (Positive-Sequence) block and the Power (3ph, Instantaneous) block. It shows that the Power (dq0, Instantaneous) block gives accurate results when the voltages and currents are balanced and harmonic free. When the voltage supply becomes unbalanced, the P output exhibits a ripple, and an error appears on the Q output.

The model sample time is parameterized by the Ts variable set to a default value of 50e-6 s. Set Ts to 0 in the command window and change the **Simulation type** parameter of the Powergui block to Continuous to simulate the model in continuous mode.

Ports

Input

Vdq0 — Voltage signal expressed in dq0 reference frame

Voltage signal expressed in the dq0 reference frame.

Idq0 — Current signal expressed in dq0 reference frame

Current signal expressed in the dq0 reference frame.

Output

P — Three-phase instantaneous active power

Three-phase instantaneous active power P, in watts.

Q — Three-phase instantaneous reactive power

Three-phase instantaneous reactive power Q, in vars.

Version History

Introduced in R2013a

Extended Capabilities**C/C++ Code Generation**

Generate C and C++ code using Simulink® Coder™.

See Also

Power (3ph, Instantaneous) | Power (3ph, Phasor)

powergui

Environment block for Simscape Electrical Specialized Power Systems models



Libraries:

Simscape / Electrical / Specialized Power Systems

Description

The powergui block allows you to choose one of these methods to solve your circuit:

- Continuous, which uses a variable-step solver from Simulink
- Discretization of the electrical system for a solution at fixed time steps
- Continuous or discrete phasor solution

The powergui block also opens tools for steady-state and simulation results analysis and for advanced parameter design.

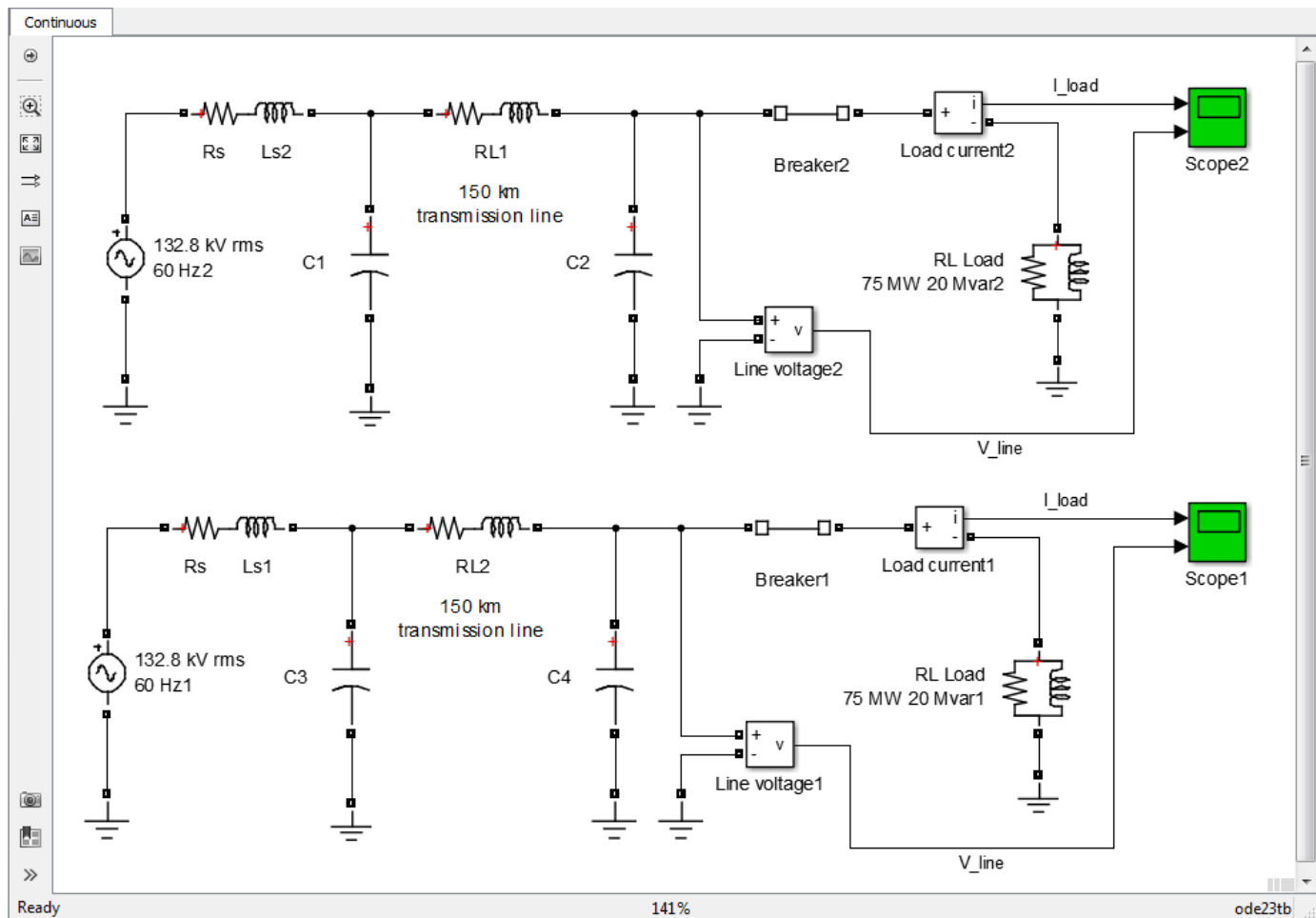
You need the powergui block to simulate any Simulink model containing Simscape Electrical Specialized Power Systems blocks. It stores the equivalent Simulink circuit that represents the state-space equations of the model.

When using one powergui block in a model:

- Place the powergui block in the top-level diagram for optimal performance.
- Make sure that the block is named **powergui**.

Note The powergui block becomes disabled during model update. To ensure proper model execution, do not restore the library link for the powergui block.

You can use multiple powergui blocks in a system that contains two or more independent electrical circuits that you want to simulate with different powergui solvers. For example, this system simulates the upper electrical circuit in discrete mode and the bottom circuit in continuous mode. The system compares the simulation performance of the two methods.



To do so, put each circuit in a different subsystem, and then add a powergui block inside each subsystem.

When you use more than one powergui block in a model:

- Do not place a powergui block in the top-level diagram.
- Place every independent model in a different subsystem.
- Place a single powergui block in the top level diagram of every subsystem.

Note Do not include commented-out powergui blocks in your model. Doing so will prevent the model from simulating.

Parameters

Solver

Simulation type — Simulation type

Continuous (default) | Discrete | Phasor | Discrete phasor

Simulation type for the model:

- **Continuous** — Perform a continuous solution of the model.
- **Discrete** — Perform a discretization of the model using the sample time specified by the **Sample time (s)** parameter.
- **Phasor** — Perform continuous phasor simulation of the model at the frequency specified by the **Frequency (Hz)** parameter.
- **Discrete phasor** — Perform a phasor simulation at fixed time steps specified by the **Sample time (s)** parameter, and at the frequency specified by the **Frequency (Hz)** parameter. The **Discrete phasor** solver uses simplified machine models that produce simulation results similar to transient stability software.

Sample time (s) — Sample time for discretization
50e-6 (default) | positive scalar

Sample time used to discretize the electrical circuit, in s. The powergui block displays the value of the sample time.

Dependencies

To enable this parameter, set **Simulation type** to **Discrete** or **Discrete phasor**.

Frequency (Hz) — Frequency for phasor simulation
60 (default) | positive scalar

Frequency, in Hz, for performing the phasor simulation of the model. The powergui block displays the value of the phasor frequency.

Dependencies

To enable this parameter, set **Simulation type** to **Phasor** or **Discrete phasor**.

Apps

Measurement and States Analyzer — Display steady-state voltages and currents, initial capacitor voltages, and inductor currents
button

Open the Measurements and States Analyzer app to display the steady-state voltages and currents of the model as well as the initial capacitor voltages and inductor currents.

Line Parameter Calculator — Compute RLC parameters of overhead transmission line
button

Open the Power Line Parameters app to compute RLC parameters of overhead transmission line from its conductor characteristics and tower geometry.

Cable Parameter Calculator — Compute RLC matrices and frequency-dependent parameters
button

Open the Power Cable Parameters app to compute RLC matrices and frequency-dependent parameters of cable arrangements.

Line Decoupler — Replace Distributed Parameters Line blocks with Decoupling Line blocks
button

Open the Specify Decoupling Lines app to replace selected Distributed Parameters Line blocks with Decoupling Line blocks.

Load Flow Analyzer — Perform load flow and initialize three-phase networks and machines
button

Open the Load Flow Tool dialog box to perform load flow analysis and initialize three-phase networks and machines so that the simulation starts in steady state. The Load Flow Tool uses the Newton-Raphson method to provide a robust and faster convergence solution compared to the Machine Initialization Tool. The Load Flow Tool offers most of the functionality of other tools available in the power utility industry. For more information, see `power_loadflow`.

Max iterations — Maximum iterations used by Load Flow Tool
50 (default) | positive integer

Maximum number of times the Load Flow Tool iterates until the P and Q powers mismatch at each bus is lower than the **PQ tolerance** parameter value (in pu/Pbase). The power mismatch is defined as the difference between the net power injected into the bus by generators and loads and the power transmitted on all links leaving that bus. For example, if the base power is 100 MVA and **PQ tolerance** is set to $1e-4$, the maximum power mismatch at all buses does not exceed 0.1 MW or 0.1 Mvar.

Frequency (Hz) — Frequency for load flow computations
60 (default) | positive scalar

Frequency, in Hz, used by the Load Flow Tool to compute the normalized Ybus network admittance matrix of the model and to perform the load flow calculations.

Base power (VA) — Base power used by Load Flow Tool
100e6 (default) | positive scalar

Base power, in VA, used by the Load Flow Tool to compute the normalized Ybus network admittance matrix in pu/Pbase and bus base voltages of the model, at the frequency specified by the **Frequency (Hz)** parameter for load flow computations.

To avoid a badly conditioned Ybus matrix, select a base power value in the range of nominal powers and loads of the model. For a transmission network with voltages ranging from 120 kV to 765 kV, a 100 MVA base is usually selected. For a distribution network or for a small plant consisting of generators, motors, and loads that have a nominal power in the range of hundreds of kilowatts, a 1 MVA base power is better adapted.

PQ tolerance (pu) — Tolerance between P and Q
 $1e-4$ (default) | positive scalar

Tolerance between P and Q when the Load Flow Tool stops to iterate.

Voltage units — Voltage units used by Load Flow Tool
kV (default) | V

Voltage units used by the Load Flow Tool to display voltages.

Power units — Power units used by Load Flow Tool
MW (default) | kW | W

Power units used by the Load Flow Tool to display powers.

Impedance Calculator — Display the impedance versus frequency

button

Open the app to display the impedance versus the frequency, as defined by the Impedance Measurement blocks. For more information, see `power_zmeter`.

FFT Analyzer — Perform Fourier analysis of signals

button

Open the FFT Analysis Tool dialog box to perform Fourier analysis of signals stored in a structure with time format. For more information, see `power_fftscope`.

For an example that uses the FFT Analysis tool, see “Performing Harmonic Analysis Using the FFT Tool”.

Linear System Analyzer — Time and frequency domain responses

button

Open the Linear System Analyzer dialog box to generate the state-space model of your system (if you have a Control System Toolbox license) or to view time and frequency domain responses. For more information, see `power_ltiview`.

Hysteresis Designer — Design hysteresis characteristic for the saturable core

button

Open the Hysteresis Design Tool to design a hysteresis characteristic for the saturable core of the Saturable Transformer block and the Three-Phase Transformer blocks (Two Windings and Three Windings). For more information, see `power_hysteresis`.

Block Customizer — Create custom blocks

button

Open the `power_customize` dialog box to create custom Simscape Electrical Specialized Power Systems blocks. For more information, see `power_customize`.

Report Generator — Generate report of steady-state variables, initial states, and machine load flow

button

Open the Generate Report Tool dialog box to generate a report of steady-state variables, initial states, and machine load flow for a model. For more information, see `power_report`.

Machine Initialization — Initialize three-phase networks containing three-phase machines

button

Open the Machine Initialization Tool dialog box to initialize three-phase networks containing three-phase machines so that the simulation starts in steady state. The Machine Initialization Tool offers simplified load flow features, but can still initialize machine initial currents of your models. For more information, see `power_loadflow`.

Preferences

The load flow parameters are for model initialization only. They do not have an impact on simulation performance.

Disable Specialized Power Systems warnings — Disable warnings during analysis and simulation

off (default) | on

Control display of Simscape Electrical Specialized Power Systems warnings during model analysis and simulation.

Display Specialized Power Systems compilation messages — Display echo messages during analysis

off (default) | on

Control display of the command-line echo messages during model analysis.

Use TLC file when in Accelerator Simulation Mode and for code generation — Use TLC state-space S-functions

off (default) | on

Select to use TLC state-space S-functions in accelerator mode and for code generation.

Dependencies

To enable this parameter, set **Simulation type** to Discrete.

Disable ideal switching — Control model switching devices as current sources

off (default) | on

Select to model switching devices as current sources. By default, this option is not selected, which corresponds to the recommended setting for most of your applications.

Modeling switches, such as circuit breakers or power electronic devices, as current sources implies that the on-state switch resistance (R_{on}) cannot be zero. In this modeling method, the switches cannot be connected in a series with an inductive circuit or with another switch or current source.

When this option is enabled, you must add a circuit (R or RC snubber) in parallel with the switches in your model so that the switches' off-state impedance has a finite value. If your real circuit does not use snubbers, or if you want to simulate ideal switches with no snubber, you must at least use resistive snubbers with a high resistance value to introduce a negligible leakage current. The drawback of introducing such high-impedance snubbers is that the large difference between the on-state and the off-state switch impedance produces a stiff state-space model.

Dependencies

To enable this parameter, set **Simulation type** to Continuous.

Disable snubbers in switching devices — Control snubber devices of power electronic and breaker blocks

off (default) | on

Select to disable the snubber devices of the power electronic and breaker blocks in your model.

Dependencies

To enable this parameter, set **Simulation type** to Continuous and clear **Disable ideal switching**.

Disable R_{on} resistances in switching devices — Control internal resistance of switches and power electronic devices

off (default) | on

Select to disable the internal resistance of switches and power electronic devices and to force the value to zero ohms.

Dependencies

To enable this parameter, set **Simulation type** to Continuous and clear **Disable ideal switching**.

Disable forward voltages in switching devices — Control internal forward voltage of power electronic devices

off (default) | on

Select to disable the internal forward voltage of power electronic devices and to force the value to zero volts.

Dependencies

To enable this parameter, set **Simulation type** to Continuous and clear **Disable ideal switching**.

Display circuit differential equations — Display differential equations in Diagnostic Viewer

off (default) | on

Select to display the differential equations of the model in the Diagnostic Viewer when the simulation starts.

Dependencies

To enable this parameter, set **Simulation type** to Continuous and clear **Disable ideal switching**.

Automatically handle Discrete solver and Advanced tab solver settings of blocks — Set discretization method automatically

off (default) | on

Select to automatically set the discretization method to Trapezoidal robust in models that contain at least one of the following blocks:

- Asynchronous Machine
- Permanent Magnet Synchronous Machine
- Simplified Synchronous Machine
- Synchronous Machine
- Surge Arrester
- Variable Resistor
- Variable Inductor
- Variable Capacitor
- Nonlinear Resistor
- Nonlinear Inductor
- Variable-Ratio Transformer
- Any transformer blocks that are modeling saturation

If your model contains none of these blocks, the discretization method is automatically set to Tustin/Backward Euler (TBE).

Dependencies

To enable this parameter, set **Simulation type** to Discrete.

Discrete solver — Discrete solver to use to simulate model

Tustin/Backward Euler (TBE) (default) | Tustin | Backward Euler

Set to Tustin/Backward Euler (TBE) to simulate the model using a combination of the Tustin and Backward Euler methods.

Set to Tustin to discretize the electrical model using the Tustin method. If you use this solver, you need to specify the R_s and C_s snubber values to avoid numerical oscillations when the firing pulses are blocked (when the bridge is operating as a rectifier). You can use the following formulas to compute the values of R_s and C_s :

$$R_s > 2 * T_s / C_s$$

$$C_s < P_n / (1000 * 2 * \pi * f * V_n^2)$$

where:

- P_n is the nominal power of the single-phase or three-phase converter, in VA.
- V_n is the nominal line-to-line AC voltage, in Vrms.
- f is the fundamental frequency, in Hz.
- T_s is the sample time, in s.

These values are derived from these criteria:

- The snubber leakage current at the fundamental frequency is less than 0.1% of the nominal current when power electronic devices are not conducting.
- The RC time constant of the snubbers is larger than two times the sample time, T_s .

Note The R_s and C_s values that guarantee the numerical stability of the discretized bridge can be different from the actual values used in the physical circuit.

Set to Backward Euler to discretize the electrical model using the Backward Euler method.

Dependencies

To enable this parameter, set **Simulation type** to **Discrete** and clear **Automatically handle Discrete solver and Advanced tab solver settings of blocks**.

Interpolate switching events — Increase simulation speed

off (default) | on

Select to increase simulation speed by enabling the solver to interpolate in discrete models using power electronics. When selected, the solver captures gate transitions of power electronic devices occurring between two sample times, allowing larger sample times (typically 20×) than you use with the standard solvers. For example, simulating a 5 kHz PWM converter with Tustin (no interpolation) or Tustin/Backward Euler normally requires a 1.0 μs sample time (sampling frequency = 200 × PWM frequency) to obtain a good resolution on pulse generation and guarantee accurate results. With interpolation enabled, using a sample time as large as 20 μs executes faster while preserving model accuracy.

When you select this option:

- Use a continuous pulse generator to guarantee the best accuracy on pulse generation. (Specify sample time = 0 in pulse-generation blocks.)

- In the Configuration Parameters dialog box, select a continuous, variable-step solver (ode45 or ode23tb with default settings). The continuous solver is required by the interpolation solver to compute the gate signals' time delays with respect to discrete sample times. The solver uses these pulse delays to interpolate between sample times and produce accurate results.

To see how interpolation increases accuracy and simulation speed, see the “Increased Accuracy and Simulation Speed Using Interpolation” example model.

Dependencies

To enable this parameter, set **Simulation type** to Discrete, clear **Automatically handle Discrete solver and Advanced tab solver settings of blocks**, and set **Discrete solver** to Tustin.

Use time-stamped gate signals — Capture evolution of states at different switching times

off (default) | on

When selected, the interpolation method computes model outputs at fixed sample times while taking into account switching events that occur between two sample times. The method receives pulses at fixed time steps and computes the time delays of the gate signals arriving within each time step. Computing the time delays enables the method to capture the evolution of states at different switching times.

When cleared, the interpolation method computes the time delays of the gate signal.

When selected, the block does not compute the time delays of gate signals. You then need to directly provide time-stamped gate signals to the switching devices in your model. For more information on the concept of time-stamped gate signals in Simscape Electrical Specialized Power Systems switching devices, see the “Increased Accuracy and Simulation Speed Using Interpolation” example.

Dependencies

To enable this parameter, set **Simulation type** to Discrete, clear **Automatically handle Discrete solver and Advanced tab solver settings of blocks**, set **Discrete solver** to Tustin, and select **Interpolate switching events**.

Store switching topologies — Increase simulation speed

off (default) | on

Select to increase simulation speed by enabling the solver to store and reuse matrix computation results.

Dependencies

To enable this parameter, set **Simulation type** to Continuous or Discrete and clear **Automatically handle Discrete solver and Advanced tab solver settings of blocks**.

Buffer size (MBytes) — Buffer size for saving computations

100 (default) | positive scalar

Buffer size for saving state-space matrix computations.

Dependencies

To enable this parameter, set **Simulation type** to Continuous or Discrete, clear **Automatically handle Discrete solver and Advanced tab solver settings of blocks**, and select **Store switching topologies**.

Start simulation with initial electrical states from — Initial state value
blocks (default) | steady | zero

If you select:

- **blocks** — Initial state values defined in blocks are used for the simulation
- **steady** — Forces all initial electrical state values to steady-state values
- **zero** — Forces all initial electrical state values to zero

Solver tolerance — Maximum solver error
1e-4 (default) | positive scalar

Largest acceptable solver error.

Dependencies

To enable this parameter, set **Simulation type** to Discrete and expand **Solver details for nonlinear elements**.

Maximum number of iterations — Iteration limit
100 (default) | positive scalar

Maximum number of iterations. Iterations stop when the **Solver tolerance** is achieved, or when the **Maximum number of iterations** is reached. A solution is usually found within 1 to 3 iterations. An error message is returned and simulation stops if a solution is not found when the maximum number of iterations is exceeded.

Dependencies

To enable this parameter, set **Simulation type** to Discrete and expand **Solver details for nonlinear elements**.

Continue Simulation if maximum number of iterations is reached — Limit maximum number of iterations
off (default) | on

Select to limit the maximum number of iterations. This parameter is used for real-time applications. Usually, limiting the number of iterations to 2 produces acceptable results.

Dependencies

To enable this parameter, set **Simulation type** to Discrete and expand **Solver details for nonlinear elements**.

Version History

Introduced before R2006a

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

`power_customize` | `power_fftscope` | `power_hysteresis` | `power_initstates` |
`power_lineparam` | `power_ltiview` | `power_loadflow` | `power_report` | `power_steadystate`
| `power_zmeter`

Power (Phasor)

Compute active and reactive powers using voltage and current phasors



Library

Simscape / Electrical / Specialized Power Systems / Sensors and Measurements

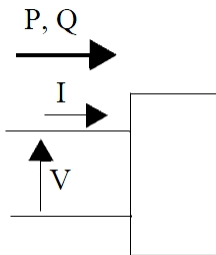
Description

The Power (Phasor) block computes the active power P and reactive power Q of a pair of voltage-current signals. P and Q are computed as follows:

$$P + jQ = \frac{1}{2}(V \times I^*)$$

I^* = complex conjugate of I .

Sign conventions are defined as



Inputs and Outputs

V

The phasor (complex signal) voltage, in volts peak.

I

The phasor (complex signal) current, in amperes peak.

P

Returns the active power, in watts.

Q

Returns the active power, in vars.

Characteristics

Direct Feedthrough	Yes
Dimensionalized	Yes

Examples

The power_PhasePowerMeasurements example shows the use of the Power (Phasor) block.

Version History

Introduced in R2013a

Power (PLL-Driven, Positive-Sequence)

Compute positive-sequence active and reactive powers



Library

Simscape / Electrical / Specialized Power Systems / Sensors and Measurements

Description

The Power (PLL-Driven, Positive-Sequence) block computes the positive-sequence active power P (in watts) and reactive power Q (in vars) of a periodic set of three-phase voltages and currents. To perform this computation, the block first computes the positive sequence of the input voltages and currents with a running window over one cycle of the fundamental frequency given by input 1. The reference frame required for the computation is given by the input 2. The first two inputs are normally connected to the outputs of a PLL block.

These formulas are then evaluated:

$$P = 3 \times \frac{|V_1|}{\sqrt{2}} \times \frac{|I_1|}{\sqrt{2}} \times \cos(\varphi)$$

$$Q = 3 \times \frac{|V_1|}{\sqrt{2}} \times \frac{|I_1|}{\sqrt{2}} \times \sin(\varphi)$$

$$\varphi = \angle V_1 - \angle I_1$$

V_1 is the positive-sequence component of input V_{abc} . I_1 is the positive-sequence component of input I_{abc} .

With these formulas, a current flowing into an RL circuit produces a positive P and a positive Q .

As this block uses a running average window, one cycle of simulation must complete before the outputs give the correct value. For the first cycle of simulation, the output is held constant using the values specified by the **Voltage initial input** and **Current initial input** parameters.

Parameters

Initial frequency (Hz)

Specify the frequency of the first cycle of simulation. Default is 60.

Minimum frequency (Hz)

Specify the minimum frequency value to determine the buffer size of the Variable Time Delay block used by the block to compute the phasors. Default is 45.

Voltage initial input [Mag, Phase (degrees)]

Specify the initial positive-sequence magnitude and phase (relative to the PLL phase), in degrees, of the voltage signals. Default is [1, 0].

Current initial input [Mag, Phase (degrees)]

Specify the initial positive-sequence magnitude and phase (relative to the PLL phase), in degrees, of the current signals. Default is [1, 0].

Sample time

Specify the sample time of the block, in seconds. Set to 0 to implement a continuous block. Default is 0.

Inputs and Outputs**Freq**

Fundamental frequency (Hz) required by the computation. This input is normally connected to the output Freq of a PLL block.

wt

Angle of the reference frame (rad/s) required for the computation. This input is normally connected to the output wt of a PLL block.

Vabc

The vectorized signal of the three [a b c] voltage sinusoidal signals. Typical input signals are voltages measured by the Three-Phase V-I Measurement block.

Iabc

The vectorized signal of the three [a b c] current sinusoidal signals. Typical input signals are currents measured by the Three-Phase V-I Measurement block.

P

Positive-sequence active power (watts)

Q

Positive-sequence reactive power (vars)

Characteristics

Sample Time	Specified in the Sample Time parameter Continuous if Sample Time = 0
Scalar Expansion	No
Dimensionalized	No

Examples

The `power_PowerPLLDriivenPositiveSequence` model shows how the block evaluates the positive-sequence active and reactive powers of a voltage source connected to a three-phase load. It shows that the block outputs accurate values for P and Q even if the fundamental frequency of the voltage supply (containing harmonics) varies during the simulation.

The model sample time is parameterized by the T_s variable set to a default value of $50\text{e-}6$ s. Set T_s to 0 in the command window and change the **Simulation type** parameter of the Powergui block to Continuous to simulate the model in continuous mode.

Version History

Introduced in R2013a

Power (Positive-Sequence)

Compute positive-sequence active and reactive powers



Library

Simscape / Electrical / Specialized Power Systems / Sensors and Measurements

Description

The Power (Positive-Sequence) block computes the positive-sequence active power P (in watts) and reactive power Q (in vars) of a periodic set of three-phase voltages and currents. To perform this computation, the block first computes the positive sequence of the input voltages and currents with a running window over one cycle of the specified fundamental frequency. These formulas are then evaluated:

$$P = 3 \times \frac{|V_1|}{\sqrt{2}} \times \frac{|I_1|}{\sqrt{2}} \times \cos(\varphi)$$

$$Q = 3 \times \frac{|V_1|}{\sqrt{2}} \times \frac{|I_1|}{\sqrt{2}} \times \sin(\varphi)$$

$$\varphi = \angle V_1 - \angle I_1$$

V_1 is the positive-sequence component of input V_{abc} . I_1 is the positive-sequence component of input I_{abc} .

A current flowing into an RL circuit produces a positive P and a positive Q .

As this block uses a running average window, one cycle of simulation must complete before the outputs give the correct value. For the first cycle of simulation, the output is held constant using the values specified by the **Voltage initial input** and **Current initial input** parameters.

Parameters

Fundamental frequency (Hz)

Specify the fundamental frequency, in hertz, of the input signals. Default is 60.

Voltage initial input [Mag, Phase (degrees)]

Specify the voltage initial magnitude and phase used by the block to compute the constant output for the first cycle of simulation. Default is [0, 0].

Current initial input [Mag, Phase (degrees)]

Specify the current initial magnitude and phase used by the block to compute the constant output for the first cycle of simulation. Default is [0, 0].

Sample time

Specify the sample time of the block, in seconds. Set to 0 to implement a continuous block. Default is 0.

Inputs and Outputs

Vabc

The three-phase voltage signal.

Iabc

The three-phase current signal.

P

The positive-sequence active power, in watts.

Q

The positive-sequence reactive power, in vars.

Characteristics

Sample Time	Specified in the Sample Time parameter Continuous if Sample Time = 0
Scalar Expansion	No
Dimensionalized	No

Examples

The `power_ThreePhasePower` model compares the outputs of the Power (Positive-Sequence) block with the Power (3ph, Instantaneous) and the Power (dq0, Instantaneous) block. The Power (Positive-Sequence) block gives ripple-free accurate results even with the presence of zero and negative sequences in the voltage supply.

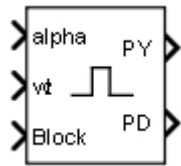
The model sample time is parameterized by the `Ts` variable set to a default value of 50e-6 s. Set `Ts` to 0 in the command window and change the **Simulation type** parameter of the Powergui block to Continuous to simulate the model in continuous mode.

Version History

Introduced in R2013a

Pulse Generator (Thyristor)

Generate pulses for twelve-pulse and six-pulse thyristor converters



Libraries:

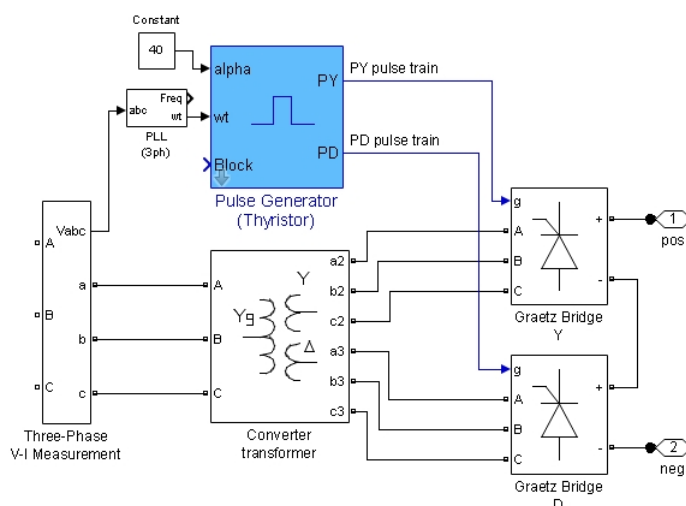
Simscape / Electrical / Specialized Power Systems / Power Electronics / Power Electronics Control

Description

The Pulse Generator (Thyristor) block generates two pulse trains. These control a twelve-pulse thyristor converter made of two three-phase full-wave thyristor bridges (also called Graetz bridges). In steady-state condition, each pulse train consists of six equidistant square pulses with 60 degrees shift between them.

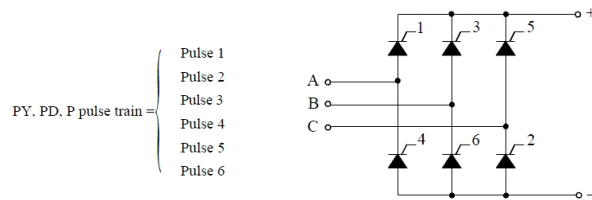
The first set of pulses (PY) is sent to the six-pulse bridge connected to the wye (Y) secondary winding of the Y/Y/Delta converter transformer. The second set of pulses (PD) is sent to the six-pulse bridge connected to the delta secondary winding of the converter transformer. The PD pulses can be set to lead or lag the PY pulses by 30 electrical degrees, depending on the configuration of the delta connection of the converter transformer.

The following figure shows an example of a Pulse Generator (Thyristor) block connected to a twelve-pulse thyristor converter.



The Pulse Generator (Thyristor) block can be programmed to control a six-pulse thyristor converter made of one three-phase full-wave thyristor bridge. In this configuration, the PD pulse train is not generated and the block outputs only the PY pulse train. The pulse train, renamed P, is suitable for a Graetz bridge connected to a converter transformer, with no phase shift between primary and secondary windings.

The pulse ordering in the pulse trains corresponds to the natural order of commutation of a three-phase thyristor bridge, as shown in the following figure.

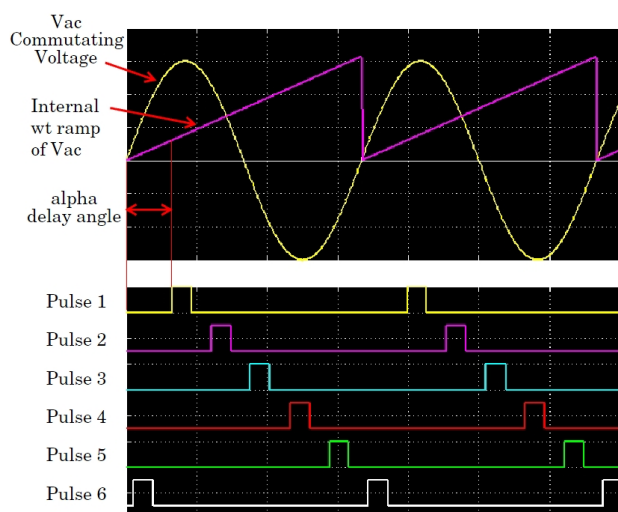


The table below gives the commutating voltages for the thyristors according to the transformer connection. The transformer connection reflects the phase shift between the AC source and the thyristors.

Thyristor to be ignited	1	2	3	4	5	6
Thyristor to be extinguished	5	6	1	2	3	4
Commutating voltage for a Y-Y connection	V_{ac}	V_{bc}	V_{ba}	V_{ca}	V_{cb}	V_{ab}
for a Y-D1 (lagging)	$-V_c$	V_b	$-V_a$	V_c	$-V_b$	V_a
for a Y-D11 (leading)	V_a	$-V_c$	V_b	$-V_a$	V_c	$-V_b$

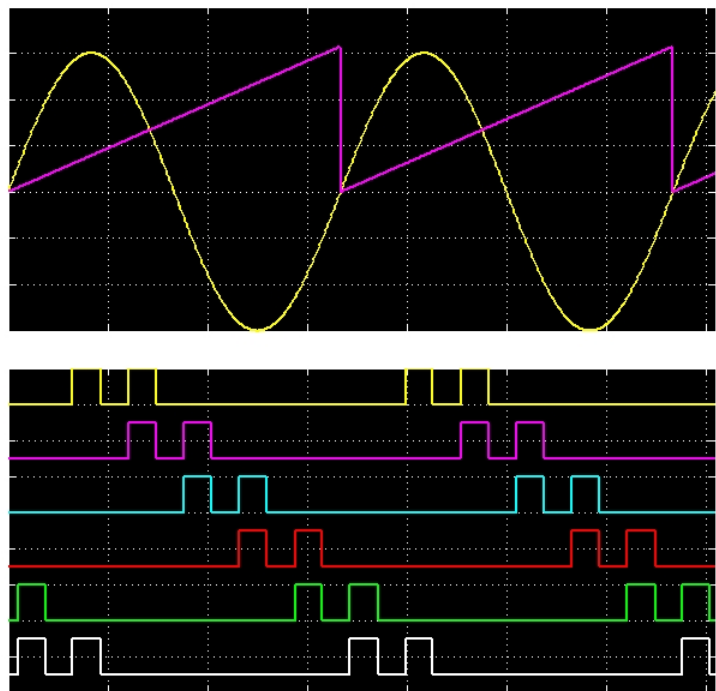
The Pulse Generator (Thyristor) block is controlled by the alpha angle reference signal and by the synchronization signal ωt . The ωt signal is an angle varying between 0 and 2π radians, synchronized on zero crossings of the fundamental (positive-sequence) of phase A of the primary voltage of the converter transformer. The ωt signal is normally obtained from a phase locked loop (PLL) system.

The Pulse Generator (Thyristor) block generates internal ωt ramps to control the pulses. The alpha delay angle is expressed in electrical degrees by which the pulse is delayed relative to angle zero of its commutating voltage. The figure shows how the PY pulse train is generated.



The Pulse Generator (Thyristor) block can be configured to work in double-pulsing mode. In this mode two pulses are sent to each thyristor: a first pulse when the alpha angle is reached, and then a

second pulse 60 degrees later, when the next thyristor is fired. This figure shows the double pulses in a PY pulse train.



Characteristics

Sample Time	Specified in the Sample Time parameter
Scalar Expansion	No
Dimensionalized	No
Zero-Crossing Detection	Yes

Examples

The `power_PulseGenerator` example shows various uses of the pulse generator.

The model sample time is parameterized with the variable `Ts` (default value `Ts = 50e-6`). To simulate continuous mode, specify `Ts = 0` in the MATLAB Command Window.

Port

Input

alpha — Alpha delay angle
scalar

Alpha delay angle signal, in degrees. You can connect this input to a Constant block or to a controller system to control the pulses of the generator.

wt — Angle synchronized on zero crossings of fundamental
scalar in the range `[0, 2*pi]`

Angle, in radians, varying between 0 and 2π , synchronized on zero crossings of the fundamental (positive-sequence) of phase A of the primary voltage of the converter transformer. The wt signal is normally obtained from a phase locked loop (PLL) system.

Block — Generator operation flag

1 | 0

When you set this input to true (Boolean 1), the pulses are disabled and the operation of the generator is blocked.

Output

P — Output for Y secondary winding of converter

scalar | vector

Outputs the six-pulse signals to be sent to the six-pulse thyristor converter connected to the Y secondary winding of the converter transformer. The output is named PY when the **Generator type** parameter is set to 12-pulse.

Dependencies

To enable this parameter, set the **Generator type** parameter to 6-pulse.

PY — Output for Y secondary winding of converter

scalar | vector

Outputs the six-pulse signals to be sent to the six-pulse thyristor converter connected to the Y secondary winding of the converter transformer. The output is named P when the **Generator type** parameter is set to 12-pulse.

Dependencies

To enable this parameter, set the **Generator type** parameter to 12-pulse.

PD — Output for delta secondary winding of converter

scalar | vector

Outputs the six-pulse signals to be sent to the six-pulse thyristor converter connected to the Delta (D) secondary winding of the converter transformer.

Dependencies

To enable this parameter, set the **Generator type** parameter to 12-pulse.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Generator type — Type of thyristor converter to control

6-pulse (default) | 12-pulse

Type of thyristor converter to control. When you set this parameter to 6-pulse, the PD pulse train output is not generated, and the PY output is renamed to P.

Delta winding connection — Type of delta secondary winding connection of converter transformer
D1(lagging) (default) | D11(leading)

Type of delta secondary winding connection of the converter transformer. When you set this parameter to D1(lagging), the PD pulses lag the PY pulses by 30 degrees. When you set this parameter to D11(leading), the PD pulses lead the PY pulses by 30 degrees.

Pulse width (deg) — Width of pulses
60 (default) | scalar

Width of the pulses, in degrees.

Double pulsing — Double pulsing option
off (default) | on

Whether the Pulse Generator (Thyristor) block implements the double-pulsing technique.

Sample time — Block sample time
0 (default) | nonnegative scalar

Sample time of the block, in seconds. Set to 0 to implement a continuous block.

Version History

Introduced in R2013a

Extended Capabilities

C/C++ Code Generation

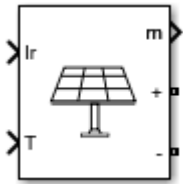
Generate C and C++ code using Simulink® Coder™.

See Also

PWM Generator

PV Array

Implement PV array modules



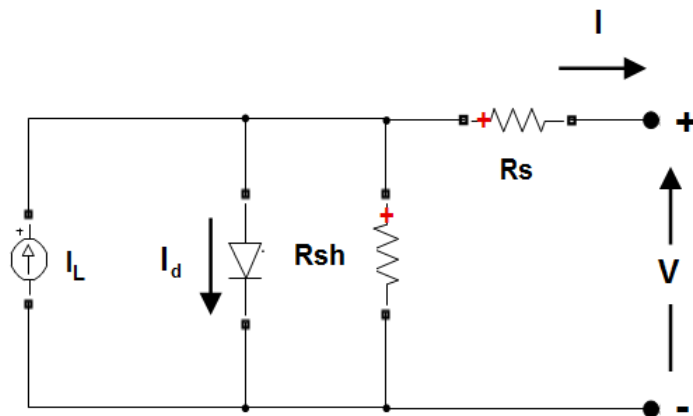
Libraries:

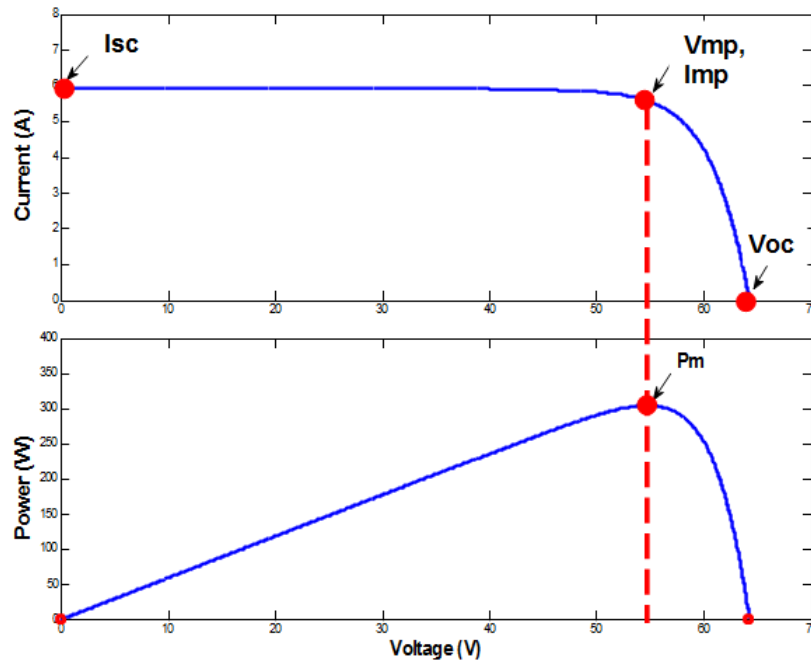
Simscape / Electrical / Specialized Power Systems / Sources

Description

The PV Array block implements an array of photovoltaic (PV) modules. The array is built of strings of modules connected in parallel, each string consisting of modules connected in series. This block allows you to model preset PV modules from the National Renewable Energy Laboratory (NREL) System Advisor Model (2018) as well as PV modules that you define.

The PV Array block is a five-parameter model using a light-generated current source (I_L), diode, series resistance (R_s), and shunt resistance (R_{sh}) to represent the irradiance- and temperature-dependent I-V characteristics of the modules.





The diode I-V characteristics for a single module are defined by the equations

$$I_d = I_0 \left[\exp\left(\frac{V_d}{V_T}\right) - 1 \right]$$

$$V_T = \frac{kT}{q} \times nI \times N_{cell}$$

where:

I_d Diode current (A)

V_d Diode voltage (V)

I_0 Diode saturation current (A)

nI Diode ideality factor, a number close to 1.0

k Boltzman constant = 1.3806×10^{-23} J.K⁻¹

q Electron charge = 1.6022×10^{-19} C

T Cell temperature (K)

N_{cell} Number of cells connected in series in a module

Ports

Input

Ir — Control signal defining Irradiance applied to solar panels, W/m²
 scalar in the range [0, 1000]

Control signal defining that irradiance applied to solar panels, specified as a scalar in the range [0, 1000], in W/m².

T — Control signal defining temperature of cells, degrees Celsius
scalar

Control signal defining temperature of cells, specified as a scalar, in degrees Celsius. The input can be a finite negative, zero, or positive value.

Dependencies

To enable this port, clear the **Robust discrete model** parameter.

Output

m — Measurements
five-element vector

Measurements, returned as a five-element vector. You can select these signals using the Bus Selector block in the Simulink library.

Signal	Signal Name	Definition
1	V_PV	PV array voltage (V)
2	I_PV	PV array current (A)
3	I_diode	diode current (A)
4	Irradiance (W/m2)	Irradiance (W/m2)
5	Temperature (deg C)	Temperature (degrees C)

Conserving

+ — Positive terminal
specialized electrical

Specialized electrical conserving port associated with the positive terminal of the PV array.

- — Negative terminal
specialized electrical

Specialized electrical conserving port associated with the negative terminal of the PV array.

Parameters

Parameters

Parallel strings — Strings of series-connected modules connected in parallel
40 (default) | positive integer

Number of strings of series-connected modules that are connected in parallel.

Series-connected modules per string — PV modules connected in series in each string
10 (default) | positive integer

Number of PV modules connected in series in each string.

Display I-V and P-V characteristics of ... — Characteristics of one module or whole array
array @ 1000 W/m² & specified temperatures (default) | one module @ 25 deg.C &
specified irradiances | array @ 25 deg.C & specified irradiances

Display the I-V and P-V characteristics of one module or of the whole array, for variable irradiance or for variable temperatures. The selected characteristics display when you click **Plot**.

Irradiances (W/m²) — Vector of irradiances
[1000 500 100] (default) | vector of 1 or more elements

Vector of irradiances in W/m². The vector must have at least one element. The **Plot** button will show I-V and P-V module characteristics for every irradiance listed in the vector.

Dependencies

To enable this parameter, set **Display I-V and P-V characteristics of** to one module @ 25 deg.C & specified irradiances or array @ 25 deg. C & specified irradiances.

T_cell (deg. C) — Temperatures
[45 25] (default) | vector of 1 or more elements

Vector of temperatures in degrees Celsius. The vector must have at least one element. The **Plot** button will show I-V and P-V module characteristics for every temperature operating point listed in the vector.

Dependencies

To enable this parameter, set **Display I-V and P-V characteristics of** to array @ 1000 W/m² & specified temperatures.

Module — User-defined or preset PV module
User-defined (default) | A10Green Technology A10J-S72-175 | A10Green Technology A10J-S72-180 | A10Green Technology A10J-S72-185 | A10Green Technology A10J-M60-220 | ...

Select User-defined or a preset PV module from the NREL System Advisory Model database. Over 10,000 modules are listed from main manufacturers, sorted in alphabetical order. The NREL database includes manufacturer datasheets measured under standard test conditions (STC), where the irradiance is 1000 W/m² and the temperature is 25 degrees Celsius.

Note In releases earlier than R2021a, the module list includes manufacturer modules that are no longer present in the list. If you use one of these modules, the block sets the **Module** parameter to User-defined and does not change the module data parameter values.

When you select a module, these parameters are updated with data from the NREL database:

- **Cells per module (Ncell)**
- **Open circuit voltage Voc (V)**
- **Short-circuit current Isc (A)**
- **Voltage at maximum power point Vmp (V)**
- **Current at maximum power point Imp (A)**

- **Temperature coefficient of Voc (%/deg.C)**
- **Temperature coefficient of Isc (%/deg.C)**

The function computes these five corresponding model parameters using an optimization function and displays them on the right side of the dialog box.

- **Light-generated current IL (A)**
- **Diode saturation current IO (A)**
- **Diode ideality factor**
- **Shunt resistance Rsh (ohms)**
- **Series resistance Rs (ohms)**

When you select User-defined, enter your own specifications for the module data parameters. When you apply the changes, the function computes the five model parameters.

Maximum Power (W) — Power at maximum power point

213.15 (default) | positive scalar

Power obtained at the maximum power point. The maximum power is calculated as $P_{max} = V_{mp} \times I_{mp}$, where:

- P_{max} is the value of the **Maximum Power (W)** parameter.
- V_{mp} is the value of the **Voltage at maximum power point Vmp (V)** parameter.
- I_{mp} is the value of the **Current at maximum power point Imp (A)** parameter.

This parameter is read-only.

Cells per module (Ncell) — Cells per module

60 (default) | positive integer

Number of cells per module.

Open circuit voltage Voc (V) — Voltage when circuit open

36.3 (default) | positive scalar

Voltage at 25 degrees Celsius obtained when array terminals are left open.

Short-circuit current Isc (A) — Current when short circuited

7.84 (default) | positive scalar

Current at 25 degrees Celsius obtained when array terminals are short circuited.

Voltage at maximum power point Vmp (V) — Maximum power point voltage

29 (default) | positive scalar

Voltage at maximum power point.

Current at maximum power point Imp (A) — Maximum power point current

7.35 (default) | positive scalar

Current at maximum power point.

Temperature coefficient of Voc (%/deg.C) — Variation of Voc as function of temperature

-0.36099 (default) | scalar

Defines variation of V_{oc} as a function of temperature. The open-circuit voltage at temperature T is obtained as

$$V_{oc\ T} = V_{oc} (1 + \beta_{V_{oc}} (T - 25)),$$

where V_{oc} is the open-circuit voltage at 25 degrees C, V_{ocT} is the open-circuit voltage at temperature T (in degrees C), $\beta_{V_{oc}}$ is the temperature coefficient (in %/degrees C), and T is the temperature in degrees C.

Temperature coefficient of I_{sc} (%/deg.C) — Variation of I_{sc} as function of temperature

0.102 (default) | positive scalar

Defines variation of I_{sc} as a function of temperature. The short-circuit current at temperature T is obtained as

$$I_{sc\ T} = I_{sc} (1 + \alpha_{I_{sc}} (T - 25)),$$

where I_{sc} is the short-circuit current at 25 degrees C, I_{scT} is the short-circuit current at temperature T (in degrees C), $\alpha_{I_{sc}}$ is the temperature coefficient (in %/degrees C) and T is the temperature in degrees C.

Light-generated current I_L (A) — Current for one module under STC

no default

Current for one module under STC, flowing out of the controllable current source that models the light-generated current. An optimization function determines this parameter to fit the module data. This parameter is read-only.

Diode saturation current I_0 (A) — Saturation current of diode

no default

Saturation current of the diode modeling the PV array for one module under STC. An optimization function determines this parameter to fit the module data. This parameter is read-only.

Diode ideality factor — Ideality factor of diode modeling PV array

no default

Ideality factor of the diode modeling the PV array. An optimization function determines this parameter to fit the module data. This parameter is read-only.

Shunt resistance R_{sh} (ohms) — Shunt resistance of the model

no default

Shunt resistance of the model for one module under STC. An optimization function determines this parameter to fit the module data. This parameter is read-only.

Series resistance R_s (ohms) — Series resistance of the model

no default

Series resistance of the model for one module under STC. An optimization function determines this parameter to fit the module data. This parameter is read-only.

Advanced

Robust discrete model — Iterate to resolve algebraic loop

off (default) | on

When selected, the robust solver iterates to resolve the algebraic loop in the internal diode model. The maximum number of iterations is specified in the **Preferences** tab of the powergui block, in the **Solver details for nonlinear elements** section. For real-time applications, you may need to limit the number of iterations. The robust solver is the recommended method for discretizing the PV Array model. When selected, the cell temperature is specified by the **Cell temperature, in deg.C. (input 2 is disabled)** parameter, and the **T** port is hidden.

When cleared, the algebraic loop is controlled by the **Break algebraic loop in internal model** parameter.

For more information on what method to use in your application, see “Simulating Discretized Electrical Systems”.

Dependencies

To enable this parameter, set the **Simulation type** parameter of the powergui block to Discrete.

Cell temperature, in deg.C. (input 2 is disabled) — Specify cell temperature
25 (default)

Specify the cell temperature in degrees Celsius.

Dependencies

To enable this parameter, select **Robust discrete model**.

Break algebraic loop in internal model — Break algebraic loop of internal diode model
on (default) | off

Select to break the algebraic loop of the internal diode model. In discrete models, the parameter is visible only if the **Robust discrete model** parameter is cleared.

If you use the block in a continuous system, the algebraic loop is cut by using a first-order filter.

If you use the block in a discrete system, the algebraic loop is cut by using a one-simulation-step time delay. This approach can cause numerical oscillations if the sample time of the simulation is too large. For example, a PV Array block connected to an average model of power converter can run with a sample time as large as $50\text{e-}6$ seconds. In this case, the algebraic loop is required to get an iterative, accurate solution for the highly nonlinear diode characteristics. If the sample time of the simulation is small, such as $1\text{e-}6$ seconds for a PV Array block connected to a detailed power electronic converter using real switches and a 5 kHz PWM inverter, select this parameter to speed up simulation and get accurate resolution.

Dependencies

To enable this parameter, clear **Robust discrete model**.

Break loop time constant (s) — Time constant of filter block that breaks algebraic loop
 $1\text{e-}6$ (default) | positive scalar

When the **Break algebraic loop in internal model** parameter is selected, the algebraic loop is cut by using a first-order filter when the simulation is continuous. If you use the block in a discrete system, the algebraic loop is cut by using a one-simulation-step time delay.

Dependencies

To enable this parameter, select **Break algebraic loop in internal model** and, in the powergui block, set **Simulation type** to Continuous.

Measurement filter time constant (s) — Measurement filters

5e-5 (default) | positive scalar

When the **Break algebraic loop in internal model** parameter is cleared, measurement filters are used inside the measurement output, m, for the PV array voltage (signal 1) and PV array current (signal 2).

Dependencies

To enable this parameter, clear **Robust discrete model** and **Break algebraic loop in internal model**.

Version History

Introduced in R2015a

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

Topics

"250-kW Grid-Connected PV Array"

"400-kW Grid-Connected PV Farm"

"Partial Shading of a PV Module"

"Single-Phase, 240 Vrms, 3500 W Transformerless Grid-Connected PV Array"

PWM Generator

Generate pulses for carrier-based two-level pulse width modulator (PWM) in converter bridge



Library

powerlib_extras/Control Blocks, powerlib_extras/Discrete Control Blocks

Note The Pulse & Signal Generators section of the Control and Measurements library contains the PWM Generator (2-level) block. This is an improved version of the PWM Generator block. The new block features a mechanism that eliminates duplicate continuous and discrete versions of the same block by basing the block configuration on the simulation mode. If your legacy models contain the PWM Generator block, they will continue to work. However, for best performance, use the PWM Generator (2-level) block in your new models.

Description

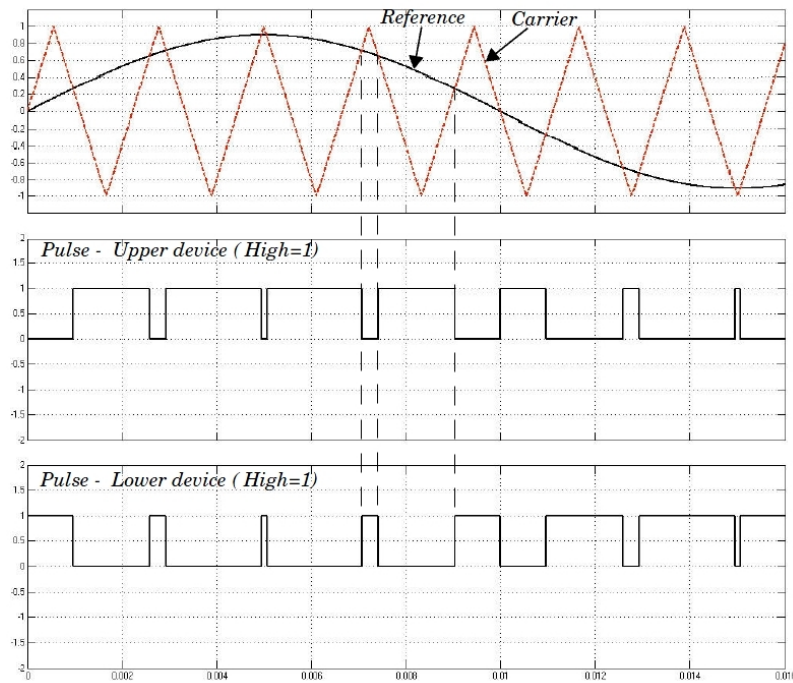
The PWM Generator block generates pulses for carrier-based pulse width modulation (PWM) converters using two-level topology. The block can be used to fire the forced-commutated devices (FETs, GTOs, or IGBTs) of single-phase, two-phase, three-phase, two-level bridges, or a combination of two three-phase bridges.

The pulses are generated by comparing a triangular carrier waveform to a reference modulating signal. The modulating signals can be generated by the PWM generator itself, or they can be a vector of external signals connected at the input of the block. One reference signal is required to generate the pulses for a single- or a two-arm bridge, and three reference signals are required to generate the pulses for a three-phase, single or double bridge.

The amplitude (modulation), phase, and frequency of the reference signals are set to control the output voltage (on the AC terminals) of the bridge connected to the PWM Generator block.

The two pulses firing the two devices of a given arm bridge are complementary. For example, pulse 4 is low (0) when pulse 3 is high (1). This is illustrated in the next two figures.

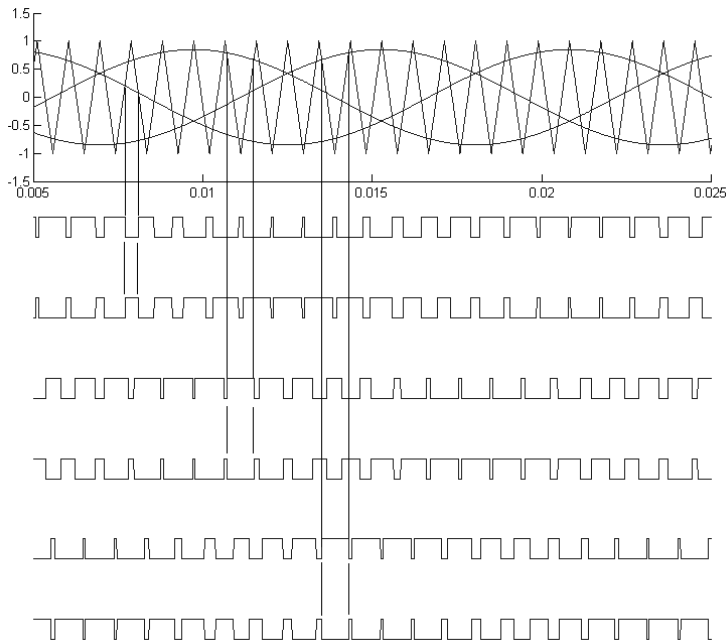
The following figure displays the two pulses generated by the PWM Generator block when it is programmed to control a one-arm bridge.



The triangular carrier signal is compared with the sinusoidal modulating signal. When the modulating signal is greater than the carrier pulse 1 is high (1) and pulse 2 is low (0).

For a single-phase two-arm bridge the modulating signal used for arm 2 is the negative of modulating signal used for arm 1 (180 degrees phase shift). For a three-phase six-arm bridge the three modulating signals used for bridge 2 are the negative of the modulating signals applied to bridge 1.

The following figure displays the six pulses generated by the PWM Generator block when it is programmed to control a three-arm bridge.

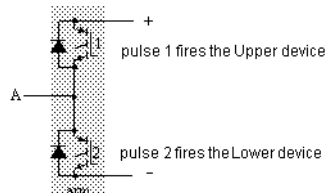


Parameters

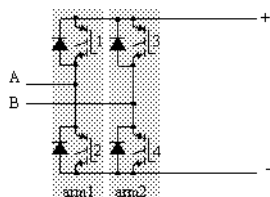
Generator Mode

Specify the number of pulses to generate. The number of pulses is proportional to the number of bridge arms to fire.

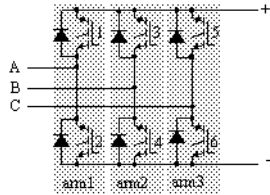
Select **1-arm bridge (2 pulses)** to fire the self-commutated devices of a one-arm bridge. Pulse 1 fires the upper device and pulse 2 fires the lower device.



Select **2-arm bridge (4 pulses)** to fire the self-commutated devices of a two-arm bridge. Four pulses are generated for a two-arm bridge. Pulses 1 and 3 fire the upper devices of the first and second arm. Pulses 2 and 4 fire the lower devices.



Select 3-arm bridge (6 pulses) to fire the self-commutated devices of a three-arm bridge. Pulses 1, 3, and 5 fire the upper devices of the first, second, and third arms. Pulses 2, 4, and 6 fire the lower devices.



Select Double 3-arm bridges (12 pulses) to fire the self-commutated devices of two six-pulse bridges connected in a twelve-pulse bridge configuration. The first six pulses (1 to 6) fire the six devices of the first three-arm bridge and the last six pulses (7 to 12) fire the six devices of the second three-arm bridge.

Carrier frequency

The frequency, in hertz, of the carrier triangular signal.

Internal generation of modulating signal(s)

If selected, the modulating signal is generated by the block. Otherwise, external modulating signals are used for pulse generation. The default is unselected.

Modulation index ($0 < m < 1$)

The **Modulation index** parameter is visible only if the **Internal generation of modulating signal (s)** parameter is selected.

The amplitude of the internal sinusoidal modulating signal. The **Modulation index** must be greater than 0, and lower than or equal to 1. This parameter is used to control the amplitude of the fundamental component of the output voltage of the controlled bridge. This parameter is only visible when **Internal generation of modulating signal(s)** is selected.

Frequency of output voltage

The **Frequency of output voltage (Hz)** parameter is visible only if the **Internal generation of modulating signal (s)** parameter is selected. This parameter is only visible when **Internal generation of modulating signal(s)** is selected.

The frequency, in hertz, of the internal modulating signals. This parameter is used to control the fundamental frequency of the output voltage of the controlled bridge.

Phase of output voltage

The **Phase of output voltage** parameter is visible only if the **Internal generation of modulating signal (s)** parameter is selected. This parameter is only visible when **Internal generation of modulating signal(s)** is selected.

The phase, in degrees, of the internal modulating signal. This parameter is used to control the phase of the fundamental component of the output voltage of the controlled bridge.

Inputs and Outputs

Signal(s)

The input is not visible when **Internal generation of modulating signal (s)** is selected.

The input is the vector of modulating signals when **Internal generation of modulating signal** is not selected. Connect this input to a single-phase sinusoidal signal when the block is used to control a single- or a two-arm bridge, or to a three-phase sinusoidal signal when the PWM Generator block is controlling one or two three-phase bridges.

Pulses

The output contains the two, four, six, or twelve pulse signals used to fire the self-commutated devices (MOSFETs, GTOs, or IGBTs) of single-phase, two-phase, or three-phase bridges or a combination of two three-phase bridges.

Version History

Introduced before R2006a

See Also

PWM Generator (2-level) | PWM Generator (3-level) | Universal Bridge

PWM Generator (2-Level)

Generate pulses for PWM-controlled 2-level converter

**Libraries:**

Simscape / Electrical / Specialized Power Systems / Power Electronics / Power Electronics Control

Description

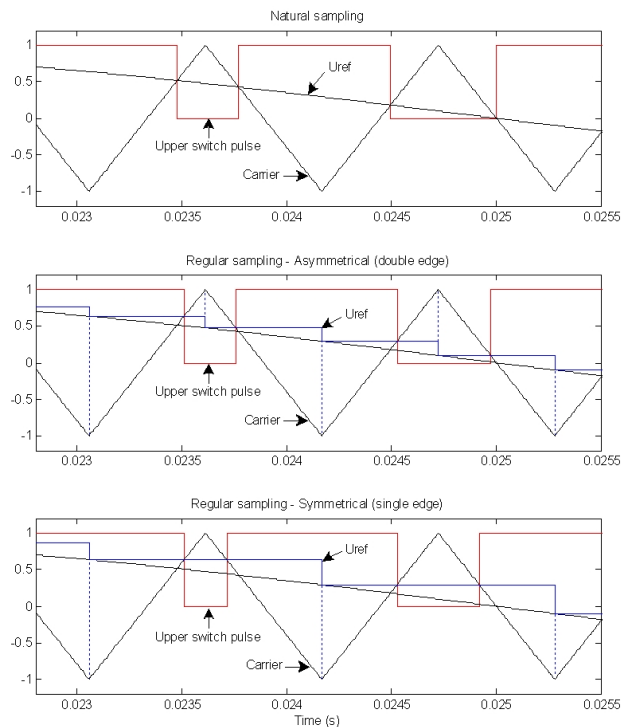
The PWM Generator (2-Level) block generates pulses for carrier-based pulse width modulation (PWM) converters using two-level topology. The block can control switching devices (FETs, GTOs, or IGBTs) of three different converter types: single-phase half-bridge (1 arm), single-phase full-bridge (2 arms), or three-phase bridge (3 arms).

The reference signal (Uref input), also called modulating signal, is compared with a symmetrical triangle carrier. When the reference signal is greater than the carrier, the pulse for the upper switching device is high (1), and the pulse for the lower device is low (0).

To control a single-phase full-bridge device, you can select unipolar or bipolar PWM modulation. Using the unipolar modulation, each arm is controlled independently. A second reference signal is internally generated by phase-shifting the original reference signal by 180 degrees. Using the bipolar modulation, the state of the lower switching device of the second arm is the same as the state of the upper switch of the first arm, and the state of the upper switch of the second arm is the same as the state of the lower switch of the first arm. The unipolar modulation produces better quality AC waveform, but the bipolar modulation produces very low-varying common-mode voltage.

The figure describes the three techniques to sample the reference signal Uref. The natural sampling technique models the behavior of an analog implementation of a PWM generator. Using the two regular sampling techniques, Uref can be sampled twice at both the valley and the peak of the carrier or only once at the valley of the carrier. The former is referred to as asymmetrical sampling or double-edge technique. The latter is called symmetrical sampling or single-edge technique.

Reference Signal Sampling Techniques



Examples

The `power_PWMGenerator2Level` model uses a simple circuit to illustrate the operation of the PWM Generator (2-Level). Run the simulation and use the FFT Analysis tool of the Powergui block to see the harmonics and the THD value of the voltages produced by the three-phase two-level converter.

The model sample time is parameterized by the `Ts` variable set to a default value of 50×10^{-6} s. Set `Ts` to 0 in the command window and change the **Simulation type** parameter of the Powergui block to Continuous to simulate the model in continuous mode.

Ports

Input

Uref — Reference signal used to generate output pulses

Vectorized reference signal used to generate the output pulses. Connect this input to a single-phase sinusoidal signal when the block is used to control a single-phase half- or full-bridge converter, or to a three-phase sinusoidal signal when the PWM Generator block is controlling a three-phase bridge converter. For linear operation of this block, the magnitude of **Uref** must be between -1 and $+1$.

Dependencies

To enable this port, clear the **Internal generation of modulating signal (s)** parameter.

wt — External reference signal used to synchronize carrier

External reference signal used to synchronize the carrier.

Dependencies

To enable this port, set **Carrier: Mode of operation** to Unsynchronized.

Output

P — Pulse signals used to fire self-commutated devices

Output contains the two, four, or six pulse signals used to fire the self-commutated devices (MOSFETs, GTOs, or IGBTs) of a one-, two- or three-arm converter.

m — Measurement output

Measurement output that returns the carrier signal used to determine the output pulses and the sampled reference signal.

Dependencies

To enable this port, select the **Show measurement port** parameter.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Generator type — Number of pulses to generate

Single-phase half-bridge (2 pulses) (default) | Single-phase full-bridge (4 pulses) | Single-phase full-bridge - Bipolar modulation (4 pulses) | Three-phase bridge (6 pulses)

Specify the number of pulses to generate. The number of pulses generated by the block is proportional to the number of bridge arms to fire.

Select **Single-phase half-bridge (2 pulses)** to fire the self-commutated devices of a single-phase half-bridge converter. Pulse 1 fires the upper device, and pulse 2 fires the lower device.

Select **Single-phase full-bridge (4 pulses)** to fire the self-commutated devices of a single-phase full-bridge converter. Four pulses are then generated. Pulses 1 and 3 fire the upper devices of the first and second arm. Pulses 2 and 4 fire the lower devices.

Select **Single-phase full-bridge - Bipolar modulation (4 pulses)** to fire the self-commutated devices of a single-phase full-bridge converter. Four pulses are then generated. Pulses 1 and 3 fire the upper devices of the first and second arm. Pulses 2 and 4 fire the lower devices. Pulses 1 and 4 are identical. Pulses 2 and 3 are identical.

Select **Three-phase bridge (6 pulses)** (default) to fire the self-commutated devices of a three-phase bridge converter. Pulses 1, 3, and 5 fire the upper devices of the first, second, and third arms. Pulses 2, 4, and 6 fire the lower devices.

Carrier: Mode of operation — Mode of operation of carrier signal

Unsynchronized (default) | Synchronized

When set to Unsynchronized (default), the frequency of the unsynchronized carrier signal is determined by the **Frequency** parameter.

When set to **Synchronized**, the carrier signal is synchronized to an external reference signal (input wt) and the carrier frequency is determined by the **Switching ratio** parameter.

Carrier: Frequency (Hz) — Frequency of triangular carrier signal

27*60 (default)

Frequency, in hertz, of the triangular carrier signal.

Dependencies

To enable this parameter, set **Mode of operation** to **Unsynchronized**.

Carrier: Initial phase (degrees) — Carrier initial phase

90 (default)

Carrier initial phase, in degrees. A value of 90 degrees means that the triangle carrier initial position is set to midpoint between its minimum and maximum value and the slope is positive.

Dependencies

To enable this parameter, set **Mode of operation** to **Unsynchronized**.

Carrier: Minimum and maximum values — Minimum and maximum values of triangular carrier signal

[-1 1] (default)

Minimum (valley) and maximum (peak) values of the triangular carrier signal.

Switching ratio (carrier frequency/output frequency) — Frequency of triangular carrier signal

27 (default)

Frequency (F_c) of the triangular carrier signal.

$$F_c = \text{SwitchingRatio} \times \text{OutputVoltageFrequency}$$

Dependencies

To enable this parameter, set **Mode of operation** to **Synchronized**.

Reference signal: Sampling technique — Sampling technique of reference signal

Natural (default) | Asymmetrical regular (double edge) | Symmetrical regular (single edge)

Specify how the reference signal is sampled: **Natural** (default), **Asymmetrical regular** (double edge), or **Symmetrical regular** (single edge).

If you select a regular sampling technique, the **Sample time** parameter must be an integer submultiple of the sampling period. The sampling period is equal to $1/\text{Carrier Frequency}/2$ for asymmetrical sampling and to $1/\text{Carrier Frequency}$ for symmetrical sampling.

Reference signal: Internal generation of reference signal — Internal generation of reference signal

off (default) | on

When selected, the reference signal is generated by the block. Default is cleared.

When not selected, external reference signals are used for pulse generation.

This parameter is available only if the **Mode of operation** parameter is set to Unsynchronized.

Dependencies

To enable this parameter, set **Mode of operation** to Unsynchronized.

Reference signal: Modulation index — Modulation index
0.8 (default)

Modulation index to control the amplitude of the fundamental component of the output voltage of the converter. The modulation index must be greater than 0 and lower than or equal to 1.

Dependencies

To enable this parameter, select the **Internal generation of reference signal** parameter.

Reference signal: Frequency (Hz) — Frequency of reference signal
60 (default)

Output voltage frequency used to control the frequency of the fundamental component of the output voltage of the converter.

Dependencies

To enable this parameter, select the **Internal generation of modulating signal (s)** parameter.

Reference signal: Phase (degrees) — Phase of reference signal
0 (default)

Phase of the fundamental component of the output voltage of the converter.

Dependencies

To enable this parameter, select the **Internal generation of modulating signal (s)** parameter.

Sample time — Sample time
0 (default)

Sample time of the block, in seconds. Set to 0 to implement a continuous block. If you select a regular sampling technique, the **Sample time** parameter must be an integer submultiple of the sampling period. The sampling period is equal to $1/\text{Carrier Frequency}/2$ for asymmetrical sampling and to $1/\text{Carrier Frequency}$ for symmetrical sampling.

Show measurement port — Show measurement port
off (default) | on

Select this check box to add a Simulink output to the block. The output returns the carrier signal that is used to determine the output pulses and the sampled reference signal.

Version History

Introduced in R2013a

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

PWM Generator | PWM Generator (3-Level) | PWM Generator (DC-DC) | PWM Generator (Interpolation) | PWM Generator (Multilevel) | PWM Generator (Pulse Averaging)

PWM Generator (3-Level)

Generate pulses for PWM-controlled three-level converter



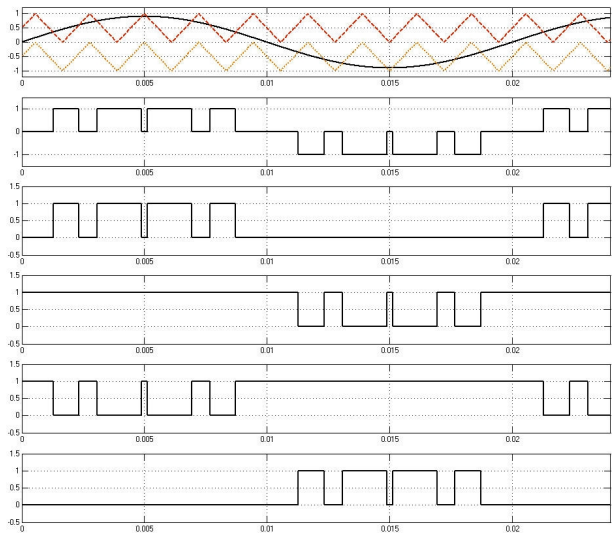
Libraries:
Simscape / Electrical / Specialized Power Systems / Power Electronics / Power Electronics Control

Description

The PWM Generator (3-Level) block generates pulses for carrier-based pulse-width modulation (PWM) converters using three-level topology. The block can control switching devices (FETs, GTOs, or IGBTs) of three different converter types: single-phase half-bridge (one arm), single-phase full-bridge (two arms), or three-phase bridge (three arms).

The reference signal (Uref input), also called the modulating signal, is naturally sampled and compared with two symmetrical level-shifted triangle carriers.

The following figure shows how the pulses are generated for a single-phase, half-bridge three-level converter.



The converter arm can have three states: +1, 0, or -1. When the reference signal is greater than the positive carrier, the state of the arm is +1; when the reference signal is smaller than the negative carrier, the state of the arm is -1.

Otherwise, the state is 0. Based on the current state of the arm, the appropriate pulses are generated.

State	Q1	Q2	Q3	Q4
1	1	1	0	0

State	Q1	Q2	Q3	Q4
0	0	1	1	0
-1	0	0	1	1

One reference signal is required to generate the four pulses of an arm. For a single-phase full-bridge converter, a second reference signal is required to generate the four pulses of the second arm. This signal is internally generated by phase-shifting the original reference signal by 180 degrees. For a three-phase bridge, three reference signals are required to generate the 12 pulses.

The reference signal also can be internally generated by the PWM generator. In this case, specify a modulation index, voltage output frequency, and phase.

Examples

The `power_PWMGenerator3Level` model uses a simple circuit to show how the PWM Generator (3-Level) operates. Run the simulation and use the FFT Analysis tool of the Powergui block to see the harmonics and the THD value of the voltages produced by the three-phase three-level converter.

The model sample time is parameterized by the `Ts` variable set to a default value of 2e-6. Set `Ts` to 0 in the command window and change the Simulation type parameter of the Powergui block to Continuous to simulate the model in continuous mode.

Characteristics

Sample Time	Specified in the Sample Time parameter Continuous if Sample Time = 0
Scalar Expansion	No
Dimensionalized	No

Ports

Input

Uref — Reference signal used to generate output pulses

Vectorized reference signal used to generate the output pulses. Connect this input to a single-phase sinusoidal signal when the block is used to control a single-phase half- or full-bridge converter, or to a three-phase sinusoidal signal when the PWM Generator block is controlling a three-phase bridge converter. For linear operation of this block, the magnitude of **Uref** must be between -1 and +1.

Dependencies

To enable this port when **Mode of operation** is Synchronized, clear the **Internal generation of modulating signal (s)** parameter.

wt — External reference signal used to synchronize carrier

External reference signal used to synchronize the carrier.

Dependencies

To enable this port, set **Mode of operation** to Synchronized.

Output

m — Measurement output

The output contains the 4, 8, or 12 pulses used to fire the self-commutated devices (MOSFETs, GTOs, or IGBTs) of a one-, two- or three-arm three-level converter.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Generator type — Number of pulses to generate

Single-phase half-bridge (4 pulses) (default) | Single-phase full-bridge (8 pulses) | Three-phase bridge (12 pulses)

Specify the number of pulses to generate. The number of pulses generated by the block is proportional to the number of bridge arms to fire.

Select **Single-phase half-bridge (4 pulses)** to fire the self-commutated devices of a single-phase half-bridge converter. Pulses (1, 2) fire the upper devices and pulses (3, 4) fire the lower devices.

Select **Single-phase full-bridge (8 pulses)** to fire the self-commutated devices of a single-phase full-bridge converter. Eight pulses are then generated. Pulses (1, 2) and (5, 6) fire the upper devices of the first and second arms. Pulses (3, 4) and (7, 8) fire the lower devices of the first and second arms.

Select **Three-phase bridge (12 pulses)** (default) to fire the self-commutated devices of a three-phase bridge converter. Pulses (1, 2), (5, 6), and (9, 10) fire the upper devices of the first, second, and third arms. Pulses (3, 4), (7, 8), and (11, 12) fire the lower devices of the three arms.

Mode of operation — Mode of operation of carrier signal

Unsynchronized (default) | Synchronized

When set to **Unsynchronized**, the frequency of the unsynchronized carrier signal is determined by the **Carrier frequency (Hz)** parameter.

When set to **Synchronized**, the carrier signal is synchronized to an external reference signal (input *wt*) and the carrier frequency is determined by the **Switching ratio (carrier frequency/output frequency)** parameter.

Carrier frequency (Hz) — Frequency of triangular carrier signal

27*60 (default)

Frequency, in hertz, of the triangular carrier signal.

Dependencies

To enable this parameter, set **Mode of operation** to **Unsynchronized**.

Switching ratio (carrier frequency/output frequency) — Frequency of triangular carrier signal

27 (default)

Frequency (F_c) of the triangular carrier signal.

$$F_c = \text{SwitchingRatio} \times \text{OutputVoltageFrequency}$$

Dependencies

To enable this parameter, set **Mode of operation** to Synchronized.

Internal generation of reference signal(s) — Internal generation of reference signal
off (default) | on

When selected, the reference signal is generated by the block.

When not selected, external reference signals are used for pulse generation.

This parameter is available only if the **Mode of operation** parameter is set to Unsynchronized.

Dependencies

To enable this parameter, set **Mode of operation** to Unsynchronized.

Modulation index — Modulation index
0.8 (default)

Modulation index to control the amplitude of the fundamental component of the output voltage of the converter. The modulation index must be greater than 0 and lower than or equal to 1.

Dependencies

To enable this parameter, set **Mode of operation** to Unsynchronized and select **Internal generation of reference signal**.

Output voltage frequency (Hz) — Frequency of reference signal
60 (default)

Output voltage frequency used to control the frequency of the fundamental component of the output voltage of the converter.

Dependencies

To enable this parameter, set **Mode of operation** to Unsynchronized and select **Internal generation of reference signal**.

Output voltage phase (degrees) — Phase of reference signal
0 (default)

Phase of the fundamental component of the output voltage of the converter.

Dependencies

To enable this parameter, set **Mode of operation** to Unsynchronized and select **Internal generation of reference signal**.

Sample time — Sample time
0 (default)

Sample time of the block, in seconds. Set to 0 to implement a continuous block.

Version History

Introduced in R2013a

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

PWM Generator | PWM Generator (DC-DC) | PWM Generator (Interpolation) | PWM Generator (Multilevel) | PWM Generator (Pulse Averaging)

PWM Generator (DC-DC)

Generate pulse for PWM-controlled DC-DC converter



Libraries:
Simscape / Electrical / Specialized Power Systems / Power Electronics / Power Electronics Control

Description

The PWM Generator (DC-DC) block outputs a pulse to fire the MOSFET, GTO, or IGBT switch of a one-quadrant converter (buck or boost). The duty cycle of the generator is specified by the input D. A value between 0 and 1 determines the percentage of the pulse period that the output is on.

Characteristics

Sample Time	Specified in the Sample Time parameter Continuous if Sample Time = 0
Scalar Expansion	No
Dimensionalized	No

Examples

The power_PWMGeneratorDCDC example shows the operation of the PWM Generator (DC-DC) block. Run the simulation for various duty cycles values and observe the resulting voltage and current on the Scope block.

The model sample time is parameterized with variable Ts (default value Ts = 500e-9). To run a continuous simulation, at the MATLAB command prompt, enter

Ts = 0

and set the **Simulation type** parameter of the Powergui block to Continuous before starting the simulation.

Ports

Input

D — Generator duty cycle
scalar in the range [0,1]

Duty cycle of the generator. Specify a signal with a value between 0 and 1.

Output

P — Signal to convert into equivalent current source

Pulse that the generator outputs.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Switching frequency (Hz) — Generator switching frequency
5000 (default) | scalar

Switching frequency Fsw of the generator, in hertz. The switching frequency is the frequency of the internal sawtooth signal used to generate the pulse.

Sample time — Block sample time
0 (default) | scalar

Sample time of the block, in seconds. Set to 0 to implement a continuous block.

Version History

Introduced in R2013a

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

PWM Generator | PWM Generator (2-Level) | PWM Generator (3-Level)

PWM Generator (Interpolation)

Carrier-based unipolar PWM generation with interpolation



Libraries:

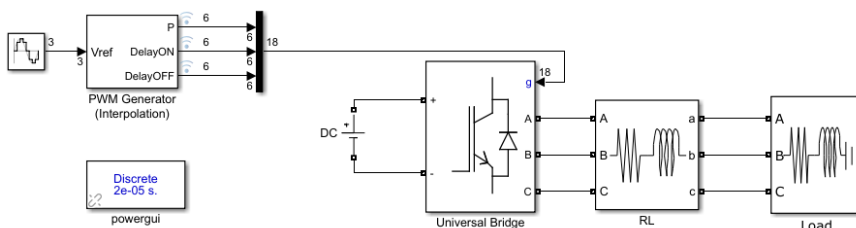
Simscape / Electrical / Specialized Power Systems / Power Electronics / Power Electronics Control

Description

The PWM Generator (Interpolation) block generates pulses for PWM-controlled power electronics converters by using a carrier-based, unipolar PWM method with interpolation.

The block can control switching devices with two-quadrant converters, single-phase full-bridges, two-level three-phase bridges, or three-level three-phase bridges. The reference signal at the **Vref** port, also called the modulating signal, is compared with a symmetrical triangle carrier. When the reference signal is greater than the carrier, the pulse for the upper switching device is high (1), and the pulse for the lower device is low (0). The generator uses this unipolar modulation technique to control each arm independently. A second reference signal is internally generated by phase-shifting the original reference signal by 180 degrees.

The interpolation method takes into account switching events that occur between the two sample times of the generator and computes the time delays to capture the evolution of the states at different switching times. The block computes the pulse delays sent to the power electronic gate inputs when the **Use time stamped gate signals** option of the powergui block is enabled.



This figure shows Pulse 1 of the output port **P**, delay 1 of the output port **DelayOn** and delay 1 of the output port **DelayOff**.



Every time the pulse signal enters the on state, **DelayOn** outputs a value that corresponds to the exact time delay between the computation step and when the pulse entered the on state. In the figure, the Pulse 1 signal enters the on state at 0.09128 seconds (indicated by the blue arrow). At the same moment, the **DelayOn** output for Pulse 1 is 1e-5 seconds. Therefore, the exact time Pulse 1 entered the on state was $0.09128 - 1e-5 = 0.09127$ seconds.

Similarly, every time the pulse signal enters the off state, **DelayOff** outputs a value that corresponds to the exact time delay between the computation step and when the pulse entered the off state. In the figure, Pulse 1 enters the off state at 0.09204 seconds (indicated by the red arrow). At this moment, the **DelayOff** output is equal to 1.09e-5 seconds. Therefore, the exact time Pulse 1 entered the off state was $0.09204 - 1.05e-5 = 0.0920295$ seconds.

This time-stamped interpolation technique works only with the following converter blocks:

- Converters built with individual switch blocks
- Universal Bridge blocks (for modelling half bridge, full bridge, and two-level three-phase converters)
- Three-Level Bridge blocks

For more information on the sampling techniques, see the PWM Generator (2-Level) block reference page.

Ports

Input

Vref — Reference voltage

three-phase vector | single-phase scalar

Reference voltage signal used to generate the output pulses, specified as a three-phase vector or a single-phase scalar. Connect this input to a single-phase sinusoidal signal to control a single-phase full-bridge converter, or to a three-phase sinusoidal signal to control a three-phase bridge converter. For linear operation of this block, the magnitude at **Vref** must be between -1 and $+1$.

Dependencies

This port is visible only if **Generator type** is set to one of these values:

- Full-bridge, single-phase
- Two-level, three-phase
- Three-level, three-phase

.

Data Types: double

D — Duty cycle
single-phase scalar

Duty cycle of the PWM generator. A value between 0 and 1 determines the percentage of the pulse period that the output is on.

Dependencies

This port is visible only if **Generator type** is set to Two-quadrant.

Data Types: double

Output

P — Pulses
vector

Output pulses, returned as a vector.

Data Types: double

DelayOn — On delay
scalar

Output representing the delay between when the pulses entered the on state, returned as a scalar.

Data Types: double

DelayOff — Off delay
scalar

Output representing the delay between when the pulses entered the off state, returned as a scalar.

Data Types: double

Parameters

Generator type — PWM model
Two-level, three-phase (default) | Two-quadrant | Full-bridge, single phase | Three-level, three-phase

Type of generator to control. The number of pulses generated by the block is proportional to the number of bridge arms to fire.

- Select Two-quadrant to fire the self-commutated devices of a two-quadrant converter. The duty cycle of the generator is specified by the input **D**, which determines the percentage of the pulse period that the pulse is on. Two pulses are generated.

- Select **Full-bridge, single-phase** to fire the self-commutated devices of a single-phase full-bridge converter. Four pulses are generated. Pulses 1 and 3 fire the upper devices of the first and second arm. Pulses 2 and 4 fire the lower devices.
- Select **Two-level, three-phase** (default) to fire the self-commutated devices of a three-phase bridge converter. Six pulses are generated. Pulses 1, 3, and 5 fire the upper devices of the first, second, and third arms. Pulses 2, 4, and 6 fire the lower devices.
- Select **Three-level, three-phase** to fire the self-commutated devices of a three-level three phase bridge. Twelves pulses are generated. Pulses (1, 2), (5, 6), and (9, 10) fire the upper devices of the first, second, and third arms. Pulses (3, 4), (7, 8), and (11, 12) fire the lower devices of the three arms.

Carrier frequency (Hz) — Carrier signal frequency
2000 (default)

Frequency, in Hz, of the triangular carrier signal. The value must be greater than 0

Carrier initial phase (deg) — Carrier signal initial phase
0 (default)

Carrier signal initial phase, in deg. A value of 90 degrees means that the triangle carrier initial position is set to the midpoint between its minimum and maximum value and the slope is positive.

Sample time (s) — Block sample time
50e-6 (default)

Sample time of the block, in s. The value must be positive and greater than 0.

Version History

Introduced in R2019b

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using Simulink® Coder™.

See Also

Pulse Generator (Thyristor) | PWM Generator (2-Level) | PWM Generator (3-Level) | PWM Generator (DC-DC) | PWM Generator (Multilevel) | PWM Generator (Pulse Averaging)

PWM Generator (Multilevel)

Generate pulses for PWM-controlled modular multilevel converter



Library

Simscape / Electrical / Specialized Power Systems / Power Electronics / Power Electronics Control

Description

The PWM Generator (Multilevel) block generates pulses for a PWM-controlled modular multilevel converter. For a half-bridge converter, carriers are phase-shifted by an angle obtained by dividing 360 degrees by the number of bridges. The number of bridges is a parameter that you specify. For a full-bridge converter, carriers are phase-shifted by an angle obtained by dividing 180 degrees by the number of bridges.

The PWM Generator (Multilevel) block generates one carrier signal for every power module (full-bridge or half-bridge) that is controlled. For example, if you set **Bridge type** to **Full-bridge**, and **Number of bridges** to 10, the block generates 10 carrier signals, each shifted by 18 degrees (180/10) from the other.

Parameters

Bridge Type

Choose whether to control a **Half-bridge** (default) or a **Full-bridge** converter.

Number of bridges

Specify the number of bridges (power modules) of the multilevel converter you want to control. The number of bridges must correspond to the number of power modules used in the modular multilevel converter. The default value is 4.

For example, if you want to control a full-bridge MMC converter that contains 22 power modules (the equivalent of 22 full-bridges), set the **Bridge type** parameter to **Full-bridge** and the **Number of bridges** parameter to 22. If you want to control a half-bridge MMC converter that contains 22 power modules (or 22 half-bridges), set **Bridge type** to **Half-bridge** and **Number of bridges** to 22.

Carrier frequency (Hz)

Specify the frequency, in hertz, of the triangular carrier signals. The default value is 540.

Sample time (s)

Specify the sample time of the block. Set to 0 to implement a continuous block. Default is 0.

Show carriers output

Select this option to show the Cr output port for the carriers' waveforms. By default, this option is cleared.

Inputs and Outputs

Uref

Vectorized single-phase sinusoidal reference signal used to generate the output pulses. You specify one reference voltage for each bridge of the multilevel converter you want to control. For linear operation of this block, the magnitude of **Uref** must be between -1 and $+1$.

P

Vectorized gating signal to control a multilevel converter. When the **Bridge type** parameter is set to **Half-bridge**, the block outputs double the **Number of bridges** pulses. When the **Bridge type** parameter is set to **Full-bridge**, the block outputs four times the **Number of bridges** pulses.

Cr

Carriers' waveforms. This output only appears if you enable the **Show carriers output** option.

Characteristics

Sample Time	Specified in the Sample Time parameter Continuous if Sample Time = 0
Scalar Expansion	No
Dimensionalized	No

Version History

Introduced in R2016b

See Also

Full-Bridge MMC | Half-Bridge MMC | Pulse Generator (Thyristor) | PWM Generator (2-Level) | PWM Generator (3-Level) | PWM Generator (DC-DC) | SVPWM Generator (2-Level)

Topics

"Switching Function Converter Controlled by Averaged Firing Pulses"

PWM Generator (Pulse Averaging)

Carrier-based PWM generator with pulse averaging



Libraries:

Simscape / Electrical / Specialized Power Systems / Power Electronics / Power Electronics Control

Description

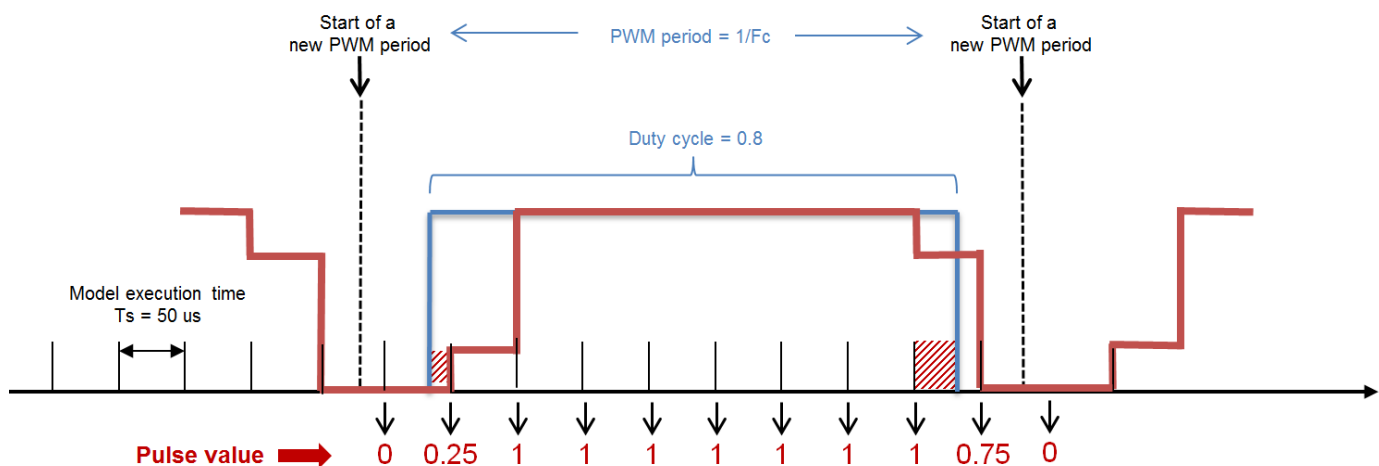
The PWM Generator (Pulse Averaging) block generates pulses, averaged over the sample time model, to use with these converter blocks when they have a switching function model selected:

- Two-quadrant DC/DC Converter block
- Full-Bridge Converter block
- Two-Level Converter block
- Three-Level NPC Converter block

To use an SPS interpolation technique or SPS switching function models, the SPS solver needs to know when a transition from high-to-low or low-to-high occurs within a sample time. The PWM Generator (Pulse Averaging) block can provide the timing for these transitions.

With this pulse-averaging generator (and the SPS switching function converter models), a much higher sample time can be used for the SPS model while maintaining a high-fidelity simulation. Consequently, this generator serves as a virtual FPGA.

This figure illustrates the operation of the generator for a specified carrier frequency (F_c) of 2 KHz and a duty cycle (D) of 0.8. The model execution time (T_s) is 50 microseconds.



The generator outputs a pulse value using a naturally sampled, carrier-based modulation technique. If a transition occurs within a sample time, the generator will output a pulse value representing the time ratio (a value between 0 and 1) for the on state over the sample time. For example, at the

sample time identified by the first blue arrow, the generator outputs a value of 0.25, which represents that the pulse entered on at $0.25 \cdot T_s$ seconds (it was on during 25% of the model execution time). The offset is represented by the red dashed area. Similarly, at the sample time identified by the second blue arrow, the generator outputs a value of 0.75, which represents that the pulse was on during 75% of the model execution time (it entered the off state $0.75 \cdot T_s$ seconds after the previous sample time).

Ports

Input

Vref — Reference voltage

three-phase vector | single-phase scalar

Reference voltage signal used to generate the output pulses, specified as a three-phase vector or a single-phase scalar. Connect this input to a single-phase sinusoidal signal to control a single-phase full-bridge converter, or to a three-phase sinusoidal signal to control a three-phase bridge converter. For linear operation of this block, the magnitude at **Vref** must be between -1 and $+1$.

Dependencies

This port is visible only if **Generator type** is set to one of these values:

- Full-bridge, single-phase
- Two-level, three-phase
- Three-level, three-phase

.

Data Types: double

D — Duty cycle

scalar

Duty cycle of the PWM generator. The value must be between 0 and 1.

Dependencies

This port is visible only if **Generator type** is set to Two-quadrant.

Output

P — Pulses

vector

Output pulses, returned as a vector.

Data Types: double

Parameters

Generator type — PWM model

Two-level, three-phase (default) | Two-quadrant | Full-bridge, single phase | Three-level, three-phase

Type of generator to control. The number of pulses generated by the block is proportional to the number of bridge arms to fire.

- Select **Two-quadrant** to fire the self-commutated devices of a two-quadrant converter. The duty cycle of the generator is specified by the input **D**, which determines the percentage of the pulse period that the pulse is on
- Select **Full-bridge, single-phase** to fire the self-commutated devices of a single-phase full-bridge converter. Four pulses are then generated. Pulses 1 and 3 fire the upper devices of the first and second arm. Pulses 2 and 4 fire the lower devices
- Select **Two-level, three-phase** (default) to fire the self-commutated devices of a three-phase bridge converter. Pulses 1, 3, and 5 fire the upper devices of the first, second, and third arms. Pulses 2, 4, and 6 fire the lower devices.
- Select **Three-level, three-phase** to fire the self-commutated devices of a three-level three phase bridge. Pulses (1, 2), (5, 6), and (9, 10) fire the upper devices of the first, second, and third arms. Pulses (3, 4), (7, 8), and (11, 12) fire the lower devices of the three arms.

Carrier frequency (Hz) — Carrier signal frequency

2000 (default)

Frequency, in Hz, of the triangular carrier signal. The value must be greater than 0.

Carrier initial phase (deg) — Carrier signal initial phase

0 (default)

Carrier signal initial phase, in deg. A value of 90 degrees means that the triangle carrier initial position is set to midpoint between its minimum and maximum value and the slope is positive.

Sample time (s) — Block sample time

50e-6 (default)

Sample time of the block, in s. The value must be positive and greater than 0.

Version History

Introduced in R2019b

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

Pulse Generator (Thyristor) | PWM Generator (2-Level) | PWM Generator (3-Level) | PWM Generator (DC-DC) | PWM Generator (Interpolation) | PWM Generator (Multilevel)

RMS

Compute true root mean square (RMS) value of signal



Libraries:
Simscape / Electrical / Specialized Power Systems / Sensors and Measurements

Description

The RMS block computes the true root mean square (RMS) value of the input signal. The true RMS value of the input signal is calculated over a running average window of one cycle of the specified fundamental frequency:

$$RMS(f(t)) = \sqrt{\frac{1}{T} \int_{t-T}^t f(t)^2 dt},$$

where $f(t)$ is the input signal and T is 1/(fundamental frequency).

As this block uses a running average window, one cycle of simulation must complete before the output gives the correct value. For the first cycle of simulation, the output is held to this specified initial RMS value.

Characteristics

Sample Time	Specified in the Sample Time parameter. Continuous if Sample Time = 0.
Scalar Expansion	Yes, of the parameters.
Dimensionalized	Yes.

Examples

Calculate True RMS Value for Signal with Harmonics and at Fundamental Frequency

The power_RMS_THD example shows two applications of the RMS Measurement block. One RMS block calculates the true RMS value of a signal with harmonics, and another RMS calculates the RMS value of the same signal at the fundamental frequency.

The model sample time is parameterized by the Ts variable set to a default value of 50e-6 s. Set Ts to 0 in the command window to simulate the model in continuous mode.

Ports

Input

Port_1 — Input signal
scalar | vector

The input signal for which the block computes the RMS value, specified as a scalar or vector.

Output

Port_1 — RMS of input signal

scalar | vector

The RMS value of the input signal, returned as a scalar or vector.

Parameters

True RMS value — Compute true or fundamental RMS value

on (default) | off

Select this check box to have the block compute the true RMS value of the input signal.

Clear this check box to have the block compute the fundamental value of the input signal divided by $\sqrt{2}$.

Fundamental frequency (Hz) — Fundamental frequency

60 (default) | positive scalar

Specify the fundamental frequency, in hertz, of the input signal.

Initial RMS value — Initial RMS value

120 (default) | nonnegative scalar

Specify the initial RMS value of the output signal.

Sample time — Sample time of block

0 (default) | nonnegative scalar

Specify the sample time of the block, in seconds. Set to 0 to implement a continuous block.

Version History

Introduced in R2013a

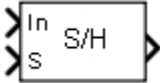
Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

Sample and Hold

Sample first input and hold its value based on value of second input



Library

Simscape / Electrical / Specialized Power Systems / Control

Description

The Sample and Hold block outputs the **In** input signal as long as **S** input signal is true. The block holds the output when the **S** input signal becomes false.

Parameters

Initial condition

Specify the initial value of the output. Default is 0.

Sample time

Specify the sample time of the block, in seconds. Set to 0 to implement a continuous block. Default is 0.

Inputs and Outputs

In

The signal.

S

The control signal, which must be a Boolean.

Out

The sampled signal.

Characteristics

Direct Feedthrough	Yes
Sample Time	Specified in the Sample Time parameter
Scalar Expansion	Yes
Dimensionalized	Yes
Zero-Crossing Detection	No

Examples

The `power_SampleAndHold` example shows the operation of the Sample and Hold block.

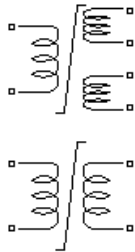
The model sample time is parameterized with the variable `Ts` (default value `Ts = 50e-6`). To simulate a continuous Bistable block, specify `Ts = 0` in the MATLAB Command Window.

Version History

Introduced in R2013a

Saturable Transformer

Implement two- or three-winding saturable transformer

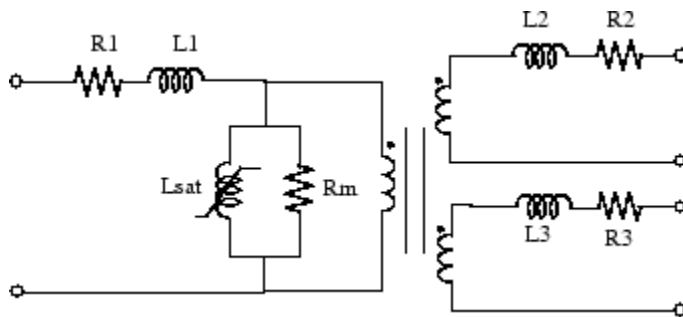


Libraries:

Simscape / Electrical / Specialized Power Systems / Power Grid Elements

Description

The Saturable Transformer block model shown consists of three coupled windings wound on the same core.



The model takes into account the winding resistances ($R1$ $R2$ $R3$) and the leakage inductances ($L1$ $L2$ $L3$) as well as the magnetizing characteristics of the core, which is modeled by a resistance Rm simulating the core active losses and a saturable inductance $Lsat$.

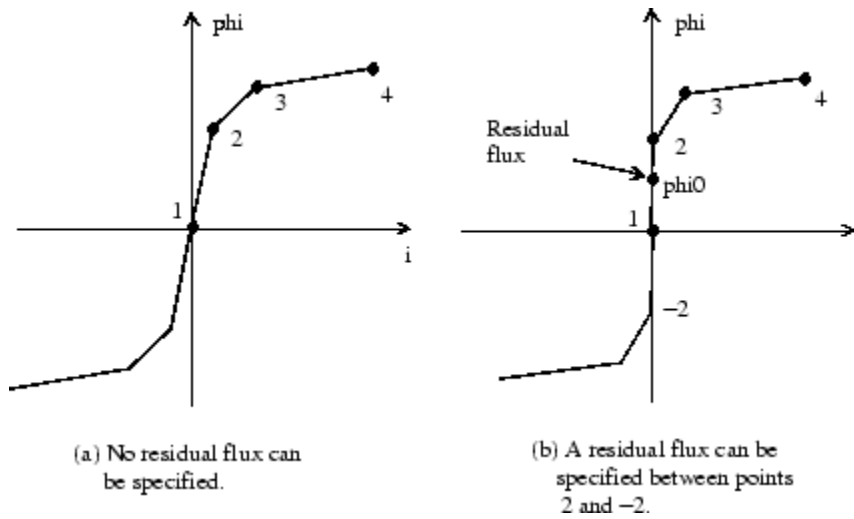
You can choose one of the following two options for the modeling of the nonlinear flux-current characteristic

- 1 Model saturation without hysteresis. The total iron losses (eddy current + hysteresis) are modeled by a linear resistance, Rm .
- 2 Model hysteresis and saturation. Specification of the hysteresis is done by means of the **Hysteresis Design Tool** of the Powergui block. The eddy current losses in the core are modeled by a linear resistance, Rm .

Note Modeling the hysteresis requires additional computation load and therefore slows down the simulation. The hysteresis model should be reserved for specific applications where this phenomenon is important.

Saturation Characteristic Without Hysteresis

When the hysteresis is not modeled, the saturation characteristic of the Saturable Transformer block is defined by a piecewise linear relationship between the flux and the magnetization current.



Therefore, if you want to specify a residual flux, ϕ_{i0} , the second point of the saturation characteristic should correspond to a null current, as shown in the figure (b).

The saturation characteristic is entered as (i, phi) pair values in per units, starting with pair (0, 0). The software converts the vector of fluxes Φ_{pu} and the vector of currents I_{pu} into standard units to be used in the saturation model of the Saturable Transformer block:

$$\Phi = \Phi_{pu} \Phi_{base} I = I_{pu} I_{base}, \quad (1-10)$$

where the base flux linkage (Φ_{base}) and base current (I_{base}) are the peak values obtained at nominal voltage power and frequency:

$$I_{base} = \frac{P_n}{V_1} \sqrt{2}$$

$$\Phi_{base} = \frac{V_1}{2\pi f_n} \sqrt{2}.$$

The base flux is defined as the peak value of the sinusoidal flux (in webers) when winding 1 is connected to a 1 pu sinusoidal voltage source (nominal voltage). The Φ_{base} value defined above represents the base flux linkage (in volt-seconds). It is related to the base flux by the following equation:

$$\Phi_{base} = \text{Base flux} \times \text{number of turns of winding 1}. \quad (1-11)$$

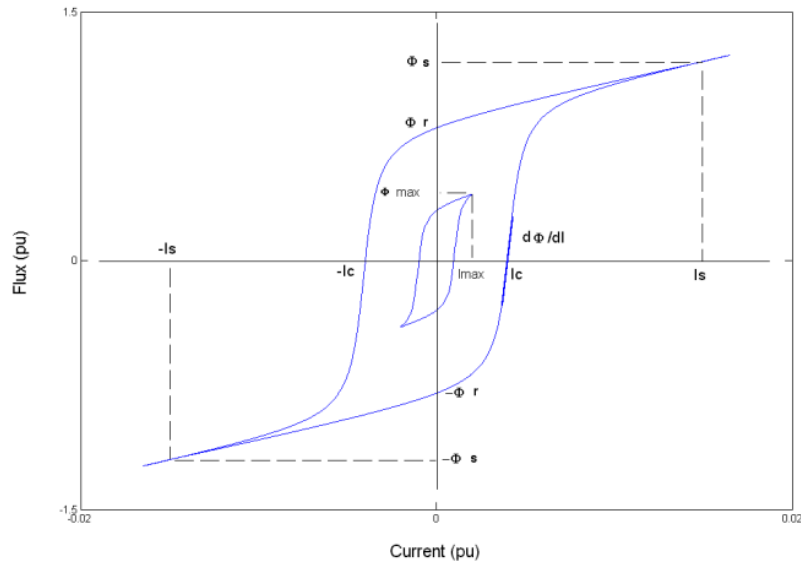
When they are expressed in pu, the flux and the flux linkage have the same value.

Saturation Characteristic with Hysteresis

The magnetizing current I is computed from the flux Φ obtained by integrating voltage across the magnetizing branch. The static model of hysteresis defines the relation between flux and the magnetization current evaluated in DC, when the eddy current losses are not present.

The hysteresis model is based on a semi-empirical characteristic, using an arctangent analytical expression $\Phi(I)$ and its inverse $I(\Phi)$ to represent the operating point trajectories. The analytical expression parameters are obtained by curve fitting empirical data defining the major loop and the single-valued saturation characteristic. The **Hysteresis design tool** of the Powergui block is used to

fit the hysteresis major loop of a particular core type to basic parameters. These parameters are defined by the remanent flux (Φ_r), the coercive current (I_c), and the slope ($d\Phi/dI$) at $(0, I_c)$ point as shown in the next figure.

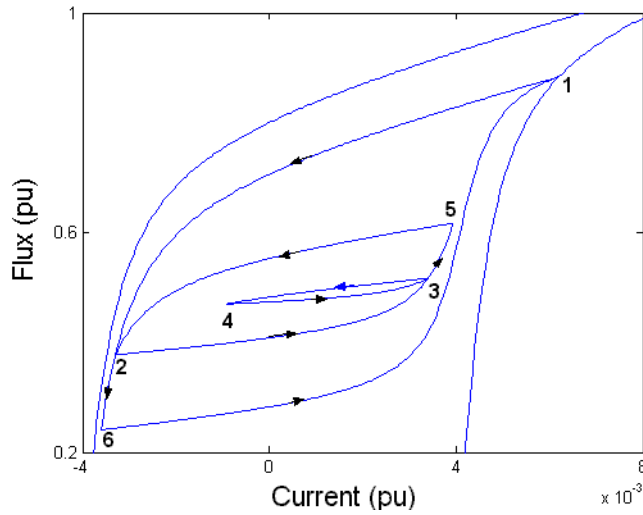


The major loop half cycle is defined by a series of N equidistant points connected by line segments. The value of N is defined in the **Hysteresis design tool** of the Powergui block. Using $N = 256$ yields a smooth curve and usually gives satisfactory results.

The single-valued saturation characteristic is defined by a set of current-flux pairs defining a saturation curve which should be asymptotic to the air core inductance L_s .

The main characteristics of the hysteresis model are summarized below:

- 1 A symmetrical variation of the flux produces a symmetrical current variation between $-I_{max}$ and $+I_{max}$, resulting in a symmetrical hysteresis loop whose shape and area depend on the value of Φ_{max} . The major loop is produced when Φ_{max} is equal to the saturation flux (Φ_s). Beyond that point the characteristic reduces to a single-valued saturation characteristic.
- 2 In transient conditions, an oscillating magnetizing current produces minor asymmetrical loops, as shown in the next figure, and all points of operation are assumed to be within the major loop. Loops once closed have no more influence on the subsequent evolution.



The trajectory starts from the initial (or residual) flux point, which must lie on the vertical axis inside the major loop. You can specify this initial flux value ϕ_0 , or it is automatically adjusted so that the simulation starts in steady state.

The Per Unit Conversion

In order to comply with industry practice, the block allows you to specify the resistance and inductance of the windings in per unit (pu). The values are based on the transformer rated power P_n in VA, nominal frequency f_n in Hz, and nominal voltage V_n , in Vrms, of the corresponding winding. For each winding the per unit resistance and inductance are defined as

$$R(\text{p.u.}) = \frac{R(\Omega)}{R_{\text{base}}}$$

$$L(\text{p.u.}) = \frac{L(H)}{L_{\text{base}}}.$$

The base resistance and base inductance used for each winding are

$$R_{\text{base}} = \frac{V_n^2}{P_n}$$

$$L_{\text{base}} = \frac{R_{\text{base}}}{2\pi f_n}.$$

For the magnetization resistance R_m , the pu values are based on the transformer rated power and on the nominal voltage of winding 1.

The default parameters of winding 1 specified in the dialog box section give the following base values:

$$R_{\text{base}} = \frac{(735 \cdot 10^3 / \sqrt{3})^2}{250 \cdot 10^6} = 720.3 \Omega$$

$$L_{\text{base}} = \frac{720.3}{2\pi \cdot 60} = 1.91 H.$$

For example, if winding 1 parameters are $R_1 = 1.44 \, \Omega$ and $L_1 = 0.1528 \, H$, the corresponding values to enter in the dialog box are

$$R_1 = \frac{1.44\Omega}{720.3\Omega} = 0.002 \, \text{p.u.}$$

$$L_1 = \frac{0.1528H}{1.91H} = 0.08 \, \text{p.u.}$$

Examples

The power_xfosaturable example illustrates the energization of one phase of a three-phase 450 MVA, 500/230 kV transformer on a 3000 MVA source. The transformer parameters are

Nominal power and frequency	Pn = 150e6 VA	fn = 60 Hz	
Winding 1 parameters (primary)	V1 = 500e3 Vrms/ sqrt(3)	R1 = 0.002 pu	L1 = 0.08 pu
Winding 2 parameters (secondary)	V2 = 230e3 Vrms/ sqrt(3)	R2 = 0.002 pu	L2 = 0.08 pu
Saturation characteristic	[0 0; 0.0 1.2; 1.0 1.52]		
Core loss resistance and initial flux	Rm = 500 pu	phi0 = 0.8 pu	

Simulation of this circuit illustrates the saturation effect on the transformer current and voltage.

As the source is resonant at the fourth harmonic, you can observe a high fourth- harmonic content in the secondary voltage. In this circuit, the flux is calculated in two ways:

- By integrating the secondary voltage
- By using the Multimeter block

Assumptions and Limitations

Windings can be left floating (that is, not connected by an impedance to the rest of the circuit). However, the floating winding is connected internally to the main circuit through a resistor. This invisible connection does not affect voltage and current measurements.

Ports

Conserving

+1 — Primary winding positive polarity
specialized electrical

Specialized electrical conserving port associated with the primary winding positive polarity.

-1 — Primary winding negative polarity
specialized electrical

Specialized electrical conserving port associated with the primary winding negative polarity.

+2 — Secondary winding positive polarity
specialized electrical

Specialized electrical conserving port associated with the secondary winding positive polarity.

-2 — Secondary winding negative polarity
specialized electrical

Specialized electrical conserving port associated with the secondary winding negative polarity.

+3 — Tertiary winding positive polarity
specialized electrical

Specialized electrical conserving port associated with the tertiary winding positive polarity.

Dependencies

To enable this port, select the **Three windings transformer** parameter.

-3 — Tertiary winding negative polarity
specialized electrical

Specialized electrical conserving port associated with the tertiary winding negative polarity.

Dependencies

To enable this port, select the **Three windings transformer** parameter.

Parameters

Configuration Tab

Three windings transformer — Number of windings to model
on (default) | off

If selected, specify a saturable transformer with three windings; otherwise it implements a two windings transformer.

Simulate hysteresis — Whether to model hysteresis
off (default) | on

Select to model hysteresis saturation characteristic instead of a single-valued saturation curve.

Hysteresis Mat file — Specify file for hysteresis
'hysteresis.mat' (default) | string

Specify a .mat file containing the data to be used for the hysteresis model. When you open the **Hysteresis Design Tool** of the Powergui, the default hysteresis loop and parameters saved in the hysteresis.mat file are displayed. Use the **Load** button of the Hysteresis Design tool to load another .mat file. Use the **Save** button of the Hysteresis Design tool to save your model in a new .mat file.

Dependencies

To enable this parameter, select **Simulate hysteresis**.

Measurements — Specify what to measure

None (default) | Winding voltages | Winding currents | Flux and excitation current (Im + IRm) | Flux and magnetization current (Im) | All measurement (V, I, Flux)

Select **Winding voltages** to measure the voltage across the winding terminals of the Saturable Transformer block.

Select **Winding currents** to measure the current flowing through the windings of the Saturable Transformer block.

Select **Flux and excitation current (Im + IRm)** to measure the flux linkage, in volt seconds (V.s), and the total excitation current including iron losses modeled by Rm.

Select **Flux and magnetization current (Im)** to measure the flux linkage, in volt seconds (V.s), and the magnetization current, in amperes (A), not including iron losses modeled by Rm.

Select **All measurement (V, I, Flux)** to measure the winding voltages, currents, magnetization currents, and the flux linkage.

Place a Multimeter block in your model to display the selected measurements during the simulation.

In the **Available Measurements** list box of the Multimeter block, the measurements are identified by a label followed by the block name.

Measurement	Label
Winding voltages	Uw1:
Winding currents	Iw1:
Excitation current	Iexc:
Magnetization current	Imag:
Flux linkage	Flux:

Dependencies**Parameters Tab****Units** — Units

pu (default) | SI

Specify the units used to enter the parameters of the block.

Set to **pu** to use per unit.

Set to **SI** to use SI units.

Changing the **Units** parameter from **pu** to **SI** or from **SI** to **pu** automatically converts the parameters displayed in the mask of the block. The per unit conversion is based on the transformer rated power Pn in VA, nominal frequency fn in Hz, and nominal voltage Vn in Vrms, of the windings.

Nominal power and frequency [Pn(VA) fn(Hz)] — Nominal power and frequency

[250e6 60] (default)

The nominal power rating Pn in VA and frequency fn in Hz, of the transformer.

This parameter does not impact the transformer model when you set the **Units** parameter to **SI**.

Winding 1 parameters [V1(Vrms) R1(pu) L1(pu)] — Winding 1 parameters

[7.35e+05 0.002 0.08] (default)

The nominal voltage V1 in volts RMS, resistance R1 in pu, and leakage inductance L1 in pu, of the primary winding.

The pu values are based on the nominal power Pn and on V1.

To implement an ideal winding, set R1 and L1 to 0.

Dependencies

To enable this parameter, set the **Units** parameter to pu.

Winding 1 parameters [V1(Vrms) R1(ohm) L1(H)] — Winding 1 parameters

[7.35e+05 4.3218 0.45856] (default)

The nominal voltage V1 in volts RMS, resistance R1 in ohms, and leakage inductance L1 in H, of the primary winding.

To implement an ideal winding, set R1 and L1 to 0.

Dependencies

To enable this parameter, set the **Units** parameter to SI.

Winding 2 parameters [V2(Vrms) R2(pu) L2(pu)] — Winding 2 parameters

[3.15e+05 0.002 0.08] (default)

The nominal voltage, V2 in volts RMS, resistance R2 in pu, and leakage inductance L2 in pu, of the secondary winding.

The pu values are based on the nominal power Pn and on V2.

To implement an ideal winding, set R2 and L2 to 0.

Dependencies

To enable this parameter, set the **Units** parameter to pu.

Winding 2 parameters [V2(Vrms) R2(ohm) L2(H)] — Winding 2 parameters

[3.15e+05 0.7938 0.084225] (default)

The nominal voltage V2 in volts RMS, resistance R2 in ohms, and leakage inductance L2 in H, of the secondary winding.

To implement an ideal winding, set R2 and L2 to 0.

Dependencies

To enable this parameter, set the **Units** parameter to SI.

Winding 3 parameters [V3(Vrms) R3(pu) L3(pu)] — Winding 3 parameters

[3.15e+05 0.002 0.08] (default)

The nominal voltage V3 in volts RMS, resistance R3 in pu, and leakage inductance L3 in pu, of the tertiary winding.

The pu values are based on the nominal power P_n and on V_3 .

To implement an ideal winding, set R_3 and L_3 to 0.

Dependencies

To enable this parameter, set the **Units** parameter to pu and select the **Three windings transformer** parameter.

Winding 3 parameters [V3(Vrms) R3(ohm) L3(H)] — Winding 3 parameters
[3.15e+05 0.7938 0.084225] (default)

The nominal voltage V_3 in volts RMS, resistance R_3 in ohms, and leakage inductance L_3 in H, of the tertiary winding.

To implement an ideal winding, set R_3 and L_3 to 0.

Dependencies

To enable this parameter, set the **Units** parameter to SI and select the **Three windings transformer** parameter.

Saturation characteristic [i1 phi1; i2 phi2; ...] (pu) — Series of magnetizing current pairs
[0,0 ; 0.0024,1.2 ; 1.0,1.52] (default) | vector

Specify a series of magnetizing current (pu) - flux (pu) pairs starting with (0,0).

Dependencies

To enable this parameter, set the **Units** parameter to pu.

Saturation characteristic [i1(A) phi1(V.s); i2 phi2; ...] — Series of magnetizing current pairs
[0 0;1.1545 3308.7;481.03 4191] (default) | vector

Specify a series of magnetizing current (pu) - flux (pu) pairs starting with (0,0).

Dependencies

To enable this parameter, set the **Units** parameter to SI.

Core loss resistance and initial flux [Rm phi0] or [Rm] (pu) — Active power dissipated in core
[500] (default) | positive scalar

Specify the active power dissipated in the core by entering the equivalent resistance R_m in pu. For example, to specify a 0.2% of active power core loss at nominal voltage, use $R_m = 500$ pu. You can also specify the initial flux ϕ_0 (pu). This initial flux becomes particularly important when the transformer is energized. If ϕ_0 is not specified, the initial flux is automatically adjusted so that the simulation starts in steady state. When simulating hysteresis, R_m models the eddy current losses only.

Dependencies

To enable this parameter, set the **Units** parameter to pu.

Core loss resistance and initial flux [Rm(ohm) phi0(V.s)] or [Rm(ohm)] — Active power dissipated in core
1.0805e+06 (default) | positive scalar

Specify the active power dissipated in the core by entering the equivalent resistance R_m in pu. For example, to specify a 0.2% of active power core loss at nominal voltage, use $R_m = 500$ pu. You can also specify the initial flux ϕ_0 (pu). This initial flux becomes particularly important when the transformer is energized. If ϕ_0 is not specified, the initial flux is automatically adjusted so that the simulation starts in steady state. When simulating hysteresis, R_m models the eddy current losses only.

Dependencies

To enable this parameter, set the **Units** parameter to SI.

Advanced Tab

Break Algebraic loop in discrete saturation model — Option to break algebraic loop

`off` (default) | `on`

When selected, a delay is inserted at the output of the saturation model computing magnetization current as a function of flux linkage (the integral of input voltage computed by a trapezoidal method). This delay eliminates the algebraic loop resulting from trapezoidal discretization methods and speeds up the simulation of the model. However, this delay introduces a one simulation step time delay in the model and can cause numerical oscillations if the sample time is too large. The algebraic loop is required in most cases to get an accurate solution.

When cleared, the discretization method of the saturation model is specified by the **Discrete solver model** parameter.

Dependencies

To enable this parameter, add a powergui block to your model, set the **Simulation type** parameter of the powergui block to **Discrete**, and clear the **Automatically handle discrete solver** parameter of the powergui block.

Discrete solver model — Method to resolve algebraic loop

`Trapezoidal iterative` (default) | `Trapezoidal robust` | `Backward Euler robust`

Select one of these methods to resolve the algebraic loop:

- **Trapezoidal iterative**—Although this method produces correct results, it is not recommended because Simulink tends to slow down and may fail to converge (simulation stops), especially when the number of saturable transformers is increased. Also, because of the Simulink algebraic loop constraint, this method cannot be used in real time. In R2018b and previous releases, you used this method when the **Break Algebraic loop in discrete saturation model** parameter was cleared.
- **Trapezoidal robust**—This method is slightly more accurate than the **Backward Euler robust** method. However, it may produce slightly damped numerical oscillations on transformer voltages when the transformer is at no load.
- **Backward Euler robust**—This method provides good accuracy and prevents oscillations when the transformer is at no load.

The maximum number of iterations for the robust methods is specified in the **Preferences** tab of the powergui block, in the **Solver details for nonlinear elements** section. For real time applications, you may need to limit the number of iterations. Usually, limiting the number of iterations to 2 produces acceptable results. The two robust solvers are the recommended methods for discretizing the saturation model of the transformer.

For more information on what method to use in your application, see “Simulating Discretized Electrical Systems”.

Dependencies

To enable this parameter, add a powergui block to your model, set the **Simulation type** parameter of the powergui block to **Discrete**, and clear the **Automatically handle discrete solver** parameter of the powergui block. Also, in the Saturable Transformer block, clear the **Break Algebraic loop in discrete saturation model** parameter.

Version History

Introduced before R2006a

References

- [1] Casoria, S., P. Brunelle, and G. Sybille, “Hysteresis Modeling in the MATLAB/Power System Blockset,” Electrimacs 2002, École de technologie supérieure, Montreal, 2002.
- [2] Frame, J.G., N. Mohan, and Tsu-huei Liu, “Hysteresis modeling in an Electro-Magnetic Transients Program,” presented at the IEEE PES winter meeting, New York, January 31 to February 5, 1982.

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

Linear Transformer | Multimeter | Mutual Inductance | powergui | Three-Phase Transformer (Two Windings) | Three-Phase Transformer (Three Windings)

Sawtooth Generator

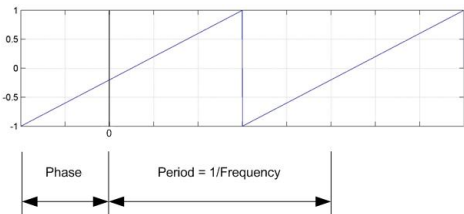
Generate sawtooth waveform at regular intervals



Libraries:
Simscape / Electrical / Specialized Power Systems / Sources / Signal Generator
Sources

Description

The Sawtooth Generator block generates a sawtooth waveform, with peak amplitude of ± 1 , at regular intervals. The figure shows how the **Frequency** and **Phase** block parameters affect the output waveform.



Examples

The power_SignalGenerators example shows various uses of the Sawtooth Generator block.

Characteristics

Sample Time	Specified in the Sample Time parameter Continuous if Sample Time = 0
Scalar Expansion	No
Dimensionalized	No
Zero-Crossing Detection	No

Ports

Output

Out1 — Output signal
scalar

Output of the block sawtooth signal.

Parameters

Frequency (Hz) — Sawtooth waveform frequency
1e3 (default) | scalar

Specify the frequency of the sawtooth waveform, in hertz.

Phase (degrees) — Sawtooth waveform

180 (default) | scalar

Specify the delay of the sawtooth waveform, in degrees. When the phase is set to 0, the waveform starts with an amplitude of -1 and a positive slope.

Sample time — Block sample time

0 (default) | nonnegative scalar

Specify the sample time of the block, in seconds. Set to 0 to implement a continuous block.

Version History

Introduced in R2013a

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

Second-Order Filter

Implement second-order filter



Libraries:
Simscape / Electrical / Specialized Power Systems / Control / Filters

Description

Based on the **Filter type** selected in the block menu, the Second-Order Filter block implements the following transfer function:

Low-pass filter:

$$H(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

High-pass filter:

$$H(s) = \frac{s^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

Band-pass filter:

$$H(s) = \frac{2\zeta\omega_n s}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

Band-stop (notch) filter:

$$H(s) = \frac{s^2 + \omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

s = Laplace operator

ω_n = natural frequency; $\omega_n = 2\pi f_n$

ζ = damping ratio (called Zeta in the block menu)

The key characteristics of the Second-Order Filter block are:

- Input accepts a vectorized input of N signals, implementing N filters. This feature is particularly useful for designing controllers in three-phase systems (N = 3).
- Filter states can be initialized for specified DC and AC inputs.
- It enables you to compute and plot filter response.

Characteristics

Direct Feedthrough	Yes

Sample Time	Specified in the Sample Time parameter Continuous if Sample Time = 0
Scalar Expansion	Yes, of the parameters
States	Two states per filter
Dimensionalized	Yes

Examples

The `power_SecondOrderFilter` example shows the Second-Order Filter block using two **Filter type** parameter settings (Lowpass and Bandstop).

The model sample time is parameterized with variable T_s (default value $T_s = 50\text{e-}6$). To simulate continuous filters, specify $T_s = 0$ in the MATLAB Command Window before starting the simulation.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Filter type — Type of filter

Bandpass (default) | Lowpass | Highpass | Bandstop (notch)

Specify the type of filter: Lowpass, Highpass, Bandpass (default), or Bandstop (notch).

Natural frequency fn (Hz) — Natural frequency of filter

120 (default) | scalar | vector

Natural frequency of the filter, in hertz.

Damping ratio Zeta (Q = 1/(2*Zeta)) — Damping ratio of filter

0.707 (default) | scalar in the range [0, 1]

Damping ratio of the filter. The damping ratio is typically a value between 0 and 1.

The damping ratio is related to the filter quality factor Q :

$$Q = \frac{1}{2\zeta}$$

For a bandpass or a bandstop filter, the 3 dB bandwidth is given by

$$BW = \frac{f_n}{Q} = 2\zeta f_n$$

Sample time — Block sample time

0 (default) | nonnegative scalar

Sample time of the block, in seconds. Set to 0 to implement a continuous block.

Initialize filter states — Option to initialize filter states

on (default) | off

Select this parameter to initialize the filter states according to the **AC initial input** and **DC initial input** parameters.

AC initial input: [Mag, Phase (degrees), Freq (Hz)] — Magnitude, phase, and frequency of initial AC component

[0, 0, 60] (default) | vector of three elements | matrix of N-by-3 elements

Magnitude of the initial AC component of the input signal, its phase, in degrees, and its frequency, in hertz.

When the input is vectorized (N signals), specify an N-by-3 matrix, where each row of the matrix corresponds to a particular input.

Dependencies

To enable this parameter, select the **Initialize filter states** parameter.

DC initial input — Initial DC component of input signal

0 (default) | vector of 1-by-N elements

Initial DC component of the input signal. When the input signal is vectorized, specify a 1-by-N vector, where each value corresponds to a particular input.

The **DC initial input** parameter is visible only when the **Initialize filter states** parameter is selected.

Dependencies

To enable this parameter, select the **Initialize filter states** parameter.

Plot filter response — Option to plot filter response

off (default) | on

Select this parameter to plot the filter step response and its Bode diagram (magnitude and phase of transfer function as a function of frequency) in a figure.

Frequency range (Hz): [Start, End, Inc.] — Frequency range for plotting filter Bode diagram

[0, 500, 1] (default) | vector of three elements

Frequency range for plotting the filter Bode diagram. Specify a vector containing the starting frequency, the end frequency, and the incremental frequency, in hertz.

Dependencies

To enable this parameter, select the **Plot filter response** parameter.

Version History

Introduced in R2013a

See Also

Second-Order Filter (Variable-Tuned) | First-Order Filter

Second-Order Filter (Variable-Tuned)

Implement second-order variable-tuned filter



Library

Simscape / Electrical / Specialized Power Systems / Control / Filters

Description

Depending on the **Filter type** selected in the block menu, the Second-Order Filter block implements the following transfer function. The Fn input determines the filter natural frequency $f_n = \omega_n/(2\pi)$ of the filter.

Lowpass filter:

$$H(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

Highpass filter:

$$H(s) = \frac{s^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

Bandpass filter:

$$H(s) = \frac{2\zeta\omega_n s}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

Bandstop (Notch) filter:

$$H(s) = \frac{s^2 + \omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

s = Laplace operator

ω_n = natural frequency; $\omega_n = 2\pi f_n$

ζ = damping ratio (called Zeta in the block menu)

Parameters

Filter type

Specify the type of filter: Lowpass, Highpass, Bandpass, or Bandstop (Notch) (default).

Initial natural frequency fn (Hz)

The initial natural frequency of the filter, in hertz. This value must be a scalar or a vector. Default is 120.

Damping ratio Zeta (Q = 1/(2*Zeta))

Specify the damping ratio of the filter. The damping ratio is typically a value between 0 and 1. Default is 0.707.

The damping ratio is related to the filter quality factor Q:

$$Q = \frac{1}{2\zeta}$$

For a bandpass or a bandstop filter, the 3 dB bandwidth is given by

$$BW = \frac{f_n}{Q} = 2\zeta f_n$$

Sample time

Specify the sample time of the block, in seconds. Set to 0 to implement a continuous block. Default is 0.

Initialize filter states

When this check box is selected, filter states are initialized according to the **AC initial input** and **DC initial input** parameters. Default is selected.

AC initial input: [Mag, Phase (degrees), Freq (Hz)]

Specify the magnitude of the initial AC component of the input signal, its phase, in degrees, and its frequency, in hertz. Default is [0, 0, 60].

When the input is vectorized (N signals), specify an N-by-3 matrix, where each row of the matrix corresponds to a particular input.

The **AC initial input** parameter is visible only when the **Initialize filter states** parameter is selected.

DC initial input

Specify the value of the initial DC component of the input signal. When the input signal is vectorized, specify a 1-by-N vector, where each value corresponds to a particular input. Default is 0.

The **DC initial input** parameter is visible only when the **Initialize filter states** parameter is selected.

Inputs and Outputs

The inputs accept vectorized signals of N values, thus implementing N filters. This capability is particularly useful for designing controllers in three-phase systems (N = 3).

Characteristics

Direct Feedthrough	Yes

Sample Time	Specified in the Sample Time parameter Continuous if Sample Time = 0
Scalar Expansion	Yes, of the parameters
States	Two states per filter
Dimensionalized	Yes

Examples

The `power_SecondOrderFilterTuned` example shows various uses of the Second-Order Filter (Variable-Tuned) block with two **Filter type** parameter settings (Lowpass and Bandstop).

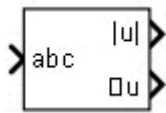
The model sample time is parameterized with variable `Ts` (default value `Ts = 50e-6`). To simulate continuous filters, specify `Ts = 0` in the MATLAB Command Window before starting the simulation.

Version History

Introduced in R2013a

Sequence Analyzer

Compute positive-, negative-, and zero-sequence components of three-phase signal



Libraries:
 Simscape / Electrical / Specialized Power Systems / Sensors and Measurements

Description

The Sequence Analyzer block outputs the magnitude and phase of the positive-, negative-, and zero-sequence components of a set of three balanced or unbalanced signals. Index 1 denotes the positive sequence, index 2 denotes the negative sequence, and index 0 denotes the zero sequence. The signals can optionally contain harmonics. The three sequence components of a three-phase signal (voltages V_1 V_2 V_0 or currents I_1 I_2 I_0) are computed as follows:

$$V_1 = \frac{1}{3}(V_a + a \cdot V_b + a^2 \cdot V_c)$$

$$V_2 = \frac{1}{3}(V_a + a^2 \cdot V_b + a \cdot V_c)$$

$$V_0 = \frac{1}{3}(V_a + V_b + V_c)$$

V_a, V_b, V_c = three voltage phasors at the specified frequency

$a = e^{j2\pi/3} = 1 \angle 120^\circ$ complex operator

A Fourier analysis over a sliding window of one cycle of the specified frequency is first applied to the three input signals. It evaluates the phasor values V_a , V_b , and V_c at the specified fundamental or harmonic frequency. Then the transformation is applied to obtain the positive sequence, negative sequence, and zero sequence.

As the block uses a running average window to perform the Fourier analysis, one cycle of simulation must complete before the outputs give the correct magnitude and angle. For example, the block response to a step change of V_1 is a one-cycle ramp. For the first cycle of simulation, the output is held constant using the values specified by the initial input parameters.

Examples

The `power_SequenceAnalyzer` model shows the use of the Sequence Analyzer block to compute the three sequence components of a three-phase sinusoidal voltage.

The model sample time The model sample time is parameterized by the T_s variable set to a default value of 50e-6 s. Set T_s to 0 in the command window to simulate the model in continuous mode.

Characteristics

Sample Time	Specified in the Sample Time parameter Continuous if Sample Time = 0

Scalar Expansion	No
Dimensionalized	No

Ports

Input

abc — Three-phase current signal
vector

Three-phase current signal. Index 1 denotes the positive sequence, index 2 denotes the negative sequence, and index 0 denotes the zero sequence.

Output

|u| — Magnitude
scalar

Outputs the magnitude (peak value) of the specified sequence component(s), in the same units as the abc input signal.

∠u — Phase
scalar

Outputs the phase, in degrees, of the specified components.

Parameters

Fundamental frequency (Hz) — Fundamental frequency of input signal
60 (default) | scalar

Specify the fundamental frequency, in hertz, of the three-phase input signal.

Harmonic n (1=fundamental) — Harmonic component to evaluate sequences
1 (default) | scalar

Specify the harmonic component to evaluate the sequences. Set to 1 to compute the sequences at the fundamental frequency or to the number corresponding to the desired harmonic.

Sequence — Specify sequence component
Positive Negative Zero (default) | Positive | Negative | Zero

Specify the sequence component the block outputs. Select Positive Negative Zero to calculate all the sequences.

Initial input [Mag, Phase (degrees)] — Initial magnitude and phase of input signal
[1, 0] (default) | vector

Specify the initial magnitude and phase, in degrees, of the positive-sequence component of the input signal.

Sample time — Block sample time
0 (default) | positive scalar

Specify the sample time of the block, in seconds

Version History

Introduced in R2013a

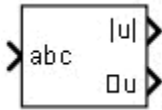
Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

Sequence Analyzer (Phasor)

Compute sequence components (positive, negative, and zero) of three-phase phasor signal



Library

Simscape / Electrical / Specialized Power Systems / Sensors and Measurements

Description

The Sequence Analyzer (Phasor) block computes the three-sequence phasor components (positive sequence u_1 , negative sequence u_2 , and zero sequence u_0) of a three-phase phasor signal as follows:

$$\begin{bmatrix} u_1 \\ u_2 \\ u_0 \end{bmatrix} = \frac{1}{3} \begin{bmatrix} 1 & a & a^2 \\ 1 & a^2 & a \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} u_a \\ u_b \\ u_c \end{bmatrix}$$

where a is the complex operator performing a $2\pi/3$ rotation:

$$a = e^{j\frac{2\pi}{3}}$$

Parameters

Sequence

Select the sequence components: Positive (default), Negative, Zero, or all three sequences
Positive Negative Zero.

Inputs and Outputs

abc

The three-phase phasor signal $[u_a \ u_b \ u_c]$.

mag u

Returns the magnitude of the selected sequence phasor component.

angle u

Returns the angle, in degrees, of the selected sequence phasor component.

Characteristics

Direct Feedthrough	Yes
Dimensionalized	No

Examples

The `power_PhaseMeasurements` example shows the use of the Sequence Analyzer (Phasor) block.

Version History

Introduced in R2013a

Series RLC Branch

Implement series RLC branch



Libraries:

Simscape / Electrical / Specialized Power Systems / Passives

Description

The Series RLC Branch block implements a single resistor, inductor, or capacitor, or a series combination of these.

Examples

Obtain the frequency response of a fifth-harmonic filter (tuned frequency = 300 Hz) connected on a 60 Hz power system. This example is available in the `power_seriesbranch` model.

To open this example, at the MATLAB Command Window, enter `power_seriesbranch`.

The network impedance in the Laplace domain is

$$Z(s) = \frac{V(s)}{I(s)} = \frac{LCs^2 + RCs + 1}{Cs}.$$

To obtain the frequency response of the impedance, get the state-space model (A B C D matrices) of the system.

This system is a one-input (Vsource) and one-output (Current Measurement block) system.

Note If you have Control System Toolbox software installed, you can use the `bode` function to get the transfer function $Z(s)$ from the state-space matrices:

```
[A,B,C,D] = power_analyze('power_seriesbranch');
freq = logspace(1,4,500);
w = 2*pi*freq;
[Ymag,Yphase] = bode(A,B,C,D,1,w);
% invert Y(s) to get Z(s)
Zmag = 1./Ymag;
Zphase = -Yphase;
subplot(2,1,1)
loglog(freq,Zmag)
grid
title('5th harmonic filter')
xlabel('Frequency, Hz')
ylabel('Impedance Zmag')
subplot(2,1,2)
semilogx(freq,Zphase)
xlabel('Frequency, Hz')
ylabel('phase Z')
grid
```

You can also use the Impedance Measurement block and the Powergui block to plot the impedance as a function of frequency. In order to measure the impedance you must disconnect the voltage source.

Ports

Conserving

+ — Positive terminal
specialized electrical

Specialized electrical conserving port associated with the positive terminal.

- — Negative terminal
specialized electrical

Specialized electrical conserving port associated with the negative terminal.

Parameters

Branch type — Branch type
RLC (default) | R | L | C | RL | RC | LC | Open circuit

Elements you want to include in the branch. The R letter defines the resistor, the L letter defines the inductor, and the C letter defines the capacitor. Set this parameter to `Open circuit` to define an open circuit ($R=0$, $L=0$, $C=\infty$). The block icon displays only existing elements.

Resistance (Ohms) — Branch resistance
1 (default) | scalar

Branch resistance, in Ohms.

Dependencies

To enable this parameter, set the **Branch type** parameter to RLC, R, RL, or RC.

Inductance (H) — Branch inductance
 $1e-3$ (default) | scalar

Branch inductance, in H.

Dependencies

To enable this parameter, set the **Branch type** parameter to RLC, L, RL, or LC.

Set the initial inductor current — Option to set the initial inductor current
off (default) | on

Select this option to specify the initial inductor current. If you clear this parameter, the software calculates the initial inductor current in order to start the simulation in steady state.

Dependencies

To enable this parameter, set the **Branch type** parameter to RLC, L, RL, or LC.

Inductor initial current (A) — Initial inductor current
0 (default) | scalar

Initial inductor current, in A, to use at the start of the simulation.

Dependencies

To enable this parameter, select the **Set the initial inductor current** parameter.

Capacitance (F) — Branch capacitance

1e-6 (default) | scalar

Branch capacitance, in F.

Dependencies

To enable this parameter, set the **Branch type** parameter to RLC, C, RC, or LC.

Set the initial capacitor voltage — Option to set the initial capacitor voltage

off (default) | on

Select this option to specify the initial capacitor voltage. If you clear this parameter, the software calculates the initial capacitor voltage in order to start the simulation in steady state.

Dependencies

To enable this parameter, set the **Branch type** parameter to RLC, C, RC, or LC.

Capacitor initial voltage (V) — Capacitor initial voltage

0 (default) | scalar

Initial capacitor voltage, in V, to use at the start of the simulation.

Dependencies

To enable this parameter, select the **Set the initial capacitor voltage** parameter.

Measurements — Measurements

None (default) | Branch voltage | Branch current | Branch voltage and current

Measurements to use during simulation.

Set this parameter to **Branch voltage** to measure the voltage across the terminals.

Set this parameter to **Branch current** to measure the current flowing through the block.

Set this parameter to **Branch voltage and current** to measure the voltage across the terminals and the current flowing through the block.

Place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurements** list box of the Multimeter block, the measurement is identified by a label followed by the block name:

Measurement	Label
Branch voltage	Ub :
Branch current	Ib :

Version History

Introduced before R2006a

Extended Capabilities

C/C++ Code Generation

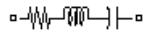
Generate C and C++ code using Simulink® Coder™.

See Also

Multimeter | Parallel RLC Branch | Parallel RLC Load | Series RLC Load

Series RLC Load

Implement linear series RLC load

**Libraries:**

Simscape / Electrical / Specialized Power Systems / Passives

Description

The Series RLC Load block implements a linear load as a series combination of R L C elements. At the specified frequency, the load exhibits a constant impedance. The active and reactive powers absorbed by the load are proportional to the square of the applied voltage. Only elements associated with nonzero powers are displayed in the block icon.

Examples

The `power_seriesload` example uses a Series RLC Load block to implement a simple load.

Ports

Conserving

+ — Positive terminal
specialized electrical

Specialized electrical conserving port associated with the positive terminal.

- — Negative terminal
specialized electrical

Specialized electrical conserving port associated with the negative terminal.

Parameters

Parameters

Nominal voltage Vn (Vrms) — Nominal voltage of load
1000 (default) | nonnegative scalar

Nominal voltage of the load, in volts RMS.

Nominal frequency fn (Hz) — Nominal frequency
60 (default) | nonnegative scalar

Nominal frequency, in hertz.

Active power P (W) — Active power of load
10e3 (default) | scalar

Active power of the load, in watts.

Inductive reactive power QL (positive var) — Inductive reactive power

100 (default) | nonnegative scalar

Inductive reactive power QL, in vars. Specify a positive value, or 0.

Capacitive reactive power Qc (negative var) — Capacitive reactive power

100 (default) | nonnegative scalar

Capacitive reactive power Qc, in vars. Specify a positive value, or 0.

Set the initial capacitor voltage — Option to set the initial capacitor voltage

off (default) | on

Select this option to specify the initial capacitor voltage. If you clear this parameter, the software calculates the initial capacitor voltage in order to start the simulation in steady state.

Capacitor initial voltage (V) — Capacitor initial voltage

0 (default) | scalar

Initial capacitor voltage, in V, to use at the start of the simulation.

Dependencies

To enable this parameter, select the **Set the initial capacitor voltage** parameter.

Set the initial inductor current — Option to set the initial inductor current

off (default) | on

Select this option to specify the initial inductor current. If you clear this parameter, the software calculates the initial inductor current in order to start the simulation in steady state.

Inductor initial current (A) — Initial inductor current

0 (default) | scalar

Initial inductor current, in A, to use at the start of the simulation.

Dependencies

To enable this parameter, select the **Set the initial inductor current** parameter.

Measurements — Measurements

None (default) | Branch voltage | Branch current | Branch voltage and current

Measurements to use during simulation.

Select **Branch voltage** to measure the voltage across the Series RLC Load block terminals.

Select **Branch current** to measure the current flowing through the Series RLC Load block.

Select **Branch voltage and current** to measure the voltage and the current of the Series RLC Load block.

Place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurements** list box of the Multimeter block, the measurement is identified by a label followed by the block name:

Measurement	Label
Branch voltage	Ub:
Branch current	Ib:

Load Flow

The Load Flow tool of the powergui block uses the parameters on this tab. These load flow parameters affect only model initialization. They do not affect simulation.

Load type — Option to specify the load type

constant Z (default) | constant PQ | constant I

Option to specify the load type.

If you select **constant Z**, the load impedance is determined from the nominal phase-to-phase voltage V_n , active power P , and reactive power (QL-QC) specified on the **Parameters** tab of the block dialog box. During the load flow solution, the impedance is kept constant. The effective P and Q therefore vary proportionally to the square of the bus voltage computed by the Load Flow tool.

If you select **constant PQ**, the active power P and reactive power Q are kept constant and equal to the values specified on the **Parameters** tab of the block dialog box. When you apply the load flow solution to the model (by clicking **Apply** in the Load Flow tool), the **Nominal phase-to-phase voltage V_n** parameter on the **Parameters** tab is automatically adjusted to the phase-to-phase bus voltage computed by the Load Flow tool.

If you select **constant I**, the load current is kept constant at its nominal value determined from the base voltage specified in the dialog box of the associated Load Flow Bus block and from active power P and reactive power Q_L and Q_C specified on the **Parameters** tab of the block dialog box. The effective P and Q therefore vary proportionally to the bus voltage computed by the Load Flow tool. When you apply the load flow solution to the model (by clicking **Apply** in the Load Flow tool), the pu value of the nominal voltage V_n on the **Parameters** tab is automatically adjusted to the square root of the load voltage (in pu) computed by the Load Flow tool.

Version History

Introduced before R2006a

Extended Capabilities

C/C++ Code Generation

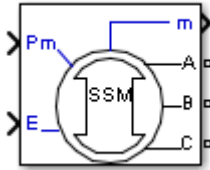
Generate C and C++ code using Simulink® Coder™.

See Also

Multimeter | Parallel RLC Branch | Parallel RLC Load | Series RLC Branch

Simplified Synchronous Machine

Model the dynamics of simplified three-phase synchronous machine



Libraries:

Simscape / Electrical / Electromechanical / Synchronous

Description

The Simplified Synchronous Machine block models both the electrical and mechanical characteristics of a simple synchronous machine.

The electrical system for each phase consists of a voltage source in series with an RL impedance, which implements the internal impedance of the machine. The value of R can be zero but the value of L must be positive.

The Simplified Synchronous Machine block implements the mechanical system described by

$$\Delta\omega(t) = \frac{1}{2H} \int_0^t (T_m - T_e) - K_d \Delta\omega(t) dt$$

$$\omega(t) = \Delta\omega(t) + \omega_0,$$

where

$\Delta\omega$ = Speed variation with respect to speed of operation (1-12)

H = constant of inertia

T_m = mechanical torque

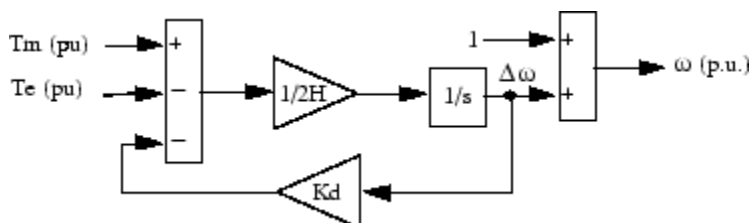
T_e = electromagnetic torque

K_d = damping factor representing the effect of damper windings

$\omega(t)$ = mechanical speed of the rotor

ω_0 = speed of operation (1 p.u.)

Although the parameters can be entered in either SI units or per unit in the dialog box, the internal calculations are done in per unit. The following block diagram illustrates how the mechanical part of the model is implemented. The model computes a deviation with respect to the speed of operation; not the absolute speed itself.



The K_d damping coefficient simulates the effect of damper windings normally used in synchronous machines. When the machine is connected to an infinite network (zero impedance), the variation of machine power angle δ resulting from a change of mechanical power (P_m) can be approximated by the following second-order transfer function:

$$\frac{\delta}{P_m} = \frac{\omega_s/(2H)}{s^2 + 2\zeta\omega_n s + \omega_n^2},$$

where

δ	Power angle delta: angle of internal voltage E with respect to terminal voltage, in radians
P_m	Mechanical power in pu
ω_n	Frequency of electromechanical oscillations = $\sqrt{\omega_s P_{\max}/(2H)}$ in rad/s
ζ	Damping ratio = $(K_d/4)\sqrt{2/(\omega_s H P_{\max})}$
ω_s	Electrical frequency in rad/s
$P_{\max}$	Maximum power in pu transmitted through reactance X at terminal voltage V_t and internal voltage E . $P_{\max} = V_t E/X$, where V_t , E , and X are in pu
H	Inertia constant(s)
K_d	Damping factor (pu_of_torque / pu_of_speed)

This approximate transfer function, which has been derived by assuming $\sin(\delta) = \delta$, is valid for small power angles ($\delta < 30$ degrees). It follows from the preceding ζ expression that the K_d value required to obtain a given ζ damping ratio:

$$K_d = 4\zeta\sqrt{\omega_s H P_{\max}/2}.$$

Examples

The `power_simplealt` example uses the Simplified Synchronous Machine block to represent a 1000 MVA, 315 kV, 60 Hz equivalent source connected to an infinite bus (Three-Phase Programmable Voltage Source block). The Simplified Synchronous Machine (SI Units) block is used as a synchronous generator. The internal resistance and reactance are set respectively to 0.02 pu (1.9845 Ω) and 0.2 pu ($X = 19.845 \Omega$; $L = 0.0526$ H). The inertia of the machine is $J = 168,870 \text{ kg.m}^2$, corresponding to an inertia constant $H = 3$ s. The electrical frequency is $\omega_s = 2\pi \cdot 60 = 377$ rad/s. The machine has two pairs of poles such that its synchronous speed is $2\pi \cdot 60/2 = 188.5$ rad/s or 1800 rpm.

The Load Flow Analyzer app has been used to initialize the machine to start simulation in steady state with the machine generating 500 MW. The required internal voltage computed by the load flow is 1.0149 pu. Therefore, an internal voltage $E = 315\sqrt{3} \cdot 1.0149 = 319,690$ Vrms phase-to-phase is specified in the Constant block connected to the E input. The maximum power that can be delivered by the machine with a terminal voltage $V_t = 1.0$ pu and an internal voltage $E = 1.0149$ pu is $P_{\max} = V_t \cdot E/X = 1.0149/0.2 = 5.0745$ pu.

The damping factor K_d is adjusted to obtain a damping ratio $\zeta = 0.3$. The required K_d value is:

$$K_d = 4\zeta\sqrt{\omega_s H P_{\max}/2} = 64.3$$

Two Fourier blocks measure the power angle δ . This angle is computed as the difference between the phase angle of phase A internal voltage and the phase angle of phase A terminal voltage.

In this example, a step is performed on the mechanical power applied to the shaft. The machine is initially running in steady state with a mechanical power of 505 MW (mechanical power required for an output electrical power of 500 MW, considering the resistive losses). At $t = 0.5$ s the mechanical power is suddenly increased to 1000 MW.

Run the example and observe the electromechanical transient on the Scope block displaying the power angle δ in degrees, the machine speed in rpm, and the electrical power in MW.

For an initial electrical power $P_e = 500$ MW (0.5 pu), the load angle δ is 5.65 degrees, which corresponds to the expected value:

$$P_e = \frac{V_t E \sin \delta}{X} = \frac{1.0 \cdot 1.0149 \cdot \sin(5.65^\circ)}{0.2} = 0.5 \text{ p.u.}$$

As the mechanical power is stepped from 0.5 pu to 1.0 pu, the load angle increases and goes through a series of under damped oscillations (damping ratio $\zeta = 0.3$) before stabilizing to its new value of 11.3 degrees. The frequency of the oscillations is given by:

$$f_n = \frac{1}{2\pi} \sqrt{\frac{\omega_s P_{\max}}{2H}} = 2.84 \text{ Hz.}$$

Assumptions and Limitations

- The electrical system of the Simplified Synchronous Machine block consists solely of a voltage source behind a synchronous reactance and resistance. All the other self- and magnetizing inductances of the armature, field, and damping windings are neglected. The effect of damper windings is approximated by the damping factor K_d . The three voltage sources and RL impedance branches are Y-connected (three wires or four wires). The load might or might not be balanced.
- When you use Simplified Synchronous Machine blocks in discrete systems, you might have to use a small parasitic resistive load, connected at the machine terminals, to avoid numerical oscillations. Large sample times require larger loads. The minimum resistive load is proportional to the sample time. Remember that with a 25 μ s time step on a 60 Hz system, the minimum load is approximately 2.5% of the machine nominal power. For example, a 200 MVA simplified synchronous machine in a power system discretized with a 50 μ s sample time requires approximately 5% of resistive load or 10 MW. If the sample time is reduced to 20 μ s, a resistive load of 4 MW should be sufficient.

Ports

Input

Pm — Mechanical power, watts
scalar

Mechanical power supplied to the machine, in watts. The input can be a constant signal or it can be connected to the output of the Hydraulic Turbine and Governor block. The frequency of the internal voltage sources depends on the mechanical speed of the machine.

Dependencies

To enable this port, set the **Mechanical input** parameter to Mechanical power Pm.

w — Machine speed, rad/s
scalar

Machine speed, in rad/s.

Dependencies

To enable this port, set the **Mechanical input** parameter to `Mechanical power Pm`.

E — Amplitude of internal voltages, volts or pu
scalar

Amplitude of the internal voltages of the block. It can be a constant signal or it can be connected to the output of a voltage regulator. If you use the SI units machine, this input must be in volts phase-to-phase RMS. If you use the pu units machine, it must be in pu.

Dependencies

To enable this port, set the **Mechanical input** parameter to `Mechanical power Pm`.

Output

m — Measurement signals
12-element vector

Measurement signals, returned as a 12-element vector. You can demultiplex these signals by using the Bus Selector block.

Name	Definition	Units
ias	Stator current is_a	A or pu
ibs	Stator current is_b	A or pu
ics	Stator current is_c	A or pu
va	Terminal voltage Va	V or pu
vb	Terminal voltage Vb	V or pu
vc	Terminal voltage Vc	V or pu
ea	Internal voltage Ea	V or pu
eb	Internal voltage Eb	V or pu
ec	Internal voltage Ec	V or pu
theta	Rotor angle theta	rad
w	Rotor speed wm	rad/s
Pe	Electrical power Pe	W

Conserving

N — Neutral terminal
specialized electrical

Electrical conserving port associated with the neutral terminal.

Dependencies

To enable this port, set the **Connection type** parameter to `4-wire Y`.

A — Phase A terminal
specialized electrical

Electrical conserving port associated with the phase A terminal.

B — Phase B terminal
specialized electrical

Electrical conserving port associated with the phase B terminal.

C — Phase C terminal
specialized electrical

Electrical conserving port associated with the phase C terminal.

S — Machine rotor shaft
mechanical

Mechanical rotational conserving port associated with the machine rotor shaft.

Dependencies

To enable this port, set the **Mechanical input** parameter to `Mechanical rotational port`.

Parameters

In the Machines library you can choose between the SI units or the pu units Simplified Synchronous Machine blocks to specify the electrical and mechanical parameters of the model. These two blocks simulate exactly the same simplified synchronous machine model; the only difference is how you enter the parameter units.

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Configuration

Connection type — Number of wires in three-phase Y connection
3-wire Y (default) | 4-wire Y

Specify the number of wires used in the three-phase Y connection: three-wire (neutral not accessible) or four-wire (neutral is accessible). Default is 3-wire Y.

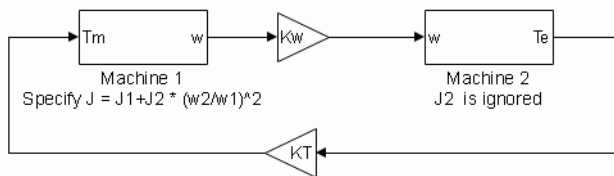
Mechanical input — Option to select mechanical input
Mechanical power P_m (default) | Speed w | Mechanical rotational port

Select the mechanical power applied to the shaft or the rotor speed as a Simulink input of the block, or to represent the machine shaft by a Simscape rotational mechanical port.

Select `Mechanical power  $P_m$` (default) to specify a mechanical power input, in W or in pu, and change labeling of the block input to P_m . The machine speed is determined by the machine Inertia J (or inertia constant H for the pu machine) and by the difference between the mechanical torque T_m , resulting from the applied mechanical power P_m , and the internal electromagnetic torque T_e . The sign convention for the mechanical power is when the speed is positive, a positive mechanical power signal indicates generator mode and a negative signal indicates motor mode.

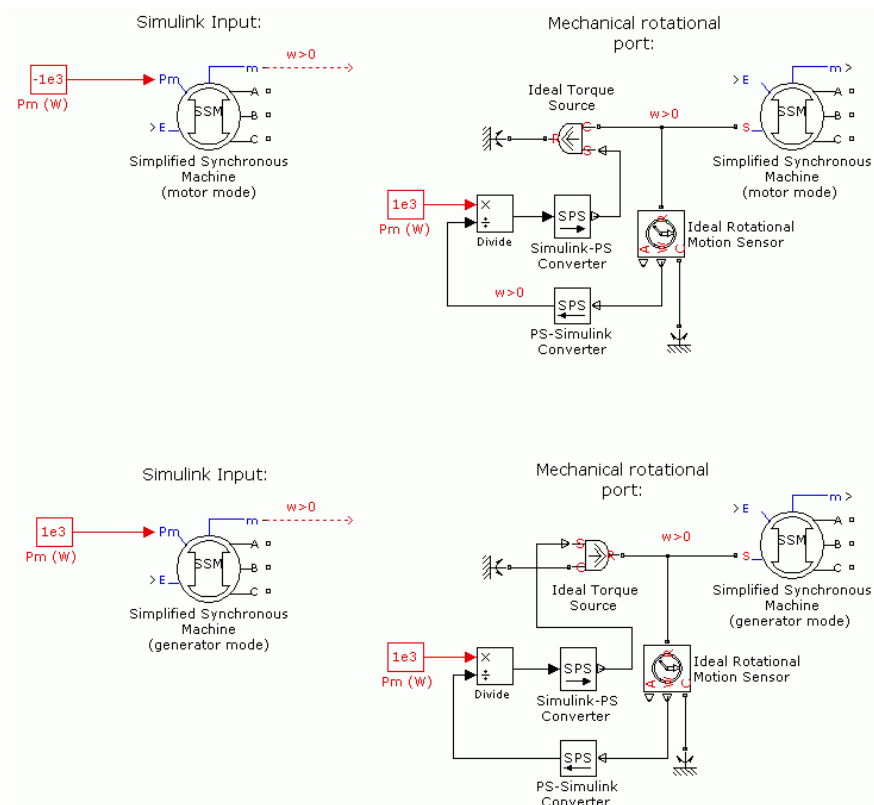
Select **Speed** w to specify a speed input, in rad/s or in pu, and change labeling of the block input to w . The machine speed is imposed and the mechanical part of the model (inertia constant H) is ignored. Using the speed as the mechanical input allows modeling a mechanical coupling between two machines.

The next figure indicates how to model a stiff shaft interconnection in a motor-generator set when friction torque is ignored in machine 2. The speed output of machine 1 (motor) is connected to the speed input of machine 2 (generator), while machine 2 electromagnetic torque output T_e is applied to the mechanical torque input T_m of machine 1. The K_w factor takes into account speed units of both machines (pu or rad/s) and gear box ratio w_2/w_1 . The K_T factor takes into account torque units of both machines (pu or N.m) and machine ratings. Also, because the inertia J_2 is ignored in machine 2, J_2 referred to machine 1 speed must be added to machine 1 inertia J_1 .



Select **Mechanical rotational port** to add to the block a Simscape mechanical rotational port that allows connection of the machine shaft with other Simscape blocks having mechanical rotational ports. The Simulink input representing the mechanical power P_m or the speed w of the machine is then removed from the block.

The next figure indicates how to connect an Ideal Torque Source block from the Simscape library to the machine shaft to represent the machine in motor mode, or in generator mode, when the rotor speed is positive.



Use signal names to identify bus labels — Option to use signal names to identify bus labels
off (default) | on

When this check box is selected, the measurement output uses the signal names to identify the bus labels. Select this option for applications that require bus signal labels to have only alphanumeric characters.

When this check box is cleared, the measurement output uses the signal definition to identify the bus labels. The labels contain nonalphanumeric characters that are incompatible with some Simulink applications.

Parameters

Nominal power, line-to-line voltage, and frequency — Nominal power, line-to-line voltage, and frequency

[187e6 13800 60] (default) | vector of three positive values

The nominal apparent power P_n (VA), frequency f_n (Hz), and RMS line-to-line voltage V_n (V). Computes nominal torque and converts SI units to pu.

Inertia, damping factor and pairs of poles — Inertia, damping factor and pairs of poles

[3.7 0 20] (default) | [3.895e6 0 20] | vector of nonnegative values

The inertia (J in $\text{kg}\cdot\text{m}^2$ or H in seconds) damping factor (K_d) and number of pairs of poles (p). The damping factor must be specified in (pu of torque)/(pu of speed) in both machine dialog boxes (in pu and in SI). Default is [3.7 0 20] for pu and [3.895e6 0 20] for SI.

Internal impedance — Resistance and reactance for each phase

[0.02 0.3] (default) | [0.0204 0.8104e-3] | positive scalar

The resistance R (Ω or pu) and reactance L (H or pu) for each phase. Default is [0.02 0.3] for pu and [0.0204 0.8104e-3] for SI.

Initial conditions — Machine initial conditions

[0,0 0,0,0 0,0,0] (default) | vector of positive values

The initial speed deviation (% of nominal), rotor angle (degrees), line current magnitudes (A or pu), and phase angles (degrees). These values are automatically computed by the load flow utility of the Powergui block. Default is [0,0 0,0,0 0,0,0].

Sample time (-1 for inherited) — Block sample time

-1 (default) | scalar

Specifies the sample time used by the block. To inherit the sample time specified in the Powergui block, set this parameter to -1. Default is -1.

Advanced

To enable the Advanced tab, set the **Simulation type** parameter of the powergui block to **Discrete** and, on the **Preferences** tab, clear the **Automatically handle discrete solver and Advanced tab solver settings of blocks** parameter.

Discrete solver model — Block integration method

Trapezoidal non iterative (default) | Trapezoidal robust | Backward Euler robust

Specifies the integration method used by the block. The choices are Trapezoidal non iterative (default), Trapezoidal robust, and Backward Euler robust.

When you select the **Automatically handle discrete solver and Advanced tab solver settings of blocks** parameter in the powergui block, the discrete solver model is automatically set to Trapezoidal robust.

Trapezoidal non iterative requires you to add a non-negligible shunt load at the machine terminals to maintain simulation stability, and the simulation may fail to converge and stop when the number of machines increases in the model.

Trapezoidal robust and Backward Euler robust allow you to eliminate the need to use parasitic loads. To eliminate the topological errors of the machines connected to an inductive circuit (for example, a circuit breaker connected in series with the machine), the machine models a negligible internal load of 0.01% of nominal power.

Trapezoidal robust is slightly more accurate than Backward Euler robust, especially when the model is simulated at larger sample times. Trapezoidal robust may produce slight damped numerical oscillations on machine voltage in no-load conditions, while Backward Euler robust prevents oscillations and maintains accuracy.

For more information on what method to use in your application, see “Simulating Discretized Electrical Systems”.

Sample time (-1 for inherited) — Block sample time

-1 (default) | scalar

Specifies the sample time used by the block. To inherit the sample time specified in the Powergui block, set this parameter to -1 . Default is -1 .

Load Flow

The load flow parameters define block parameters for use with the Load Flow tool of the Powergui block. These load flow parameters are for model initialization only. They have no impact on the block model or on the simulation performance.

The configuration of the **Load Flow** tab depends on the option selected for the **Generator type** parameter.

Generator type — Generator type of voltage source

PV (default) | swing | PQ

Specify the generator type of the voltage source.

Select **swing** to implement a generator controlling magnitude and phase angle of its terminal voltage. The reference voltage magnitude and angle are specified by the **Swing bus or PV bus voltage** and **Swing bus voltage angle** parameters of the Load Flow Bus block connected to the machine terminals.

Select **PV** (default) to implement a generator controlling its output active power P and voltage magnitude V . P is specified by the **Active power generation P** parameter of the block. V is specified by the **Swing bus or PV bus voltage** parameter of the Load Flow Bus block connected to the machine terminals. You can control the minimum and maximum reactive power generated by the block by using the **Minimum reactive power Qmin** and **Maximum reactive power Qmax** parameters.

Select **PQ** to implement a generator controlling its output active power P and reactive power Q . P and Q are specified by the **Active power generation P** and **Reactive power generation Q** parameters of the block, respectively.

Active power generation P — Active power generated by machine

0 (default) | scalar

Specify the active power that you want generated by the machine, in watts. When the machine operates in motor mode, you specify a negative value.

Dependencies

To enable this parameter, set **Generator type** to PV or PQ.

Reactive power generation Q — Reactive power generated by machine

0 (default) | scalar

Specify the reactive power that you want generated by the machine, in vars. A negative value indicates that the reactive power is absorbed by the machine.

Dependencies

To enable this parameter, set **Generator type** to PQ.

Minimum reactive power Qmin — Minimum reactive power generated by the machine

$-\infty$ (default) | scalar

Minimum reactive power that can be generated by the machine while keeping the terminal voltage at its reference value. This reference voltage is specified in the **Swing bus or PV bus voltage** parameter of the Load Flow Bus block connected to the machine terminals. The default value is `-inf`, which means that there is no lower limit on the reactive power output.

Dependencies

To enable this parameter, set **Generator type** to PV.

Maximum reactive power Qmax — Maximum reactive power generated by the machine
`inf` (default) | scalar

Maximum reactive power that can be generated by the machine while keeping the terminal voltage at its reference value. This reference voltage is specified in the **Swing bus or PV bus voltage** parameter of the Load Flow Bus block connected to the machine terminals. The default value is `inf`, which means that there is no upper limit on the reactive power output.

Dependencies

To enable this parameter, set **Generator type** to PV.

Version History

Introduced before R2006a

Extended Capabilities

C/C++ Code Generation

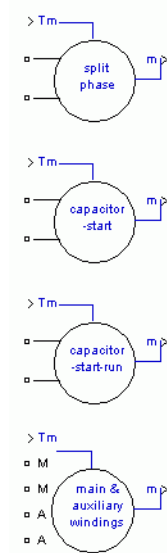
Generate C and C++ code using Simulink® Coder™.

See Also

Excitation System | Hydraulic Turbine and Governor | powergui | Steam Turbine and Governor | Synchronous Machine SI Fundamental | Synchronous Machine pu Fundamental | Synchronous Machine pu Standard

Single Phase Asynchronous Machine

Model dynamics of single phase asynchronous machine with squirrel-cage rotor



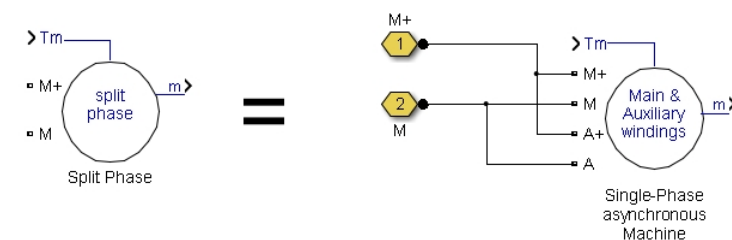
Library

Simscape / Electrical / Specialized Power Systems / Electrical Machines

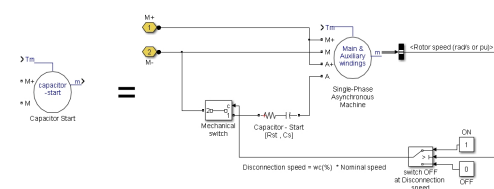
Description

This machine has two windings: main and auxiliary. With the model, you can simulate the split-phase, the capacitor-start, the capacitor-start-capacitor-run, and main & auxiliary windings operation modes.

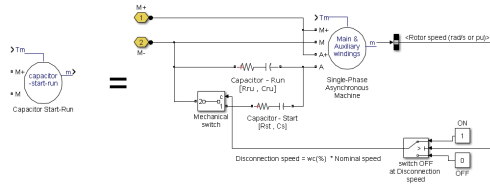
For the split-phase mode, the main and auxiliary windings are internally connected as follows:



For the capacitor-start mode, the main and auxiliary windings are internally connected as follows:



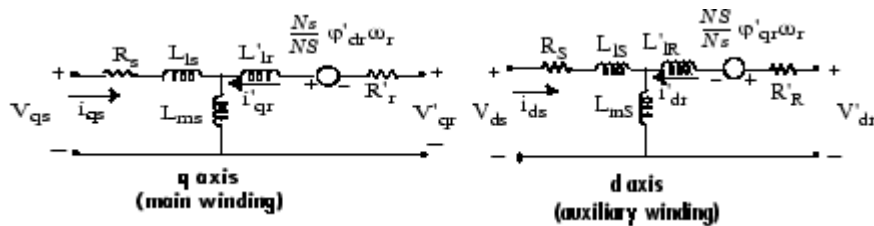
For the capacitor-start-capacitor-run mode, the main and auxiliary windings are internally connected as follows:



The electrical part of the machine is represented by a fourth-order state-space model and the mechanical part by a second-order system. All electrical variables and parameters are referred to the stator, indicated by the following prime signs in the machine equations. All stator and rotor quantities are in the stator reference frame (dq frame). The subscripts are defined in the following table.

Subscript	Definition
d	d axis quantity
q	q axis quantity
r	Referred to the main winding rotor quantity
R	Referred to the auxiliary winding rotor quantity
s	Main winding stator quantity
S	Auxiliary winding stator quantity
l	Leakage inductance
m	Magnetizing inductance

Electrical System



$$V_{qs} = R_s i_{qs} + d\phi_{qs}/dt$$

$$V_{ds} = R_s i_{ds} + d\phi_{ds}/dt$$

$$V'_{qr} = R'_r i'_{qr} + d\phi'_{qr}/dt - (N_s/N_r)\omega_r \phi'_{dr}$$

$$V'_{dr} = R'_r i'_{dr} + d\phi'_{dr}/dt + (N_s/N_r)\omega_r \phi'_{qr}$$

$$T_e = p[(N_s/N_r)\phi'_{qr} i'_{dr} - (N_s/N_r)\phi'_{dr} i'_{qr}]$$

where

$$\phi_{qs} = L_{ss} i_{qs} + L_{ms} i'_{qr}$$

$$\phi_{ds} = L_{ss} i_{ds} + L_{ms} i'_{dr}$$

$$\phi'_{qr} = L'_{rr} i'_{qr} + L_{ms} i_{qs}$$

$$\phi'_{dr} = L'_{rr} i'_{dr} + L_{ms} i_{ds}$$

$$L_{ss} = L_{ls} + L_{ms}$$

$$L_{ss} = L_{ls} + L_{ms}$$

$$L'_{rr} = L'_{lr} + L_{ms}$$

$$L'_{rr} = L'_{lr} + L_{ms}$$

Mechanical System

$$\frac{d}{dt}\omega_m = \frac{T_e - F\omega_m - T_m}{2H}$$

$$\frac{d}{dt}\theta_m = \omega_m.$$

Reference frame

The reference frame fixed in the stator converts voltages and currents to the dq frame.

The following relationships describe the ab-to-dq frame transformations applied to the single phase asynchronous machine.

$$\begin{bmatrix} f_{qs} \\ f_{ds} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} f_{as} \\ f_{bs} \end{bmatrix}$$

$$\begin{bmatrix} f_{qr} \\ f_{dr} \end{bmatrix} = \begin{bmatrix} \cos(\theta_r) & -\sin(\theta_r) \\ -\sin(\theta_r) & -\cos(\theta_r) \end{bmatrix} \begin{bmatrix} f_{ar} \\ f_{br} \end{bmatrix}.$$

The variable f can represent either voltage, currents, or flux linkage.

The single phase asynchronous machine block parameters are defined as follows (all quantities are referred to the stator).

Parameter	Definition
R_s, L_{ls}	Main winding stator resistance and leakage inductance
R_s, L_{ls}	Auxiliary winding stator resistance and leakage inductance
R'_r, L'_{lr}	Main winding rotor resistance and leakage inductance
R'_R, L'_{lR}	Auxiliary winding rotor resistance and leakage inductance. The two values are equal to the main winding rotor resistance and leakage inductances values, respectively.
L_{ms}	Main winding magnetizing inductance
L_{mS}	Auxiliary winding magnetizing inductance
L_{ss}, L'_{rr}	Total main winding stator and rotor inductances
L_{SS}, L'_{RR}	Total auxiliary winding stator and rotor inductances
V_{as}, i_{as}	Main winding stator voltage and current
V_{bs}, i_{bs}	Auxiliary winding stator voltage and current
V_{qs}, i_{qs}	q axis stator voltage and current
V'_{qr}, i'_{qr}	q axis rotor voltage and current
V_{ds}, i_{ds}	d axis stator voltage and current
V'_{dr}, i'_{dr}	d axis rotor voltage and current
ϕ_{qs}, ϕ_{ds}	Stator q and d axis fluxes
ϕ'_{qr}, ϕ'_{dr}	Rotor q and d axis fluxes
ω_m	Angular velocity of the rotor
Θ_m	Rotor angular position

Parameter	Definition
p	Number of pole pairs
ω_r	Electrical angular velocity ($\omega_m \times p$)
Θ_r	Electrical rotor angular position ($\Theta_m \times p$)
T_e	Electromagnetic torque
T_m	Shaft mechanical torque
J	Combined rotor and load inertia coefficient in (kg.m ²). Set to infinite to simulate locked rotor.
F	Combined rotor and load viscous friction coefficient.
H	Combined rotor and load inertia constant in (s). Set to infinite to simulate locked rotor.
N_s	Number of main winding's effective turns.
N_S	Number of auxiliary winding's effective turns.
R_{st}	Capacitor-Start resistance
C_s	Capacitor-Start
R_{run}	Capacitor-Run resistance
C_{run}	Capacitor-Run
N	Ratio of number of auxiliary winding's effective turns and number of main winding's effective turns.

Parameters

You can choose between two types of units to specify the electrical and mechanical parameters of the model, the per unit dialog box, and the SI dialog box. Both blocks are modeling the same machine. Depending on the dialog box that you use, Simscape Electrical Specialized Power Systems automatically converts the parameters that you specify into per unit parameters. The Simulink model of the Single Phase Asynchronous Machine block uses per unit parameters.

Configuration Tab

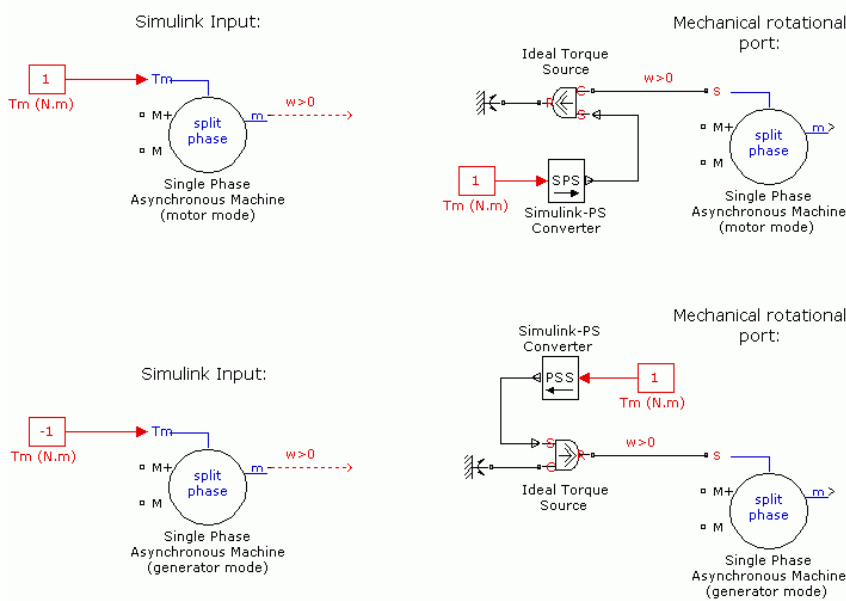
Mechanical input

Select the torque applied to the shaft as a Simulink input of the block, or to represent the machine shaft by a Simscape rotational mechanical port.

Select **Torque** T_m (default) to specify a torque input, in N.m or in pu, and change labeling of the block input to T_m . The machine speed is determined by the machine Inertia J (or inertia constant H for the pu machine) and by the difference between the applied mechanical torque T_m and the internal electromagnetic torque T_e . The sign convention for the mechanical torque is when the speed is positive, a positive torque signal indicates motor mode and a negative signal indicates generator mode.

Select **Mechanical rotational port** to add to the block a Simscape mechanical rotational port that allows connection of the machine shaft with other Simscape blocks that have mechanical rotational ports. The Simulink input representing the mechanical torque T_m of the machine is then removed from the block.

The next figure indicates how to connect an Ideal Torque Source block from the Simscape library to the machine shaft to represent the machine in motor mode, or in generator mode, when the rotor speed is positive.



Units

Specify the per unit dialog box or the SI dialog box. Default is SI.

Type of machine

Specify one of the four types of single phase asynchronous machines: Split Phase (default), Capacitor-Start, Capacitor-Start-Run, or Main & auxiliary windings.

Use signal names to identify bus labels

When this check box is selected, the measurement output uses the signal names to identify the bus labels. Select this option for applications that require bus signal labels to have only alphanumeric characters.

When this check box is cleared (default), the measurement output uses the signal definition to identify the bus labels. The labels contain nonalphanumeric characters that are incompatible with some Simulink applications.

Parameters Tab

Nominal power, voltage, and frequency

The nominal apparent power P_n (VA), RMS V_n (V), and frequency f_n (Hz). Default is [.25*746 110 60].

Main winding stator

The stator resistance R_s (Ω or pu) and leakage inductance L_{ls} (H or pu). Default is [2.02 7.4e-3] (SI) and [0.031135 0.042999] (pu).

Main winding rotor

The rotor resistance R_r (Ω or pu) and leakage inductance L_{lr} (H or pu), both referred to the stator. Default is [4.12 5.6e-3] (SI) and [0.063502 0.03254] (pu).

Main winding mutual inductance

The magnetizing inductance L_{ms} (H or pu). Default is 0.1772 (SI) and 1.0296 (pu).

Auxiliary winding stator

The stator resistance R_s (Ω or pu) and leakage inductance L_{ls} (H or pu). Note that the Auxiliary winding rotor parameters are assumed to be equal to the main winding rotor resistance and leakage inductances values. Therefore it is not required to specify them in the dialog box. Default is [7.14 8.5e-3] (SI) and [0.11005 0.049391] (pu).

Inertia, friction factor, pole pairs, turn ratio (aux/main)

For the **SI units** dialog box: the combined machine and load inertia coefficient J (kg.m^2), the combined viscous friction coefficient F (N.m.s), the number of pole pairs p and ratio of number of auxiliary winding's effective turns, and the number of main winding's effective turns. **pu units** dialog box: the inertia constant H (s), the combined viscous friction coefficient F (pu), and the number of pole pairs p . Default is [0.0146 0 2 1.18] (SI) and [1.3907 0 2 1.18] (pu).

Capacitor-Start

The start capacitance C_s (farad or pu) and capacitor series resistance R_{st} (Ω or pu). Default is [2 254.7e-6] (SI) and [0.030826 6.2297] (pu).

Capacitor-Run

The run capacitance C_{run} (farad or pu) and series resistance R_{run} (farad or pu). Default is [18 21.1e-6] (SI) and [0.27744 0.51608] (pu).

Disconnection speed

Specifies the speed (%) when the auxiliary winding may be disconnected. Default is 75.

Initial speed

Specifies the initial speed (%). Default is 0.

Advanced Tab**Sample time (–1 for inherited)**

Specifies the sample time used by the block. To inherit the sample time specified in the Powergui block, set this parameter to –1 (default).

Inputs and Outputs

T_m

The Simulink input of the block is the mechanical torque at the machine shaft. When you use the SI parameters mask, the input is a signal in N.m; otherwise it is in pu.

m

The Simulink output of the block is a vector containing measurement signals. You can demultiplex these signals by using the Bus Selector block provided in the Simulink library. Depending on the type of mask you use, the units are in SI, or in pu.

Name	Definition	Units
iar	Rotor current i_{r_a}	A or pu
ibr	Rotor current i_{r_b}	A or pu
iqr	Rotor current i_q	A or pu

Name	Definition	Units
idr	Rotor current i_d	A or pu
phiqr	Rotor flux ϕ_{ir_q}	V.s or pu
phidr	Rotor flux ϕ_{ir_d}	V.s or pu
ias	Main winding stator current i_a	A or pu
ibs	Auxiliary winding stator current i_b	A or pu
phiqs	Stator flux $\phi_{is\_q}(V.s)$	V.s or pu
phids	Stator flux $\phi_{is\_d}(V.s)$	V.s or pu
vc	Voltage capacitor V_c	V or pu
w	Rotor speed	rad/s
Te	Electromagnetic torque T_e	N.m or pu
theta	Rotor angle θ	rad

Limitations

The Single Phase Asynchronous Machine block does not include a representation of iron losses and saturation.

Examples

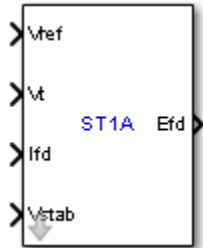
The “Single-Phase Asynchronous Machine” example shows the use of the Single Phase Asynchronous Machine block in two modes of operation.

References

[1] Krause, P.C., O. Wasynczuk, and S.D. Sudhoff, *Analysis of Electric Machinery*, IEEE Press, 1995.

ST1A Excitation System

Implements IEEE type ST1A excitation system model



Library

Simscape / Electrical / Specialized Power Systems / Electrical Machines / Synchronous Machine Control

Description

This block models an excitation system where the supply of power to the controlled bridge rectifier is provided by the generator terminals through a transformer to lower the voltage to an appropriate level. The voltage regulator gain and excitation system time constants are represented by K_a and T_a , respectively.

This block is an adaptation of the ST1A excitation system of the IEEE 421 standard, copyright IEEE 2005, all rights reserved.

Parameters

Controllers Tab

Low-pass filter time constant

The time constant T_r of the first-order system representing the stator terminal voltage transducer. Default is $20e-3$.

Voltage regulator gain and time constant

The gain K_a and time constant T_a of the first-order system representing the main regulator. Default is $[210 \ 0.001]$.

Voltage regulator input limits

The voltage regulator input limits V_{lmin} and V_{lmax} , in p.u. Default is $[-999 \ +999]$.

Voltage regulator internal limits

The voltage regulator internal limits V_{Amin} and V_{Amax} , in p.u. Default is $[-999 \ 999]$.

Voltage regulator output limits

The voltage regulator output limits V_{Rmin} and V_{Rmax} , in p.u. Default is $[-6.0 \ 6.43]$.

Damping filter gain and time constant

The gain K_f and time constant T_f of the first-order system representing the derivative feedback. Default is $[0.001 \ 1]$.

Transient gain reduction lead and lag time constants

The time constants T_b , T_c , T_{b1} , and T_{c1} of the second-order system representing the lead-lag compensator. Default is $[0 \ 0 \ 0 \ 0]$.

Exciter and Rectifier Tab**Exciter output current limiter gain KLR**

The exciter output current limiter gain. Default is 1.

Exciter output current limit reference ILR

The exciter output current limit reference. Default is 2.

Rectifier loading factor Kc

The rectifier loading factor proportional to commutating reactance. Default is 0.038.

Initial Values Tab**Initial values of terminal voltage and field voltage**

The initial values of terminal voltage V_{t0} and field voltage E_{fd0} , both in p.u. Initial terminal voltage is normally set to 1 pu. The V_{t0} and E_{fd0} values can be determined using the Powergui Load Flow tool. Default is $[1 \ 1]$.

Sample time

Specify a value greater than zero to discretize the block at the given sample time. Set to -1 to inherit the simulation type and sample time parameters of the Powergui block. Default is 0.

Ports

V_{ref}

The reference value of the stator terminal voltage, in p.u.

V_t

The measured value in p.u. of the stator terminal voltage of the controlled Synchronous Machine block.

I_{fd}

The measured value in p.u. of the stator field current of the controlled Synchronous Machine block.

V_{stab}

Connect this input to a power system stabilizer to provide additional stabilization of power system oscillations. When you do not use this option, connect to a Simulink ground block. The input is in p.u.

E_{fd}

The field voltage to apply to the V_f input of the controlled Synchronous Machine block. The output is in p.u.

References

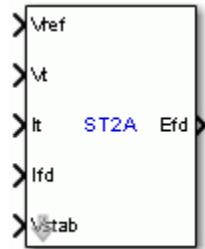
- [1] "IEEE Recommended Practice for Excitation System Models for Power System Stability Studies."
IEEE Standard, Vol. 421, No. 5, 2005 (Revision of IEEE 521.5-1992).

Version History

Introduced in R2012b

ST2A Excitation System

Implements IEEE type ST2A excitation system model



Library

Simscape / Electrical / Specialized Power Systems / Electrical Machines / Synchronous Machine Control

Description

This block models a compound source rectifier excitation system. It utilizes both terminal voltage, V_t , and terminal current, I_t , to form a model of the exciter power source.

This block is an adaptation of the ST2A excitation system of the IEEE 421 standard, copyright IEEE 2005, all rights reserved.

Parameters

Controllers Tab

Low-pass filter time constant

The time constant T_r of the first-order system representing the stator terminal voltage transducer. Default is $20e-3$.

Voltage regulator gain and time constant

The gain K_a and time constant T_a of the first-order system representing the main regulator. Default is $[120 \ 0.15]$.

Voltage regulator output limits

The voltage regulator output limits V_{Rmin} and V_{Rmax} , in p.u. Default is $[0 \ 1.0]$.

Damping filter gain and time constant

The gain K_f and time constant T_f of the first-order system representing the derivative feedback. Default is $[0.05 \ 1.0]$.

Exciter and Rectifier Tab

Exciter gain and time constant

The gain K_e and time constant T_e of the first-order system representing the exciter. Default is $[1.0 \ 0.5]$.

Current circuit gain coefficient KI

The current circuit gain coefficient. Default is 8.0.

Potential circuit gain coefficient Kp

The potential circuit gain coefficient. Default is 4.88.

Direct axis synchronous machine reactance

Enter the value of the Xd reactance of the controlled Synchronous Machine block. Default is 1.56.

Rectifier loading factor Kc

The rectifier loading factor, proportional to commutating reactance. Default is 1.82.

Initial Values Tab**Initial values of terminal voltage and field voltage**

The initial values of terminal voltage Vt0 and field voltage Efd0, both in p.u. Initial terminal voltage is normally set to 1 pu. The Vt0 and Efd0 values can be determined using the Powergui Load Flow tool. Default is [1 1].

Initial value of terminal current

The initial value of terminal current It0, in p.u. This value can be determined using the Powergui Load Flow tool. Default is 1.

Sample time

Specify a value greater than zero to discretize the block at the given sample time. Set to -1 to inherit the simulation type and sample time parameters of the Powergui block. Default is 0.

Ports

Vref

The reference value of the stator terminal voltage, in p.u.

Vt

The measured value in p.u. of the stator terminal voltage of the controlled Synchronous Machine block.

It

It The measured value in p.u. of the stator terminal current of the controlled Synchronous Machine block.

Ifd

The measured value in p.u. of the stator field current of the controlled Synchronous Machine block.

Vstab

Connect this input to a power system stabilizer to provide additional stabilization of power system oscillations. When you do not use this option, connect to a Simulink ground block. The input is in p.u.

Efd

The field voltage to apply to the Vf input of the controlled Synchronous Machine block. The output is in p.u.

References

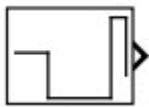
- [1] "IEEE Recommended Practice for Excitation System Models for Power System Stability Studies."
IEEE Standard, Vol. 421, No. 5, 2005 (Revision of IEEE 521.5-1992).

Version History

Introduced in R2012b

Stair Generator

Generate signal changing at specified transition times



Libraries:
Simscape / Electrical / Specialized Power Systems / Sources / Signal Generator
Sources

Description

The Stair Generator block generates a signal changing at specified transition times. Use the Stair Generator block to generate a logical signal to control the opening and closing of the Breaker block and the Ideal Switch block. You can also use the Stair Generator block to generate a signal whose amplitude changes by steps at specified transition times.

Characteristics

Sample Time	Specified in the Sample Time parameter Continuous if Sample Time = 0
Scalar Expansion	No
Dimensionalized	No
Zero-Crossing Detection	No

Examples

The power_SignalGenerators example shows the use of the Stair Generator block.

Parameters

Time (s) — Transition times
[0, 1, 3, 3.5] (default) | vector of nonnegative scalars

Transition times, in seconds, when the output changes to the values specified in the **Amplitude** parameter. The output is zero until the simulation reaches the first transition time.

Amplitude — Amplitude sequence of output signal
[1, -1, 2.5, 0] (default) | vector of scalars

Specify the amplitude sequence of the output signal. The amplitude is constant between the transition times specified in the **Time (s)** parameter. The amplitude of the output is 0 until the simulation reaches the first transition time.

Sample time — Sample time of block
0 (default) | nonnegative scalar

Specify the sample time of the block, in seconds. Set to 0 to implement a continuous block.

Version History

Introduced in R2013a

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

Sawtooth Generator | PWM Generator

Symbol	Meaning
V_2	Line to line voltage of source 2
X	Reactance of interconnection transformer and filters
δ	Phase angle of V_1 with respect to V_2

In steady state operation, the voltage, V_2 , which is generated by the VSC, is in phase with V_1 ($\delta=0$), so that only reactive power is flowing ($P=0$). If V_2 is lower than V_1 , Q is flowing from V_1 to V_2 (STATCOM is absorbing reactive power). On the reverse, if V_2 is higher than V_1 , Q is flowing from V_2 to V_1 (STATCOM is generating reactive power). The amount of reactive power is given by:

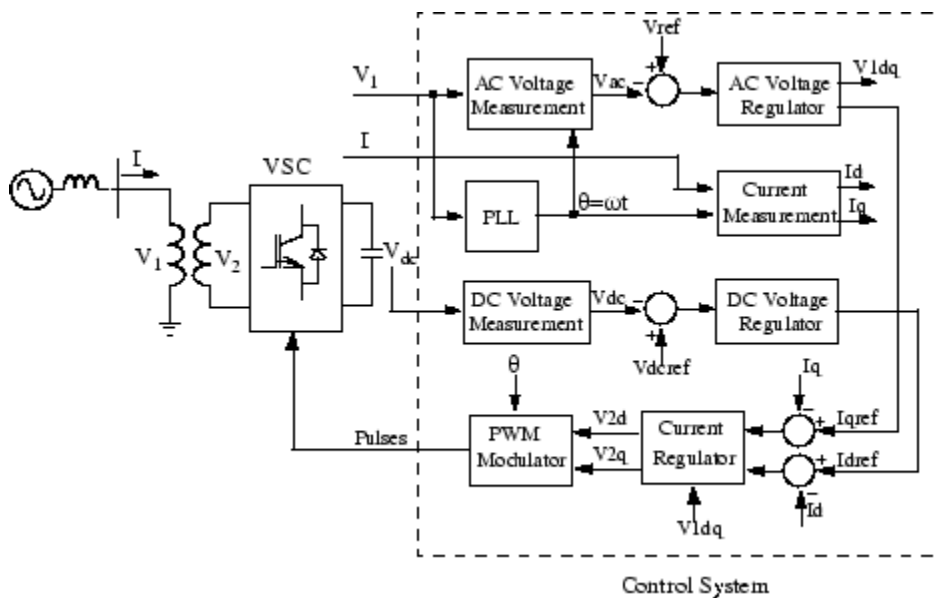
$$Q = (V_1 (V_1 - V_2)) / X.$$

A capacitor connected on the DC side of the VSC acts as a DC voltage source. In steady state, the voltage V_2 has to be phase shifted slightly behind V_1 in order to compensate for transformer and VSC losses and to keep the capacitor charged. Two VSC technologies can be used for the VSC:

- VSC using GTO-based square-wave inverters and special interconnection transformers. Typically four three-level inverters are used to build a 48-step voltage waveform. Special interconnection transformers are used to neutralize harmonics contained in the square waves generated by individual inverters. In this type of VSC, the fundamental component of voltage V_2 is proportional to the voltage V_{dc} . Therefore, V_{dc} has to be varied to control the reactive power.
- VSC using IGBT-based PWM inverters. This type of inverter uses a Pulse-Width Modulation (PWM) technique to synthesize a sinusoidal waveform from a DC voltage source with a typical chopping frequency of a few kilohertz. Harmonic voltages are cancelled by connecting filters at the AC side of the VSC. This type of VSC uses a fixed DC voltage V_{dc} . The voltage V_2 is varied by changing the modulation index of the PWM modulator.

The Static Synchronous Compensator (Phasor Type) block models an IGBT-based STATCOM (fixed DC voltage). However, as details of the inverter and harmonics are not represented, it can be also used to model a GTO-based STATCOM in transient stability studies. A detailed model of a GTO-based STATCOM is found in the "STATCOM" example.

The figure below shows a single-line diagram of the STATCOM and a simplified block diagram of its control system.



Single-line Diagram of a STATCOM and Its Control System Block Diagram

The control system consists of:

- A phase-locked loop (PLL) that synchronizes on the positive-sequence component of the three-phase primary voltage, V_1 . The output of the PLL (angle $\theta = \omega t$) is used to compute the direct-axis and quadrature-axis components of the AC three-phase voltage and currents (labeled as V_d , V_q or I_d , and I_q on the diagram).
- Measurement systems measuring the d and q components of the AC positive-sequence voltage, currents to be controlled, and the DC voltage V_{dc} .
- An outer regulation loop consisting of an AC voltage regulator and a DC voltage regulator. The output of the AC voltage regulator is the reference current I_{qref} for the current regulator, where I_q is the current in quadrature with voltage that controls reactive power flow. The output of the DC voltage regulator is the reference current I_{dref} for the current regulator, where I_d is the current in phase with voltage that controls active power flow.
- An inner current regulation loop consisting of a current regulator. The current regulator controls the magnitude and phase of the voltage generated by the PWM converter (V_{2d} and V_{2q}) from the I_{dref} and I_{qref} reference currents produced respectively by the DC voltage regulator and the AC voltage regulator (in voltage control mode). The current regulator is assisted by a feed-forward-type regulator that predicts the V2 voltage output (V_{2d} and V_{2q}) from the V_1 measurement (V_{1d} and V_{1q}) and the transformer leakage reactance.

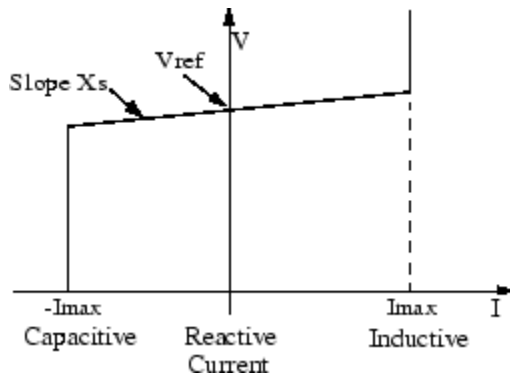
The STATCOM block is a phasor model that does not include detailed representations of the power electronics. You must use it with the phasor simulation method, activated by setting the **Simulation type** parameter of a Powergui block to **Phasor**. It can be used in three-phase power systems together with synchronous generators, motors, dynamic loads, and other FACTS and Renewable Energy systems to perform transient stability studies and observe impact of the STATCOM on electromechanical oscillations and transmission capacity at fundamental frequency.

STATCOM V-I Characteristic

The STATCOM can be operated in two different modes:

- In voltage regulation mode (the voltage is regulated within limits as explained below)
- In var control mode (the STATCOM reactive power output is kept constant)

When the STATCOM is operated in voltage regulation mode, it implements the following V-I characteristic:



STATCOM V-I characteristic

As long as the reactive current stays within the minimum and maximum current values ($-I_{max}$ and I_{max}) imposed by the converter rating, the voltage is regulated at the reference voltage V_{ref} . However, a voltage droop is normally used (usually between 1% and 4% at maximum reactive power output), and the V-I characteristic has the slope indicated in the figure. In the voltage regulation mode, the V-I characteristic is described by the following equation:

$$V = V_{ref} + X_s I$$

where,

V	Positive sequence voltage (pu)
I	Reactive current (pu/Pnom) ($I > 0$ indicates an inductive current)
$X_s I$	Slope or droop reactance (pu/Pnom)
P_{nom}	Three-phase nominal power of the converter specified in the block dialog box

Differences Between STATCOM and SVC

The STATCOM performs the same function as the SVC. However, at voltages lower than the normal voltage regulation range, the STATCOM can generate more reactive power than the SVC. This is due to the fact that the maximum capacitive power generated by an SVC is proportional to the square of the system voltage (constant susceptance) while the maximum capacitive power generated by a STATCOM decreases linearly with voltage (constant current). This ability to provide more capacitive reactive power during a fault is one important advantage of the STATCOM over the SVC. In addition, the STATCOM normally exhibits a faster response than the SVC because with the VSC, the STATCOM has no delay associated with the thyristor firing (in the order of 4 ms for a SVC).

Ports

Input

Trip — Trip signal to disable STATCOM
logical

Trip signal to disable the STATCOM, specified as a logical. When this input is high, the STATCOM is disconnected and its control system is disabled. Use this input to implement a simplified version of the protection system.

Vref — External reference voltage signal
nonnegative scalar

Simulink input of the external reference voltage signal.

Dependencies

To enable this port, on the **Controller** tab, select **External**.

Output

m — Measurement
16-element vector

Measurements, returned as a 16-element vector of STATCOM internal signals. These signals are either voltage and current phasors (complex signals) or control signals. They can be individually accessed by using the Bus Selector block. They are, in order:

Signal	Signal Group	Signal Names	Definition
1-3	Power Vabc (cmplx)	Va_prim (pu) Vb_prim (pu) Vc_prim (pu)	Phasor voltages (phase to ground) Va, Vb, and Vc at the STATCOM primary terminals (pu)
4-6	Power Iabc (cmplx)	Ia_prim (pu) Ib_prim (pu) Ic_prim (pu)	Phasor currents Ia, Ib, and Ic flowing into the STATCOM (pu)
7	Power	Vdc (V)	DC voltage (V)
8	Control	Vm (pu)	Positive-sequence value of the measured voltage (pu)
9	Control	Vref (pu)	Reference voltage (pu)
10	Control	Qm (pu)	STATCOM reactive power. A positive value indicates inductive operation.
11	Control	Qref (pu)	Reference reactive power (pu)
12	Control	Id (pu)	Direct-axis component of current (active current) flowing into STATCOM (pu). A positive value indicates active power flowing into STATCOM.
13	Control	Iq (pu)	Quadrature-axis component of current (reactive current) flowing into STATCOM (pu). A positive value indicates capacitive operation.
14	Control	Idref (pu)	Reference value of direct-axis component of current flowing into STATCOM (pu)
15	Control	Iqref (pu)	Reference value of quadrature-axis component of current flowing into STATCOM (pu)

Signal	Signal Group	Signal Names	Definition
16	Control	modindex	The modulation index, m , of the PWM modulator. A positive number such that $0 < m < 1$. $m=1$ corresponds to the maximum voltage V_2 that can be generated by the VSC without overmodulation.

Conserving

A — Electric terminal of phase A
specialized electrical

Specialized electrical conserving port associated with the electrical terminal of phase A.

B — Electric terminal of phase B
specialized electrical

Specialized electrical conserving port associated with the electrical terminal of phase B.

C — Electric terminal of phase C
specialized electrical

Specialized electrical conserving port associated with the electrical terminal of phase C.

Parameters

Power

Nominal voltage and frequency [Vrms L-L, fn(Hz)] — Nominal line-to-line voltage and the nominal system frequency
[500e3, 60] (default) | 2-element vector

The nominal line-to-line voltage in Vrms and the nominal system frequency in hertz.

Converter rating (VA) — Nominal power of converter
100e6 (default) | positive scalar

The nominal power of the converter in VA.

Converter impedance [R(pu), L(pu)] — Positive-sequence resistance and inductance
[0.22/30, 0.22] (default) | 2-element vector

The positive-sequence resistance and inductance of the converter, in pu, based on the nominal power and voltage ratings. R and L represent the resistance and leakage inductance of the coupling transformer and the resistance and inductance of the series filtering inductors connected at the VSC output.

Converter initial current [Mag(pu), Pha(deg.)] — Initial value of positive-sequence current phasor
[0, 0] (default) | 2-element vector

The initial value of the positive-sequence current phasor (with the magnitude in pu and phase in degrees). If you know the initial value of the current corresponding to the STATCOM operating point,

you can specify it to start simulation in steady state. If you don't know this value, you can leave [0 0]. The system will reach steady-state after a short transient.

DC link nominal voltage (V) — Nominal voltage of DC link

40000 (default) | positive scalar

The nominal voltage of the DC link in volts.

DC link total equivalent capacitance (F) — Total capacitance of DC link

750e-6/2 (default) | positive scalar

The total capacitance of the DC link in farads. This capacitance value is related to the STATCOM rating and to the DC link nominal voltage. The energy stored in the capacitance (in joules) divided by the STATCOM rating (in VA) is a time duration that is usually a fraction of a cycle at nominal frequency. For example, for the default parameters, (where $C=375\text{ }\mu\text{F}$, $V_{dc}=40\text{ }000\text{ V}$, and $S_{nom}=100\text{ MVA}$) this ratio, $(C \cdot V_{dc}^2/2)/S_{nom}$, is 3.0 ms, which represents 0.18 cycle for a 60 Hz frequency. If you change the default values of the nominal power rating and DC voltage, you should change the capacitance value accordingly.

Controller

Mode — STATCOM mode of operation

Voltage regulation (default) | Var Control

Specifies the STATCOM mode of operation.

Reference voltage Vref (pu) — Reference voltage of STATCOM

1.00 (default) | positive scalar

Reference voltage, in pu, used by the voltage regulator.

Dependencies

To enable this parameter, set **Mode** to Voltage regulation and clear the **External** check box.

External — Control reference voltage from external signal

off (default) | on

When **External** is selected, a Simulink input port named Vref appears on the block, allowing you to control the reference voltage from an external signal (in pu). The **Reference voltage Vref** parameter is therefore disabled.

Maximum rate of change of reference voltage (pu/s) — Maximum rate of change when external reference voltage is used

10 (default) | positive scalar

Maximum rate of change of the reference voltage, in pu/s, when an external reference voltage is used.

Dependencies

To enable this parameter, set **Mode** to Voltage regulation.

Droop (pu) — Droop reactance

0.03 (default) | positive scalar

Droop reactance, in pu, which defines the slope of the V-I characteristic.

Dependencies

To enable this parameter, set **Mode** to **Voltage regulation**.

Vac Regulator gains [Kp, Ki] — Gains of AC voltage PI regulator
[5, 1000] (default) | 2-element vector

Gains of the AC voltage PI regulator. Specify the proportional gain Kp, which is calculated by (pu of I)/(pu of V). Specify the integral gain, Ki, which is calculated as (pu of I)/(pu of V)/s, where V is the AC voltage error and I is the output of the voltage regulator.

Dependencies

To enable this parameter, set **Mode** to **Voltage regulation**.

Reactive power setpoint Qref (pu) — Reference reactive power
0 (default) | positive scalar

Reference reactive power, in pu, when the STATCOM is in **Var Control** mode.

Dependencies

To enable this parameter, set **Mode** to **Var Control**.

Maximum rate of change of reactive power setpoint (pu/s) — Maximum rate of change of reference reactive power
2 (default) | positive scalar

Maximum rate of change of the reference reactive power, in pu/s.

Dependencies

To enable this parameter, set **Mode** to **Var Control**.

Vdc Regulator gains [Kp, Ki] — Gains of DC voltage PI regulator
[0.1e-3, 20e-3] (default) | 2-element vector

Gains of the DC voltage PI regulator, which controls the voltage across the DC bus capacitor. Specify the proportional gain, Kp, which is calculated by (pu of I)/Vdc. The integral gain, Ki, is calculated as (pu of I)/Vdc/s, where Vdc is the DC voltage error and I is the output of the voltage regulator.

Current Regulator gains [Kp, Ki, Kf] — Gains of inner current regulation loop
[0.3, 10, 0.22] (default) | 3-element vector

Gains of the inner current regulation loop. Specify the proportional gain, Kp, which is calculated as (pu of V)/(pu of I). The integral gain, Ki, is calculated as (pu of V)/(pu of I)/s. The feed forward gain, Kf, is calculated as (pu of V)/(pu of I), where V is the output V2d or V2q of the current regulator and I is the Id or Iq current error.

For optimal performance, the feed forward gain should be set to the converter reactance (in pu) given by parameter L in the **Converter impedance [R(pu),L(pu)]** parameter.

Version History

Introduced in R2006a

References

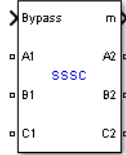
- [1] N. G. Hingorani, L. Gyugyi, "Understanding FACTS; Concepts and Technology of Flexible AC Transmission Systems," *IEEE Press book* , 2000.

See Also

Static Var Compensator (Phasor Type) | Unified Power Flow Controller (Phasor Type)

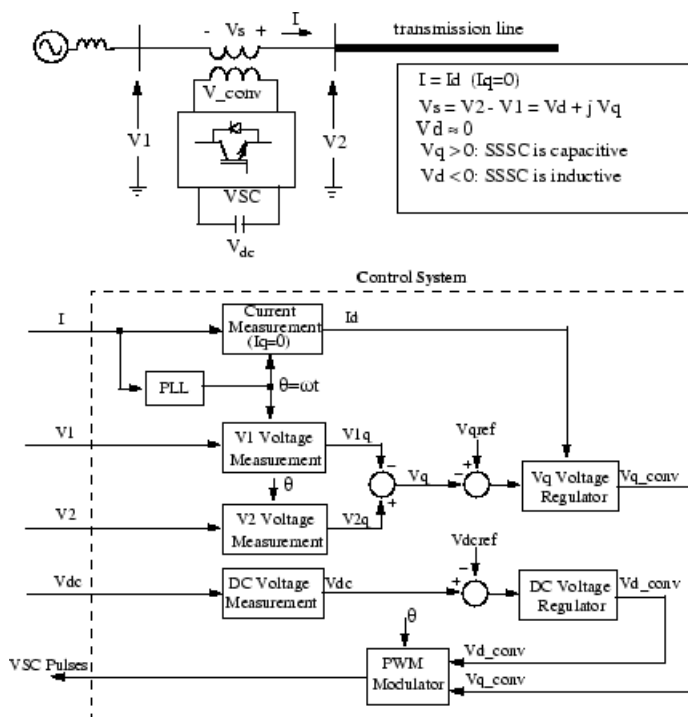
Static Synchronous Series Compensator (Phasor Type)

Implement phasor model of three-phase static synchronous series compensator



Description

The Static Synchronous Series Compensator (SSSC) is a series device of the Flexible AC Transmission Systems (FACTS) family using power electronics to control power flow and improve power oscillation damping on power grids [1]. The SSSC injects a voltage V_s in series with the transmission line where it is connected.



Single-line Diagram of a SSSC and Its Control System Block Diagram

As the SSSC does not use any active power source, the injected voltage must stay in quadrature with line current. By varying the magnitude V_q of the injected voltage in quadrature with current, the SSSC performs the function of a variable reactance compensator, either capacitive or inductive.

The variation of injected voltage is performed by means of a Voltage-Sourced Converter (VSC) connected on the secondary side of a coupling transformer. The VSC uses forced-commutated power electronic devices (GTOs, IGBTs or IGCTs) to synthesize a voltage V_{conv} from a DC voltage source.

A capacitor connected on the DC side of the VSC acts as a DC voltage source. A small active power is drawn from the line to keep the capacitor charged and to provide transformer and VSC losses, so that the injected voltage V_s is practically 90 degrees out of phase with current I . In the control system block diagram V_d_{conv} and V_q_{conv} designate the components of converter voltage V_{conv} which are respectively in phase and in quadrature with current. Two VSC technologies can be used for the VSC:

- VSC using GTO-based square-wave inverters and special interconnection transformers. Typically four three-level inverters are used to build a 48-step voltage waveform. Special interconnection transformers are used to neutralize harmonics contained in the square waves generated by individual inverters. In this type of VSC, the fundamental component of voltage V_{conv} is proportional to the voltage V_{dc} . Therefore V_{dc} has to vary for controlling the injected voltage.
- VSC using IGBT-based PWM inverters. This type of inverter uses Pulse-Width Modulation (PWM) technique to synthesize a sinusoidal waveform from a DC voltage with a typical chopping frequency of a few kilohertz. Harmonics are cancelled by connecting filters at the AC side of the VSC. This type of VSC uses a fixed DC voltage V_{dc} . Voltage V_{conv} is varied by changing the modulation index of the PWM modulator.

The SSSC (Phasor Type) block models an IGBT-based SSSC (fixed DC voltage). However, as details of the inverter and harmonics are not represented, it can be also used to model a GTO-based SSSC in transient stability studies.

The control system consists of:

- A phase-locked loop (PLL) which synchronizes on the positive-sequence component of the current I . The output of the PLL (angle $\Theta = \omega t$) is used to compute the direct-axis and quadrature-axis components of the AC three-phase voltages and currents (labeled as V_d , V_q or I_d , I_q on the diagram).
- Measurement systems measuring the q components of AC positive-sequence of voltages V_1 and V_2 (V_{1q} and V_{2q}) as well as the DC voltage V_{dc} .
- AC and DC voltage regulators which compute the two components of the converter voltage ($V_{d_{\text{conv}}}$ and $V_{q_{\text{conv}}}$) required to obtain the desired DC voltage (V_{dref}) and the injected voltage (V_{qref}). The V_q voltage regulator is assisted by a feed forward type regulator which predicts the V_{conv} voltage from the I_d current measurement.

The SSSC block is a phasor model which does not include detailed representations of the power electronics. You must use it with the phasor simulation method, activated with the Powergui block. It can be used in three-phase power systems together with synchronous generators, motors, dynamic loads, and other FACTS and Renewable Energy systems to perform transient stability studies and observe impact of the SSSC on electromechanical oscillations and transmission capacity at fundamental frequency.

Parameters

The SSSC parameters are grouped in two categories: **Power data** and **Control parameters**. Use the **Display** listbox to select which group of parameters you want to visualize.

Power Tab

Nominal voltage and frequency

The nominal line-to-line voltage in Vrms and the nominal system frequency in hertz. Default is [500e3, 60].

Series Converter rating

The nominal rating of the series converter in VA and the maximum value of the injected voltage V_{conv} on the VSC side of the transformer (see single line diagram), in pu of nominal phase-to-ground voltage. Default is [100e6, 0.1].

Series Converter impedance

The positive-sequence resistance and inductance of the converter, in pu based on the nominal converter rating and nominal voltage. R and L represent the resistance and leakage inductance of the coupling transformer plus the resistance and inductance of the series filtering inductors connected at the VSC output. Default is [0.16/30, 0.16].

Series Converter initial current

The initial value of the positive-sequence current phasor (Magnitude in pu and Phase in degrees). If you know the initial value of the current corresponding to the SSSC operating point you may specify it in order to start simulation in steady state. If you don't know this value, you can leave [0 0]. The system will reach steady-state after a short transient. Default is [0, 0].

DC link nominal voltage

The nominal voltage of the DC link in volts. Default is 40000.

DC link total equivalent capacitance

The total capacitance of the DC link in farads. This capacitance value is related to the SSSC converter rating and to the DC link nominal voltage. The energy stored in the capacitance (in joules) divided by the converter rating (in VA) is a time duration which is usually a fraction of a cycle at nominal frequency. For example, for the default parameters, ($C=375\ \mu\text{F}$, $V_{dc}=40\ 000\ \text{V}$, $S_{nom}=100\ \text{MVA}$) this ratio is 3.0 ms, which represents 0.18 cycle for a 60 Hz frequency. If you change the default values of the nominal power rating and DC voltage, you should change the capacitance value accordingly. Default is 375e-6 .

Controller Tab**Bypass Breaker**

Specifies the status of the bypass breaker connected inside the block across terminals A1, B1, C1 and A2, B2, C2. Select either **External Control** (default), **Open** or **Closed**. If the bypass breaker is in external control, a Simulink input named Bypass appears on the block, allowing to control the status of the bypass breaker from an external signal (0 or 1).

Injected voltage reference Vqref

Specify the quadrature-axis component of the voltage injected on the VSC side of the series transformer, in pu. Default is 0.05.

External

When **External** is selected, a Simulink input named Vqef appears on the block, allowing you to control the reference voltage from an external signal (in pu). The **Injected voltage reference Vqref** parameter is therefore unavailable. Default is cleared.

Maximum rate of change for Vqref

Maximum rate of change of the Vqref voltage, in pu/s. Default is 3.

Injected voltage regulator gains: [Kp Ki]

Gains of the PI regulator which controls the injected voltage. Specify proportional gain Kp in (pu of V_{q_conv})/(pu of V), and integral gain Ki, in (pu of V_{q_conv})/(pu of V)/s, where V is the Vq

voltage error and V_{q_conv} is the quadrature-axis component of the voltage generated by the VSC. Default is $[0.03, 1.5]/8$.

The feed forward gain is computed from the **Series Converter impedance** parameters.

Vdc regulator gains: [Kp Ki]

Gains of the DC voltage PI regulator which controls the voltage across the DC bus capacitor. Specify proportional gain Kp in (pu of V_{d_conv})/Vdc, and integral gain Ki, in (pu of V_{d_conv})/Vdc/s, where Vdc is the DC voltage error and V_{d_conv} is the direct-axis component of the voltage generated by the converter. Default is $[0.1e-3, 20e-3]$.

Inputs and Outputs

A1 B1 C1

The three input terminals of the SSSC.

A2 B2 C2

The three output terminals of the SSSC.

Bypass

This input is visible only when the **Bypass Breaker** parameter is set to External Control.

Apply a Simulink logical signal (0 or 1) to this input. When this input is high the bypass breaker is closed.

Vqref

This input is visible only when the **External control of injected voltage Vqref** parameter is checked.

Apply a Simulink signal specifying the reference voltage, in pu.

m

Simulink output vector containing 17 SSSC internal signals. These signals are either voltage and current phasors (complex signals) or control signals. They can be individually accessed by using the Bus Selector block. They are, in order:

Signal	Signal Group	Signal Names	Definition
1-3	Power Vabc1 (cmplx)	Va1 (pu) Vb1 (pu) Vc1 (pu)	Phasor voltages (phase to ground) Va, Vb, Vc at the SSSC input terminals A1, B1, C1 (pu)
4	Power	Vdc (V)	DC voltage (V)
5-7	Power Vabc2 (cmplx)	Va2 (pu) Vb2 (pu) Vc2 (pu)	Phasor voltages (phase to ground) Va, Vb, Vc at the SSSC output terminals A2, B2, C2 (pu)
8-10	Power Vabc_Inj (cmplx)	Va_Inj (pu) Vb_Inj (pu) Vc_Inj (pu)	Phasors of injected voltages $V_s = V_2 - V_1$ (pu)
11-13	Power Iabc (cmplx)	Ia (pu) Ib (pu) Ic (pu)	Phasor currents Ia, Ib, Ic flowing out of terminals A2, B2, C2 (pu)

Signal	Signal Group	Signal Names	Definition
14	Control	Vqref (pu)	Reference value of quadrature-axis injected voltage (pu)
15	Control	Vqinj (pu)	Measured injected voltage in quadrature-axis (pu)
16	Control	Id(pu)	Measured current (pu)
17	Control	modindex	The modulation index m of the PWM modulator. A positive number $0 < m < 1$. $m=1$ corresponds to the maximum voltage V_{conv} which can be generated by the series converter without overmodulation.

Examples

See the “Static Synchronous Series Compensator” example which illustrates the use of a SSSC for damping power oscillations on a 500 kV, 60 Hz, system.

References

- [1] N. G. Hingorani, L. Gyugyi, “Understanding FACTS; Concepts and Technology of Flexible AC Transmission Systems,” *IEEE Press book*, 2000

See Also

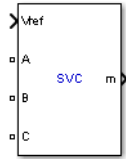
Unified Power Flow Controller (Phasor Type)

Version History

Introduced in R2006a

Static Var Compensator (Phasor Type)

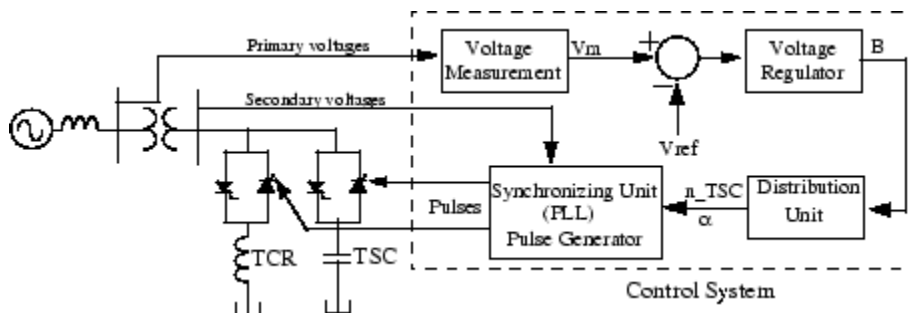
Implement phasor model of three-phase static var compensator



Description

The Static Var Compensator (SVC) is a shunt device of the Flexible AC Transmission Systems (FACTS) family using power electronics to control power flow and improve transient stability on power grids [1]. The SVC regulates voltage at its terminals by controlling the amount of reactive power injected into or absorbed from the power system. When system voltage is low, the SVC generates reactive power (SVC capacitive). When system voltage is high, it absorbs reactive power (SVC inductive). The variation of reactive power is performed by switching three-phase capacitor banks and inductor banks connected on the secondary side of a coupling transformer. Each capacitor bank is switched on and off by three thyristor switches (Thyristor Switched Capacitor or TSC). Reactors are either switched on-off (Thyristor Switched Reactor or TSR) or phase-controlled (Thyristor Controlled Reactor or TCR).

The figure below shows a single-line diagram of a static var compensator and a simplified block diagram of its control system.



Single-line Diagram of an SVC and Its Control System Block Diagram

The control system consists of

- A measurement system measuring the positive-sequence voltage to be controlled. A Fourier-based measurement system using a one-cycle running average is used.
- A voltage regulator that uses the voltage error (difference between the measured voltage V_m and the reference voltage V_{ref}) to determine the SVC susceptance B needed to keep the system voltage constant
- A distribution unit that determines the TSCs (and eventually TSRs) that must be switched in and out, and computes the firing angle α of TCRs
- A synchronizing system using a phase-locked loop (PLL) synchronized on the secondary voltages and a pulse generator that send appropriate pulses to the thyristors

The SVC (Phasor Type) block is a phasor model, and you must use it with the phasor simulation method, activated with the Powergui block. It can be used in three-phase power systems together with synchronous generators, motors, and dynamic loads to perform transient stability studies and observe impact of the SVC on electromechanical oscillations and transmission capacity. This model does not include detailed representations of the power electronics, the measurement system, or the synchronization system. These systems are approximated rather by simple transfer functions that yield a correct representation at the system's fundamental frequency.

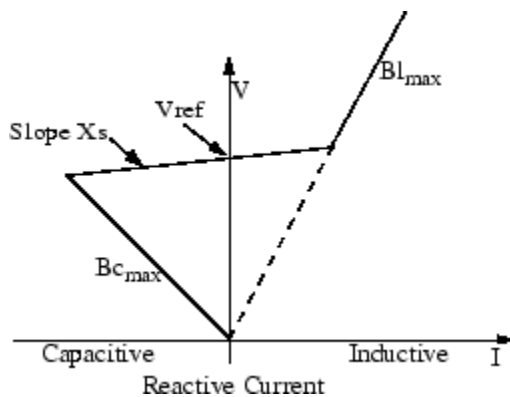
A detailed model of a SVC using three TSCs and one TCR is provided in the “Static Var Compensator (SVC)” example.

SVC V-I Characteristic

The SVC can be operated in two different modes:

- In voltage regulation mode (the voltage is regulated within limits as explained below)
- In var control mode (the SVC susceptance is kept constant)

When the SVC is operated in voltage regulation mode, it implements the following V-I characteristic.



SVC V-I characteristic

As long as the SVC susceptance B stays within the maximum and minimum susceptance values imposed by the total reactive power of capacitor banks ($B_{c_{max}}$) and reactor banks ($B_{l_{max}}$), the voltage is regulated at the reference voltage V_{ref} . However, a voltage droop is normally used (usually between 1% and 4% at maximum reactive power output), and the V-I characteristic has the slope indicated in the figure. The V-I characteristic is described by the following three equations:

$$V = \begin{cases} V_{ref} + X_s \cdot I & \text{if SVC is in regulation range } (-B_{c_{max}} < B < B_{l_{max}}) \\ -\frac{I}{B_{c_{max}}} & \text{if SVC is fully capacitive } (B = B_{c_{max}}) \\ \frac{I}{B_{l_{max}}} & \text{if SVC is fully inductive } (B = B_{l_{max}}), \end{cases}$$

where

V	Positive sequence voltage (pu)
I	Reactive current (pu/Phase) ($I > 0$ indicates an inductive current)

Xs	Slope or droop reactance (pu/Pbase)
Bcmax	Maximum capacitive susceptance (pu/Pbase) with all TSCs in service, no TSR or TCR
Blmax	Maximum inductive susceptance (pu/Pbase) with all TSRs in service or TCRs at full conduction, no TSC
Pbase	Three-phase base power specified in the block dialog box

SVC Dynamic Response

When the SVC is operating in voltage regulation mode, its response speed to a change of system voltage depends on the voltage regulator gains (proportional gain Kp and integral gain Ki), the droop reactance Xs, and the system strength (short-circuit level).

For an integral-type voltage regulator (Kp = 0), if the voltage measurement time constant Tm and the average time delay Td due to valve firing are neglected, the closed-loop system consisting of the SVC and the power system can be approximated by a first-order system having the following closed-loop time constant:

$$T_c = 1 / (Ki(Xs + Xn))$$

where

T _c	Closed loop time constant
Ki	Proportional gain of the voltage regulator (pu_B/pu_V/s)
Xs	Slope reactance pu/Pbase
Xn	Equivalent power system reactance (pu/Pbase)

This equation demonstrates that you obtain a faster response speed when the regulator gain is increased or when the system short-circuit level decreases (higher Xn values). If you take into account the time delays due to voltage measurement system and valve firing, you obtain an oscillatory response and, eventually, an instability with too weak a system or too large a regulator gain.

Parameters

The SVC parameters are grouped in two categories: **Power Data** and **Control Parameters**. Use the **Display** listbox to select which group of parameters you want to visualize.

Power Tab

Ignore negative-sequence current

The SVC is modeled by a three-wire system using two current sources. The SVC does not generate any zero-sequence current, but it can generate negative-sequence currents during unbalanced system operation. The negative-sequence susceptance of the SVC is assumed to be identical to its positive-sequence value, as determined by the B value computed by the voltage regulator.

Select this box to ignore negative-sequence current. Default is selected.

Nominal voltage and frequency

The nominal line-to-line voltage in Vrms and the nominal system frequency in hertz. Default is [500e3, 60].

Three-phase base power

Three-phase base power, in VA, used to specify the following parameters in pu: droop reactance X_s , gains K_p and K_i of the voltage PI regulator, and reference susceptance B_{ref} . This base power is also used to normalize the output B susceptance signal. Default is $200e6$.

Reactive power limits

The maximum SVC reactive powers at 1 pu voltage, in vars. Enter a positive value for the capacitive reactive power Q_c (vars generated by the SVC) and a negative value for the inductive reactive power Q_l (vars absorbed by the SVC). Default is $[200e6, -200e6]$.

Average time delay due to thyristor valves firing

Average time delay simulating the non instantaneous variation of thyristor fundamental current when the distribution unit sends a switching order to the pulse generator. Because pulses have to be synchronized with thyristor commutation voltages, this delay normally varies between 0 and 1/2 cycle. The suggested average value is 4 ms. Default is $4e-3$.

Controller Tab**Mode**

Specifies the SVC mode of operation. Select either **Voltage regulation** (default) or **Var control (Fixed susceptance B_{ref})**.

Reference voltage V_{ref}

This parameter is not available when the **Mode** parameter is set to **Var control (Fixed susceptance B_{ref})** or when **External** is selected.

Reference voltage, in pu, used by the voltage regulator. Default is 1.0 .

External

When **External** is selected, a Simulink input named V_{ref} appears on the block, allowing you to control the reference voltage from an external signal (in pu). The **Reference voltage** parameter is therefore unavailable. Default is selected.

Droop X_s

This parameter is not available when the **Mode** parameter is set to **Var control (Fixed susceptance B_{ref})**.

Droop reactance, in pu/Pbase, defining the slope of the V-I characteristic. Default is 0.03 .

Voltage regulator gains

This parameter is not available when the **Mode** parameter is set to **Var control (Fixed susceptance B_{ref})**.

Proportional gain, in $(pu \text{ of } B)/(pu \text{ of } V)$, and integral gain, in $pu\_B/pu\_V/s$, of the voltage regulator. Default is $[0 \ 300]$.

Susceptance B_{ref}

This parameter is not available when the **Mode** parameter is set to **Voltage regulation**.

Reference susceptance, in pu/Pbase, when the SVC is operating in var control mode. Default is 0.0 .

Inputs and Outputs

A B C

The three terminals of the SVC.

Vref

Simulink input of the reference voltage signal.

This input is visible only when the **External control of reference voltage Vref** parameter is checked.

m

Simulink output vector containing six SVC internal signals. These signals are either voltage and current phasors (complex signals) or control signals. They can be individually accessed by using the Bus Selector block. They are, in order:

Signal	Signal Group	Signal Names	Definition
1-3	Power Iabc (cmplx)	Ia(pu) Ib(pu) Ic(pu)	Phasor currents Ia, Ib, Ic flowing into the SVC (pu)
4	Control	Vm (pu)	Positive-sequence value of measured voltage (pu)
5	Control	B (pu)	SVC susceptance output of the voltage regulator (pu). A positive value indicates that the SVC is capacitive.
6	Control	Q (pu)	SVC reactive power output (pu). A positive value indicates inductive operation.

Examples

The “Static Var Compensator (SVC) Phasor Model” example illustrates a static var compensator (SVC) used to regulate voltage on a 500 kV, 3000 MVA system. When the system voltage is low, the SVC generates reactive power (SVC capacitive) and when the system voltage is high it absorbs reactive power (SVC inductive).

References

[1] N. G. Hingorani, L. Gyugyi, “Understanding FACTS; Concepts and Technology of Flexible AC Transmission Systems,” *IEEE Press book*, 2000

See Also

Static Synchronous Compensator (Phasor Type)

Version History

Introduced in R2006a

Steam Turbine and Governor

Model the dynamics of speed governing system, steam turbine, and multimass shaft

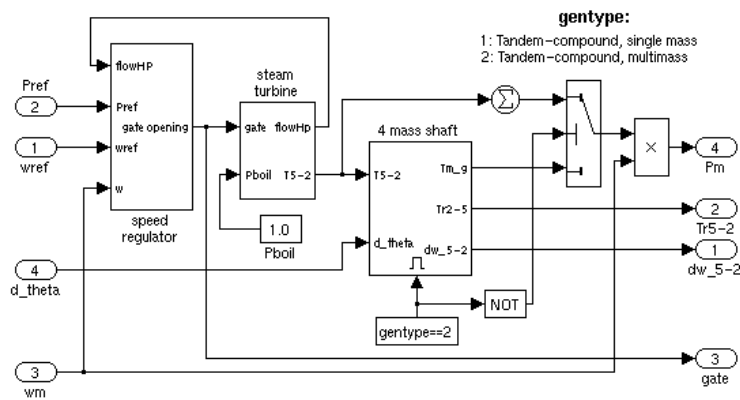


Libraries:

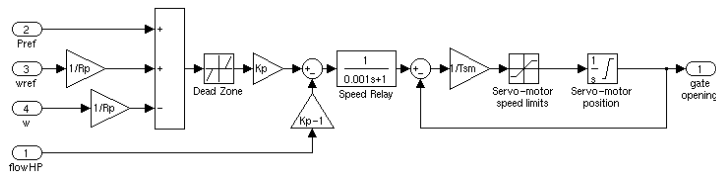
Simscape / Electrical / Specialized Power Systems / Electrical Machines / Synchronous Machine Control

Description

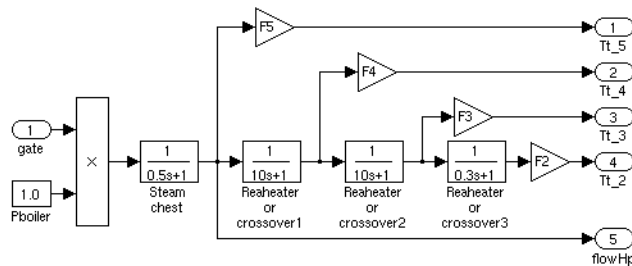
The Steam Turbine and Governor block implements a complete tandem-compound steam prime mover, including a speed governing system, a four-stage steam turbine, and a shaft with up to four masses.



The speed governing system consists of a proportional regulator, a speed relay, and a servomotor controlling the gate opening. It is similar to one of the models proposed in [1].



The steam turbine has four stages, each modeled by a first-order transfer function. The first stage represents the steam chest while the three other stages represent either reheaters or crossover piping. The boiler is not modeled and boiler pressure is constant at 1.0 pu. Fractions F2 to F5 are used to distribute the turbine power to the various shaft stages:



The shaft models a four-mass system, which is coupled to the mass in the Synchronous Machine model for a total of five masses. The machine's mass is labeled mass #1. The mass in the Steam Turbine and Governor block, which is closest to the machine's mass, is mass #2, while the mass farthest from the machine is mass #5. The shaft is characterized by mass inertias H , damping factors D , and rigidity coefficients K . If you choose to simulate a single-mass shaft, the entire four-mass shaft subsystem in the Steam Turbine and Governor block is disabled and all the torque from the turbine is added together and applied to the machine's mass.

Ports

Input

wref — Reference speed
scalar

Reference speed, in pu. It is normally connected to a Constant block with the value set to 1.0 pu.

Pref — Reference electrical power
scalar

Reference electrical power in pu. It is set to a constant value corresponding to the initial active power drawn from the Synchronous Machine block connected to the Steam Turbine and Governor block.

wm — Generator speed
scalar

Generator speed, in pu. This is one of the signals in the last output of the Synchronous Machine model (internal variables).

d_theta — Generator power angle deviation
scalar

Generator power angle deviation. It is also one of the signals in the last output of the Synchronous Machine model (internal variables).

Output

dw_5-2 — Speed deviations of masses
vector

Speed deviations, in pu, of masses 5, 4, 3, and 2.

Tr5-2 — Torques transmitted by masses
vector

Torques, in pu, transmitted by masses 5, 4, 3, and 2.

gate — Gate opening
scalar

Gate opening, in pu.

Pm — Mechanical power
scalar

Mechanical power Pm for the Synchronous Machine block, in pu.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Generator type — Rotor type
Tandem-compound (multi-mass) (default) | Tandem-compound (single mass)

Rotor type: single mass or multimass tandem-compound. If you choose a single-mass system, the multimass shaft subsystem in the Steam Turbine and Governor block is disabled and the turbine's output torques are summed together and applied to the single mass in the Synchronous Machine block.

Regulator gain, perm. droop, dead zone [Kp Rp(pu) Dz(pu)] — Gain, permanent droop, and dead-zone width
[1 0.05 0] (default) | vector of three nonnegative elements

Gain Kp, permanent droop Rp (pu), and dead-zone width Dz (pu). Set gain to 3 if you want to use the steam flow feedback loop. Otherwise, set gain to 1.

Speed relay and servo-motor time constants [Tsr Tsm] (s) — Speed relay and gate servomotor time constants
[0.001 0.15] (default) | vector of two nonnegative elements

Speed relay and gate servomotor time constants Tsr (s) and Tsm (s).

Gate opening limits [vgmin,vgmax (pu/s) gmin,gmax (pu)] — Gate opening limits
[-0.1 0.1 0 4.496] (default) | vector of four elements

Minimum and maximum gate opening speed vgmin and vgmax (both in pu/s), and minimum and maximum gate opening gmin and gmax (both in pu).

Nominal speed of synchronous machine (rpm) — Synchronous speed of generator
3600 (default) | nonnegative scalar

Synchronous speed of the generator driven by the steam turbine (rpm).

Steam turbine time constants [T2 T3 T4 T5] (s) — Turbine time constants
[0 10 3.3 0.5] (default) | vector of four nonnegative elements

Turbine time constants T2 to T5 (s). Numbered consistently with turbine torque fractions and mass numbers; i.e., T5 is the time constant of the first turbine stage, which models the steam chest.

Turbine torque fractions [F2 F3 F4 F5] — Turbine torque fractions

[0.5 0.5 0 0] (default) | vector of four nonnegative elements

Turbine torque fractions F2 to F5. Must total 1, otherwise an error message appears. Fraction numbers correspond to mass numbers; i.e., F2 is the fraction of torque to be applied to mass #2 of the multimass shaft.

Coeff. of inertia [H2 H3 H4 H5] (s) — Coefficients of inertia

[1.5498 0.24894 0 0] (default) | vector of four nonnegative elements

Coefficients of inertia H2 to H5 (s) are associated with the masses of the multimass shaft.

Dependencies

To enable this parameter, set **Generator type** to Tandem-compound (multi-mass).

Stiffness coeff. [K12 K23 K34 K45] (pu/rad) — Stiffness coefficients

[83.47 42.702 0 0] (default) | vector of four nonnegative elements

Stiffness coefficients K12 to K45 (pu/rad), associated with the masses of the multimass shaft. K12 corresponds to the rigidity coefficient between masses #1 and #2, and so on.

Dependencies

To enable this parameter, set **Generator type** to Tandem-compound (multi-mass).

Damping factors [D2 D3 D4 D5] (pu T/pu dw) — Damping factors

[0.3104 0.05 0 0]*8 (default) | vector of four nonnegative elements

Damping factors D2 to D5 (pu torque / puspeed deviation) associated with the masses of the multimass shaft.

Dependencies

To enable this parameter, set **Generator type** to Tandem-compound (multi-mass).

Initial power and generator rotor angle [Pm0 (pu) th0(deg)] — Initial power and generator rotor angle

[2.7247e-008, -120.13] (default) | vector of two elements

Initial mechanical power Pm0 (pu) and initial generator angle θ_{e0} (degrees).

Initial mechanical power is automatically updated by the load flow utility of the Powergui block. Initial angle is also computed by the load flow utility and is written in the associated Synchronous Machine block dialog box.

Dependencies

To enable this parameter, set **Generator type** to Tandem-compound (multi-mass).

Initial power Pm0 (pu) — Initial mechanical power

250.35/555 (default) | scalar

Initial mechanical power.

Initial mechanical power is automatically updated by the load flow utility of the Powergui block. Initial angle is also computed by the load flow utility and is written in the associated Synchronous Machine block dialog box.

Dependencies

To enable this parameter, set **Generator type** to Tandem-compound (single mass).

Version History

Introduced before R2006a

References

- [1] IEEE committee report, "Dynamic models for steam and hydro turbines in power system studies," *IEEE Transactions on Power Apparatus and Systems*, Vol. PAS-92, No. 6, 1973, pp. 1904-1915.
- [2] IEEE Subsynchronous resonance working group, "Second benchmark model for computer simulation of subsynchronous resonance," *IEEE Transactions on Power Apparatus and Systems*, Vol. PAS-104, No. 5, 1985, pp. 1057-1066.

Extended Capabilities**C/C++ Code Generation**

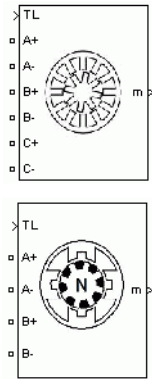
Generate C and C++ code using Simulink® Coder™.

See Also

Excitation System | Hydraulic Turbine and Governor | powergui | Synchronous Machine SI Fundamental | Synchronous Machine pu Fundamental | Synchronous Machine pu Standard

Stepper Motor

Implement stepper motor model



Libraries:

Simscape / Electrical / Specialized Power Systems / Electrical Machines

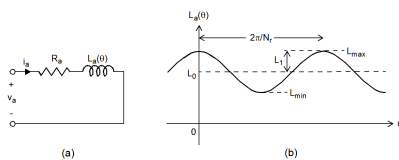
Description

The Stepper Motor (STM) block implements a generic model that represents two most popular families of stepper motors:

- Variable-reluctance stepper motors
- Permanent-magnet or hybrid stepper motors

The Stepper Motor model consists of electrical and mechanical sections. The electrical section is represented by an equivalent circuit, configuration of which depends on the motor type. The equivalent circuits assume that the magnetic circuit is linear (no saturation) and the mutual inductance between phases is negligible. The mechanical section is represented by a state-space model based on inertia moment and viscous friction coefficient.

This figure shows the equivalent circuit for one phase in a variable-reluctance stepper motor.



In this model, R_a and $L_a(\theta)$, respectively, represent the resistance and the inductance of phase A winding. The winding inductance varies as a function of the rotor position:

$$L_a(\theta) = L_0 + L_1 \cos(N_r \theta), \quad (1-13)$$

where,

- L_0 is the average inductance.
- L_1 is the maximum inductance variation.
- N_r is the rotor teeth number.

At the reference position ($\theta = 0$), the rotor tooth is fully aligned with the A-axis pole to achieve the maximum A-phase winding inductance.

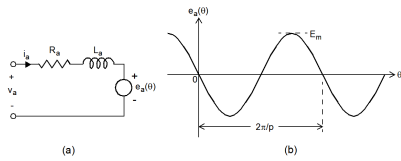
The total electromagnetic torque produced by the motor is the sum of the torques produced by the motor phases:

$$T_e = \sum_{x=1}^m 0.5 i_x^2 \frac{dL_x}{d\theta},$$

where,

- m is the number of phases.
- i_x is the winding current in phase x .
- L_x is the inductance function of phase x winding.

This figure shows the equivalent circuit for one phase in a permanent-magnet (PM) or hybrid stepper motor.



In this model, R_a and L_a , respectively, represent the resistance and inductance of A-phase winding. Due to the large value of the air gap introduced by the magnets, the winding inductance of the PM or hybrid stepper motor can be considered to be independent of the rotor position. The voltage source $e_a(\theta)$ represents the motor back electromotive force (EMF), which is a sinusoidal function of the rotor position:

$$e_a(\theta) = -p\psi_m \sin(p\theta) \frac{d\theta}{dt},$$

where,

- p is the number of pole pairs. The number of pole pairs p is given by $p = Nr/2$.
- ψ_m is the motor maximum magnetic flux.

Note that at the reference position ($\theta = 0$), the north pole on the rotor is fully aligned with the A-axis pole to achieve zero value of the A-phase back EMF.

The electromagnetic torque produced by a two-phase PM or hybrid stepper motor is equal to the sum of the torque resulting from the interaction of the phase currents and magnetic fluxes created by the magnets and the detent torque, which results from the saliency of the rotor:

$$T_e = -p\psi_m i_a \sin(p\theta) - p\psi_m i_b \sin(p\theta - \pi/2) - T_{dm} \sin(mNr\theta). \quad (1-14)$$

where,

- m is the number of phase ($m=2$) of the motor.
- Nr the number of teeth on the rotor ($Nr = 2*p$).

How to Get Stepper Motor Parameters

The parameters used in the stepper model are usually obtained from the manufacturer data sheets. In case the parameters are not available, you can determine them from experimental measurements.

Variable-Reluctance Stepper Motor Parameters

The parameters provided by manufacturer data sheets are usually: number of phases, holding torque, step angle, voltage per phase, current per phase, winding resistance, R_a , maximum inductance, L_{max} , average inductance, L_0 , and rotor inertia, J .

Permanent-Magnet/Hybrid Stepper Motor Parameters

The parameters provided by manufacturer data sheets are usually:

- number of phases
- holding torque
- step angle
- voltage per phase
- current per phase
- winding resistance, R_a
- winding inductance, L_a
- rotor inertia, J

The maximum detent torque, T_{dm} , is not always specified. This parameter can be assumed to be equal to 1-10% of the maximum holding torque.

The maximum flux linkage, ψ_m , is not always specified. This parameter can be obtained experimentally by driving the motor to a constant speed, N , in rpm, and by measuring the maximum open-circuit winding voltage, E_m , in V.

The parameter ψ_m is then computed by the following relation:

$$\psi_m = (30/p\pi)(E_m/N), \quad (1-15)$$

where p is the number of pole pairs given by $p = 360 / (2m \cdot \text{step})$. Here m = phase number, step = step angle in degrees.

Ports

Input

TL — Mechanical load torque
scalar

Mechanical load torque, in N.m. TL is positive in motor operation and negative in generator operation.

Output

m — Motor outputs
vector of five elements

The Simulink output of the block is a vector containing five signals. You can demultiplex these signals by using the Bus Selector block provided in the Simulink library.

Signal	Definition	Units	Symbol
1	Phase voltage	V	V_{ph}
2	Phase current	A	I_{ph}
3	Electromagnetic torque	N.m	T_e
4	Rotor speed	rad/s	w
5	Rotor position	rad	Theta

Conserving

A+ — Positive phase A
specialized electrical

Specialized electrical conserving port associated with the positive terminal of phase A.

A- — Negative phase A
specialized electrical

Specialized electrical conserving port associated with the negative terminal of phase A.

B+ — Positive phase B
specialized electrical

Specialized electrical conserving port associated with the positive terminal of phase B.

B- — Negative phase B
specialized electrical

Specialized electrical conserving port associated with the negative terminal of phase B.

C+ — Positive phase C
specialized electrical

Specialized electrical conserving port associated with the positive terminal of phase C.

Dependencies

To enable this port, set the **Motor type** parameter to Variable reluctance and the **Number of phases** parameter to 3, 4, or 5.

C- — Negative phase C
specialized electrical

Specialized electrical conserving port associated with the negative terminal of phase C.

Dependencies

To enable this port, set the **Motor type** parameter to Variable reluctance and the **Number of phases** parameter to 3, 4, or 5.

D+ — Positive phase D
specialized electrical

Specialized electrical conserving port associated with the positive terminal of phase D.

Dependencies

To enable this port, set the **Motor type** parameter to `Variable reluctance` and the **Number of phases** parameter to 4 or 5.

D- — Negative phase D

specialized electrical

Specialized electrical conserving port associated with the negative terminal of phase D.

Dependencies

To enable this port, set the **Motor type** parameter to `Variable reluctance` and the **Number of phases** parameter to 4 or 5.

E+ — Positive phase E

specialized electrical

Specialized electrical conserving port associated with the positive terminal of phase E.

Dependencies

To enable this port, set the **Motor type** parameter to `Variable reluctance` and the **Number of phases** parameter to 5.

E- — Negative phase E

specialized electrical

Specialized electrical conserving port associated with the negative terminal of phase E.

Dependencies

To enable this port, set the **Motor type** parameter to `Variable reluctance` and the **Number of phases** parameter to 5.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Motor type — Type of motor

`Permanent-magnet / Hybrid (default)` | `Variable reluctance`

Option to implement a permanent-magnet/hybrid stepper motor or a variable-reluctance stepper motor.

Number of phases — Number of phases to model

`2 (default)` | `3` | `4` | `5`

Number of phases to model. If you set **Motor type** to `Permanent-magnet / Hybrid`, you can set this parameter only to 2 and 4. If you set **Motor type** to `Permanent-magnet / Hybrid`, you can set this parameter only to 3, 4, and 5.

Maximum winding inductance (H) — Maximum inductance of each phase winding

`10e-3 (default)`

Maximum inductance, L_{max} , in H, of each phase winding.

Dependencies

To enable this parameter, set **Motor type** to Variable reluctance.

Winding Inductance (H) — Inductance of each phase winding
10e-3 (default)

Inductance, in H, of each phase winding.

Dependencies

To enable this parameter, set **Motor type** to Permanent-magnet / Hybrid.

Minimum winding inductance — Minimum inductance of each phase winding
2e-3 (default)

Minimum inductance, L_{min} , in H, of each phase winding.

Dependencies

To enable this parameter, set **Motor type** to Variable reluctance.

Winding resistance (Ohm) — Resistance of each phase winding
1.2 (default)

Resistance, R_a , in ohms, of each phase winding.

Maximum Flux Linkage (Vs) — Maximum flux linkage produced by magnets
0.04 (default)

Maximum flux linkage, ψ_m , in Vs, produced by the magnets

Dependencies

To enable this parameter, set **Motor type** to Permanent-magnet / Hybrid.

Maximum Detent Torque (N.m) — Maximum detent torque from rotor saliency
0.02 (default)

Maximum detent torque, T_{dm} , in N.m, resulting from the saliency of the rotor.

Dependencies

To enable this parameter, set **Motor type** to Permanent-magnet / Hybrid.

Step angle (degree) — Step angle of rotor movement
30 (default)

Step angle, in degrees, of the rotor movement.

Total inertia (Kg.m.m) — Total inertia momentum of motor and load
1e-4/5 (default)

Total inertia momentum, J , in kg.m², of the motor and the load.

Total friction (N.m.s) — Total viscous friction coefficient of motor and load
1e-3 (default)

Total viscous friction coefficient, B , in N.m.s, of the motor and the load.

Initial speed (rad/s) — Initial rotation speed

$\emptyset$ (default)

Initial rotation speed, ω_0 , in rad/s.

Initial position (degrees) — Initial rotor position

$\emptyset$ (default)

Initial rotor position, θ_0 , in degrees.

Sample time (-1 for inherited) — Stepper motor sample time

-1 (default)

Stepper motor sample time, in s. Specify -1 to inherit the sample time of the powergui block in your model.

Version History

Introduced in R2008a

References

- [1] T. Kenjo, A. Sugawara, *Stepping Motors and Their Microprocessor Controls*, 2nd Edition, Oxford University Press, Oxford, 2003.
- [2] P. Acarnley, *Stepping Motors - A guide to theory and practice*, 4th Edition, The Institution of Electrical Engineers, London, 2002.

Extended Capabilities

C/C++ Code Generation

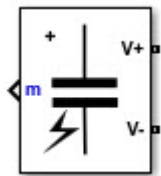
Generate C and C++ code using Simulink® Coder™.

See Also

Switched Reluctance Motor

Supercapacitor

Generic supercapacitor model

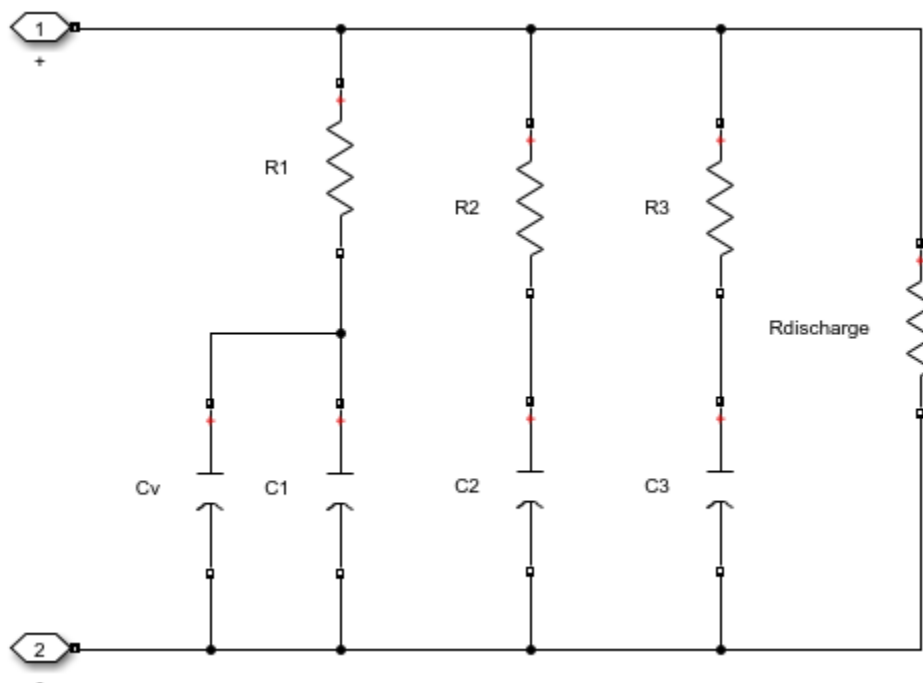


Libraries:

Simscape / Electrical / Specialized Power Systems / Sources

Description

The Supercapacitor block implements a generic supercapacitor model. This figure shows the equivalent circuit of the supercapacitor:



The capacitors C_1 , C_2 , and C_3 have fixed capacitances. The capacitance of capacitor C_v depends on the voltage across it. The resistors R_1 , R_2 , and R_3 have fixed resistances.

The equation for the current through the first branch of the supercapacitor depends on the voltage across the capacitors in the branch,

$$i_1 = \begin{cases} (C_1 + K_v V_{C1}) \frac{dV_{C1}}{dt} & \text{if } V_{C1} > 0 \\ C_1 \frac{dV_{C1}}{dt} & \text{if } V_{C1} < 0 \end{cases}$$

where:

- V_{C_1} is the voltage across the capacitors in the first branch.
- C_1 is the capacitance of the fixed capacitor in the first branch.
- K_v is the voltage-dependent capacitance gain.
- i_1 is the current through the first branch.

This equation defines the current through the remaining branches:

$$i_n = C_n \frac{dV_{C_n}}{dt},$$

where n is the number of the branch in the range [2,3] and C_n is the capacitance of the n th branch.

The block derives the values of the resistors (R_1 , R_2 , and R_3), capacitors (C_1 , C_2 , and C_3), and K_v from the block parameters by using the methodology in [1].

Ports

Conserving

V+ — Positive terminal
specialized electrical

Specialized electrical conserving port associated with the supercapacitor positive terminal.

V- — Negative terminal
specialized electrical

Specialized electrical conserving port associated with the supercapacitor negative terminal.

Output

m — Measurement signals
vector of three elements

Measurement signals. You can demultiplex these signals using the Bus Selector block.

Signal	Definition	Units	Symbol
1	Supercapacitor current	A	Current
2	Supercapacitor voltage	V	Voltage
3	State of charge (SOC), in the range [0,100]	%	SOC

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Parameters

Rated capacitance (F) — Nominal capacitance of supercapacitor
99.5 (default)

Nominal capacitance of the supercapacitor, in farad.

Equivalent DC series resistance (Ohms) — Internal resistance of supercapacitor
8.9e-3 (default)

Internal resistance of the supercapacitor, in ohms.

Rated voltage (V) — Rated voltage of supercapacitor
48 (default)

Rated voltage of the supercapacitor, in volts. Typical rated voltage is equal to 2.7 V.

Number of series capacitors — Number of series capacitors
18 (default)

Number of series capacitors to be represented.

Number of parallel capacitors — Number of parallel capacitors
1 (default)

Number of parallel capacitors to represent.

Initial voltage (V) — Initial voltage of supercapacitor
0 (default)

Initial voltage of the supercapacitor, in volts.

Operating temperature (Celsius) — Operating temperature of supercapacitor
25 (default)

Operating temperature of the supercapacitor. The nominal temperature is 25° C.

Stern

Number of layers — Number of layers
1 (default)

Number of layers related to the Stern model.

Molecular radius (m) — Molecular radius
1e-9 (default)

Molecular radius related to the Stern model, in meters.

Permittivity of electrolyte material (F/m) — Permittivity of electrolyte material
6.0208e-10 (default)

Permittivity of the electrolyte material, in farad/meter.

Self-discharge

Current prior open-circuit (A) — Current prior to open-circuit event
10 (default)

Current prior to an open-circuit event, in amperes.

Voltage @ 0 s, 10 s, 100 s, and 1000 s [V_oc, V_3, V_4, V_5] (V) — Supercapacitor voltage at 0 s, 10 s, 100 s
[48 47.8 47.06 44.65] (default)

Supercapacitor voltage, in volts, at 0 s, 10 s, 100 s, and at 1000 s, when the supercapacitor is open-circuit. Specify the corresponding current prior to open-circuit in the **Current prior open-circuit (A)** parameter.

Version History

Introduced in R2013a

R2024b: Internal implementation has changed

The internal implementation of the Supercapacitor block has changed. The block no longer models the self-discharge effects and the ability to load predetermined parameters or test data of the Stern model.

As a result of this change, these parameters have been removed:

- **Use predetermined parameters**
- **Estimate using test data (requires the Optimization Toolbox)**
- **Charge current (A)**
- **Voltage @ 0 s, 20 s, and 60 s [V_0, V_2, V_3] (V)**
- **Simulate self-discharge**
- **Plot charge characteristics**
- **Charge current [i_1, i_2, i_3, ...] (A)**
- **Time units**

References

[1] Zubieta, L. and R. Bonert. "*Characterization of Double-Layer Capacitors for Power Electronics Applications*." IEEE Transactions on Industry Applications, Vol. 36, No. 1, 2000, pp. 199-205.

Extended Capabilities

C/C++ Code Generation

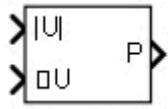
Generate C and C++ code using Simulink® Coder™.

See Also

Variable Capacitor

SVPWM Generator (2-Level)

Generate pulses for SVPWM-controlled two-level converter



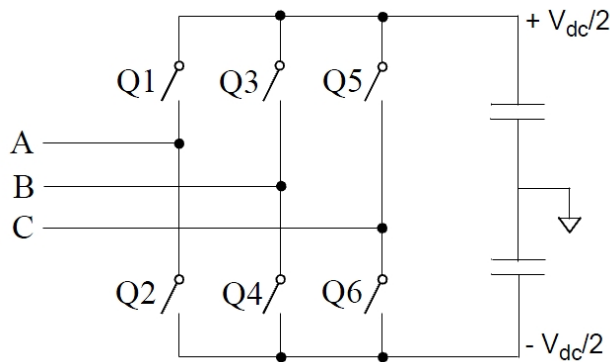
Libraries:

Simscape / Electrical / Specialized Power Systems / Power Electronics / Power Electronics Control

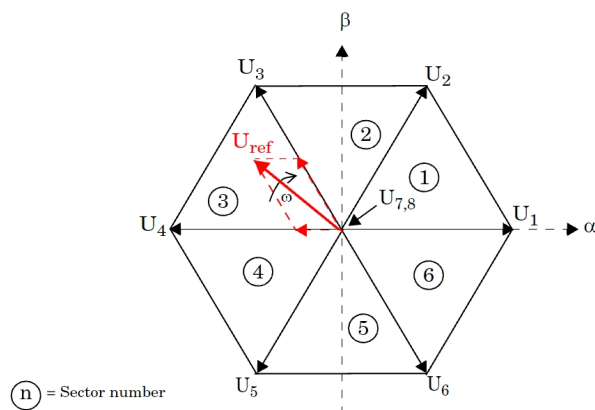
Description

The SVPWM Generator (2-Level) block generates pulses for three-phase two-level DC/AC converters using the space-vector pulse width modulation (SVPWM) technique.

The converter switches are represented by the following equivalent circuit:



As shown in the following figure, the objective of the SVPWM technique is to approximate the reference voltage vector (U_{ref}) instantaneously by combining the switching states corresponding to the basic space vectors.



More precisely, for every PWM period, the reference vector U_{ref} is averaged by using its two adjacent space vectors (U_3 and U_4 in the figure) for a certain duration of time and a null vector (U_7 or U_8) for the rest of the period.

Vector	Q1	Q3	Q5
U ₁	1	0	0
U ₂	1	1	0
U ₃	0	1	0
U ₄	0	1	1
U ₅	0	0	1
U ₆	1	0	1
U ₇	0	0	0
U ₈	1	1	1

The block implements two symmetrical switching patterns [1] on page 1-487:

Pattern #1: With this pattern, known as Software-Determined, each PWM channel switches twice per every PWM period.

Pattern #2: With this pattern, known as Hardware-Determined, one PWM channel remains constant for the entire PWM period. Consequently, the number of switching times for this pattern is less than pattern #1. As a result, switching losses are reduced with Pattern #2.

Characteristics

Sample Time	Specified in the Sample Time parameter Continuous if Sample Time = 0
Scalar Expansion	No
Dimensionalized	No

Examples

The power_SVPWMGenerator2Level model uses two simple circuits to show how the SVPWM Generator (2-Level) works and to compare the two switching pattern options. Run the simulation and open the FFT Analysis tool of the Powergui block to see the harmonics and the THD value of the voltages produced by the two converters.

The model sample time is parameterized by the Ts variable set to a default value of 2e-6 s. Set Ts to 0 in the command window and change the **Simulation type** parameter of the Powergui block to Continuous to simulate the model in continuous mode.

Ports

Input

|U| — Magnitude of rotating reference vector

Magnitude of the rotating reference vector U_{ref} .

Dependencies

To enable this port, set **Data type of input reference vector (Uref)** to Magnitude-Angle (rad).

θ_U — Phase of rotating reference vector

Phase, in radians, of the rotating reference vector U_{ref} .

Dependencies

To enable this port, set **Data type of input reference vector (Uref)** to Magnitude-Angle (rad).

U_α — Alpha components of rotating reference vector

Alpha components of the rotating reference vector U_{ref} .

Dependencies

To enable this port, set **Data type of input reference vector (Uref)** to alpha-beta components.

U_β — Beta components of rotating reference vector

Beta components of the rotating reference vector U_{ref} .

Dependencies

To enable this port, set **Data type of input reference vector (Uref)** to alpha-beta components.

Output

P — Pulse signals used to fire self-commutated devices

Six pulse signals to fire the self-commutated devices (Q1 to Q6) of a converter device.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Data type of input reference vector (Uref) — Type of reference vector

Magnitude-Angle (rad) (default) | alpha-beta components | Internally generated

Specify the type of reference vector.

When set to Magnitude-Angle (rad) (default), the U_{ref} vector is specified by the following inputs:

$|U|$ Magnitude of Uref (value between 0 and 1)

When set to alpha-beta components, the U_{ref} vector is specified by the U_α and U_β inputs.

When set to Internally generated, the U_{ref} is no longer an input to the block. It is internally generated in order to control the converter output voltage as specified in the output voltage parameter.

Switching pattern — Switching pattern

Pattern #1 (default) | Pattern #2

When set to Pattern #1 (default), every device of the converter switches twice per every PWM period.

When set to **Pattern #2**, the status of one of the three arms stays constant (no switching) for the entire PWM period.

PWM frequency (Hz) — PWM frequency
2000 (default)

PWM frequency that determines the PWM period.

$$PWMperiod = 1/PWMFrequency$$

Output voltage: [Mag (0<m<1), Phase (degrees), Freq (Hz)] — Magnitude, phase, and frequency of output voltage
[0.8 -30 50] (default)

Magnitude, phase, and frequency of the output voltage of the two level converter controlled by the block. The magnitude of the fundamental component of the converter line-to-line output voltage is defined as

$$V_{LLrms} = m \times \frac{V_{dc}}{\sqrt{2}}$$

Dependencies

To enable this parameter, set **Data type of input reference vector Uref** to Internally generated.

Sample time — Sample time
0 (default)

Specify the sample time of the block, in seconds. Set to 0 to implement a continuous block.

Version History

Introduced in R2013a

References

[1] Yu, Z., *Application Report SPRA524, Space-Vector PWM with TMS320C24x Using H/W & S/W Determined Switching Patterns*, Texas Instruments, 1999.

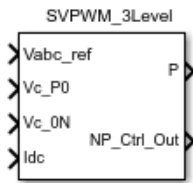
Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using Simulink® Coder™.

See Also
SVPWM Generator (3-Level)

SVPWM Generator (3-Level)

Generate pulses for SVPWM-controlled three-level converter



Libraries:

Simscape / Electrical / Specialized Power Systems / Power Electronics / Power Electronics Control

Description

The SVPWM Generator (3-Level) generates pulses for three-phase three-level Neutral-Point-Clamped (NPC) converters. The block generates twelve pulses using the space vector pulse width modulation (SVPWM) technique.

The neutral-point voltage deviation is controlled by a proportional regulator using two DC voltages, as well as the DC current flowing in or out of the DC link. The defining equation is

$$V_{out} = m \cdot \frac{V_{dc}}{\sqrt{2}},$$

where:

- V_{out} is the line-to-line rms voltage generated by the NPC.
- m is the modulation index and $0 < m < 1$.
- V_{dc} is the DC current flowing in or out of the DC link.

Ports

Input

Vabc_ref — Three-phase sinusoidal reference signal
vector

Reference voltage in terms of sinusoidal voltages. Specify three voltages, one per phase, that you want the attached converter to output.

Dependencies

Setting the **Voltage reference (Vref)** parameter to Three-phase signals exposes this parameter.

Data Types: single | double

|Vref|/∠Vref — Reference voltage magnitude and angle
vector

Reference voltage in terms of magnitude and angle, in rad. Specify the magnitude and angle of the voltage that you want the attached converter to output.

Dependencies

Setting the **Voltage reference (Vref)** parameter to Magnitude-Angle (rad) exposes this parameter.

Data Types: single | double

V α V β ref — Reference voltage alpha and beta components
vector

Reference voltage in terms of alpha and beta components. Specify the alpha and beta components of the voltage that you want the attached converter to output.

Dependencies

Setting the **Voltage reference (Vref)** parameter to alpha-beta components exposes this parameter.

Data Types: single | double

X (internal) — No input
null

When the block generates the reference voltage internally, no reference signal is input.

Dependencies

Setting the **Voltage reference (Vref)** parameter to Internally generated exposes this parameter.

Vc_PO — Three-level NPC converter positive-to-neutral voltage
scalar

Voltage between the + and N terminals of the three-level NPC converter block that is controlled by the SVPWM Generator (3-Level) block.

Data Types: single | double

Vc_ON — Three-level NPC converter neutral-to-negative voltage
scalar

Voltage between the N and - terminals of the three-level NPC converter block that is controlled by the SVPWM Generator (3-Level) block.

Data Types: single | double

Idc — DC link current
scalar

Current of the DC link flowing out the three-level NPC converter block that is controlled by the SVPWM Generator (3-Level) block

Data Types: single | double

Output

P — Gate control
scalar

12-pulse waveforms that determine switching behavior in the attached power converter.

Data Types: `single` | `double`

NP_Ctrl_Out — Internal control voltage
scalar

Neutral-point deviation voltage of the generator control system.

Data Types: `single` | `double`

Parameters

Voltage reference (Vref) — Parameterization method
Three-phase signals (default) | Magnitude-Angle (rad) | alpha-beta components | Internally generated

Reference-voltage parameterization method.

Dependencies

Setting this parameter to:

- **Three-phase signals** exposes the **Vabc_ref** port, hides the **Reference vector [Mag (0<m<1), Phase (degrees), Freq (Hz)]** parameter, and hides the **|Vref|∠Vref**, **Vα Vβ ref**, and **X (internal)** ports.
- **Magnitude-Angle (rad)** exposes the **|Vref|∠Vref** port, hides the **Reference vector [Mag (0<m<1), Phase (degrees), Freq (Hz)]** parameter, and hides the **Vabc_ref**, **Vα Vβ ref**, and **X (internal)** ports.
- **alpha-beta components** exposes the **Vα Vβ ref** port, hides the **Reference vector [Mag (0<m<1), Phase (degrees), Freq (Hz)]** parameter, and hides the **Vabc_ref**, **|Vref|∠Vref**, and **X (internal)** ports.
- **Internally generated** exposes the **Reference vector [Mag (0<m<1), Phase (degrees), Freq (Hz)]** parameter and the **X (internal)** port and hides the **Vabc_ref**, **|Vref|∠Vref**, and **Vα Vβ ref** ports.

Proportional gain — Proportional gain
0.04 (default) | scalar

Constant for proportional gain.

Idc filter cut-off frequency (Hz) — Cut-off frequency
30 (default) | positive scalar

I_{dc} filter cut-off frequency, in Hz.

PWM switching frequency (Hz) — Switching frequency
1620 (default) | positive scalar

PWM switching frequency, in Hz.

Reference vector [Mag (0<m<1), Phase (degrees), Freq (Hz)] — Reference vector
[0.8 -30 60] (default) | vector

Three-element reference vector of magnitude, in m, phase, in degrees, and frequency, in Hz.

Dependencies

Setting the **Voltage reference (Vref)** parameter to Internally generated exposes this parameter.

Sample time — Block sample time
0 (default) | scalar

Time between consecutive block executions. During execution, the block produces outputs and, if appropriate, updates its internal state.

Version History

Introduced in R2018b

Extended Capabilities**C/C++ Code Generation**

Generate C and C++ code using Simulink® Coder™.

See Also

SVPWM Generator (2-Level)

Surge Arrester

Implement metal-oxide surge arrester



Library

Simscape / Electrical / Specialized Power Systems / Passives

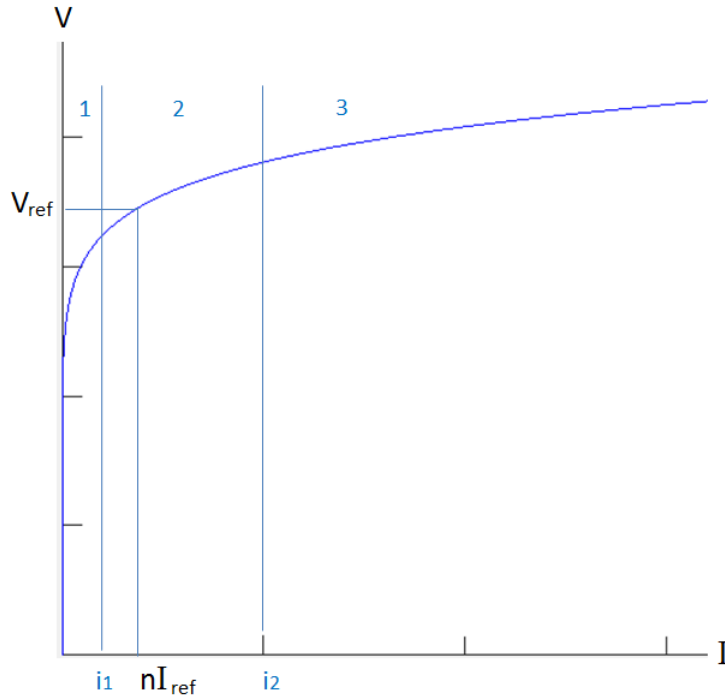
Description

The Surge Arrester block implements a highly nonlinear resistor used to protect power equipment against overvoltages. For applications requiring high power dissipation, several columns of metal-oxide discs are connected in parallel inside the same porcelain housing. The nonlinear V-I characteristic of each column of the surge arrester is modeled by a combination of three exponential functions of the form

$$\frac{V}{V_{\text{ref}}} = k_i \left(\frac{I}{I_{\text{ref}}} \right)^{1/\alpha_i}, \quad i = 1, 2, 3.$$

The protection voltage obtained with a single column is specified at a reference current (usually 500 A or 1 kA). Default parameters k and α given in the dialog box fit the average V-I characteristic provided by the main metal-oxide arrester manufacturers and they do not change with the protection voltage. The required protection voltage is obtained by adding discs of zinc oxide in series in each column.

This figure represents such a V-I characteristic.



Numbers 1, 2, and 3 identify the three segments (or the exponential functions defining the model). The letter n is the number of metal-oxide disc columns.

The transition currents from segment 1 to segment 2 and from segment 2 to segment 3 are equal to:

$$i_1 = P_1 \left(\frac{P_1}{P_2} \right)^{\frac{\alpha_1}{(\alpha_2 - \alpha_1)}}$$

$$i_2 = P_2 \left(\frac{P_2}{P_3} \right)^{\frac{\alpha_2}{(\alpha_3 - \alpha_2)}}$$

Where:

$$P_i = \frac{nI_{ref}}{K_i^{\alpha_i}}, \quad i = 1, 2, 3.$$

You may need to tune the parameters of the three segments to fit the V-I characteristic usually provided in the surge arrester datasheets.

Parameters

Protection voltage Vref

The protection voltage of the Surge Arrester block, in peak voltage (V). Default is 500e+03.

Number of columns

The number of metal-oxide disc columns. The minimum is one. Default is 2.

Reference current per column Iref

The reference current of one column used to specify the protection voltage, in peak amperes (A). Default is 500.

Segment 1 characteristics

The k and α parameters of segment 1. Default is [.955 50].

Segment 2 characteristics

The k and α parameters of segment 2. Default is [1.0 25].

Segment 3 characteristics

The k and α characteristics of segment 3. Default is [.9915 16.5].

Measurements

Select Branch voltage to measure the voltage across the Surge Arrester block terminals.

Select Branch current to measure the current flowing through the Surge Arrester block.

Select Branch voltage and current to measure the surge arrester voltage and current.

Default is None.

Place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurements** list box of the Multimeter block, the measurement is identified by a label followed by the block name.

Measurement	Label
Branch voltage	Ub :
Branch current	Ib :

Advanced Tab

The **Advanced** tab of the block is not visible when you set the **Simulation type** parameter of the powergui block to Continuous, or when you select the **Automatically handle discrete solver** parameter of the powergui block. The tab is visible when you set the **Simulation type** parameter of the powergui block to Discrete, and when the **Automatically handle discrete solver** parameter of the powergui block is cleared.

Break algebraic loop in discrete model

When selected, a delay is inserted at the output of the model. This delay eliminates the algebraic loop resulting from trapezoidal discretization methods and speeds up the simulation of the model. However, this delay introduces a one simulation step time delay in the model and can cause numerical oscillations if the sample time is too large. The algebraic loop is required in most cases to get an accurate solution.

When cleared (default), the discretization method of the model is specified by the **Use robust solver in discrete model** parameter.

Use robust solver in discrete model

When cleared, Simulink iterates to resolve the algebraic loop. Although this method produces correct results, it is not recommended because Simulink tends to slow down and may fail to converge (simulation stops) especially when the number of surge arresters is increased. Also, because of the Simulink algebraic loop constraint, this method cannot be used in real time. In

R2018b and earlier releases, this method was used when the **Break Algebraic loop in discrete saturation model** parameter was cleared.

When selected, the robust solver iterates to resolve the algebraic loop. The maximum number of iterations is specified in the **Preferences** tab of the powergui block, in the Solver details for nonlinear elements section. For real time applications, you may need to limit the number of iterations. Usually, limiting the number of iterations to 2 produces acceptable results. The robust solver is the recommended method for discretizing the surge arrester model.

For more information on what method to use in your application, see “Simulating Discretized Electrical Systems”.

Limitations

The Surge Arrester block is modeled as a current source driven by the voltage appearing across its terminals. Therefore, it cannot be connected in series with an inductor or another current source. As the Surge Arrester block is highly nonlinear, a stiff integrator algorithm must be used to simulate the circuit. `de23t` with default parameters usually gives the best simulation speed. For continuous simulation, in order to avoid an algebraic loop, the voltage applied to the nonlinear resistance is filtered by a first-order filter with a time constant of 0.01 microseconds. This very fast time constant does not significantly affect the result accuracy.

When you use the Surge Arrester block in a discrete system, see the **Advanced** tab.

Version History

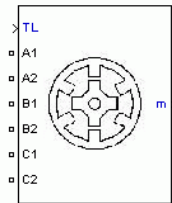
Introduced before R2006a

See Also

Multimeter

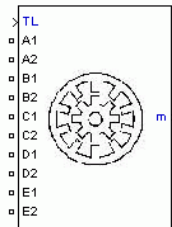
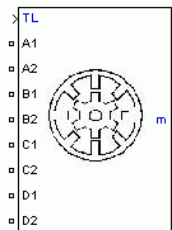
Switched Reluctance Motor

Model the dynamics of switched reluctance motor



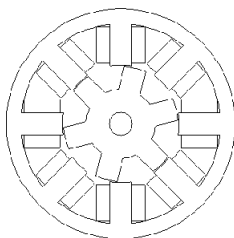
Libraries:

Simscape / Electrical / Specialized Power Systems / Electrical Machines



Description

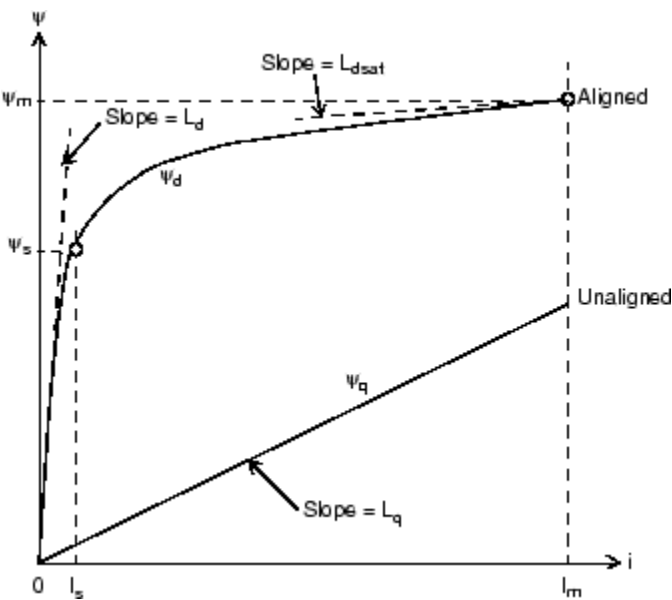
The Switched Reluctance Motor (SRM) block represents three most common switched reluctance motors: three-phase 6/4 SRM, four-phase 8/6 SRM, five-phase 10/8 SRM, as shown in the following figure.



The electric part of the motor is represented by a nonlinear model based on the magnetization characteristic composed of several magnetizing curves and on the torque characteristic computed from the magnetization curves. The mechanic part is represented by a state-space model based on inertia moment and viscous friction coefficient.

To be versatile, two models are implemented for the SRM block: specific and generic models. In the specific SRM model, the magnetization characteristic of the motor is provided in a lookup table. The values are obtained by experimental measurement or calculated by finite-element analysis. In the

generic model, the magnetization characteristic is calculated using nonlinear functions and readily available parameters.



Ports

Input

TL — Mechanical load torque
physical signal

Mechanical load torque (in N.m). TL is positive in motor operation and negative in generator operation.

Output

m — Measurement signal
vector

Vector containing several signals. You can demultiplex these signals by using the Bus Selector block from Simulink library.

Signal	Definition	Units
V	Stator voltages	V
flux	Flux linkage	V.s
I	Stator currents	A
Te	Electromagnetic torque	N.m
w	Rotor speed	rad/s
theta	Rotor position	rad

Conserving**A1 — Phase A**

specialized electrical

Specialized electrical conserving port associated with phase A.

A2 — Phase A

specialized electrical

Specialized electrical conserving port associated with phase A.

B1 — Phase B

specialized electrical

Specialized electrical conserving port associated with phase B.

B2 — Phase B

specialized electrical

Specialized electrical conserving port associated with phase B.

C1 — Phase C

specialized electrical

Specialized electrical conserving port associated with phase C.

C2 — Phase C

specialized electrical

Specialized electrical conserving port associated with phase C.

D1 — Phase D

specialized electrical

Specialized electrical conserving port associated with phase D.

Dependencies

To enable this port, set **Type** to 8/6, 10/8, 8/6 (75 kw preset model), or 10/8 (10 kw preset model).

D2 — Phase D

specialized electrical

Specialized electrical conserving port associated with phase D.

Dependencies

To enable this port, set **Type** to 8/6, 10/8, 8/6 (75 kw preset model), or 10/8 (10 kw preset model).

E1 — Phase E

specialized electrical

Specialized electrical conserving port associated with phase E.

Dependencies

To enable this port, set **Type** to 10/8 or 10/8 (10 kw preset model).

E2 — Phase E

specialized electrical

Specialized electrical conserving port associated with phase E.

Dependencies

To enable this port, set **Type** to 10/8 or 10/8 (10 kw preset model).

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Parameters**Type — Machine type**

6/4 (default) | 8/6 | 10/8 | 6/4 (60 kw preset model) | 8/6 (75 kw preset model) | 10/8 (10 kw preset model)

Set to 6/4 (default) to specify a three-phase 6/4 motor.

Set to 8/6 to specify a four-phase 8/6 motor.

Set to 10/8 to specify a five-phase 10/8 motor.

Set to 6/4 (60 kw preset model), 8/6 (75 kw preset model), or 10/8 (10 kw preset model) to use a predetermined specific model of a switched reluctance motor. When you use these presets, you do not need to specify the parameters in the **Model** tab.

Stator resistance (Ohm) — Stator resistance

0.01 (default)

Resistance R_s (Ω) of each stator phase winding.

Inertia (kg.m.m) — Inertia momentum

0.0082 (default)

Inertia momentum J (kg.m^2).

Friction (N.m.s) — Friction coefficient

0.01 (default)

Friction coefficient B (N.m.s).

Initial speed and position [w0(rad/s) Theta0(rad)] — Initial rotation speed and rotor position

[0 0] (default)

Initial rotation speed w_0 (rad/s) and initial rotor position Theta_0 (rad).

Sample time (-1 for inherited) — Block sample time

-1 (default)

Sample time used by the block. To inherit the sample time specified in the Powergui block, set this parameter to `-1`.

Model

Machine model — Machine model
`Generic model` (default) | `Specific model`

Select `Generic model` (default) or `Specific model`. The **Model** tab is modified accordingly.

Unaligned inductance (H) — Stator inductance when rotor is in unaligned position
`0.67e-3` (default)

Stator inductance when the rotor is in unaligned position L_q (H).

Dependencies

To enable this parameter, set **Machine model** to `Generic model`.

Aligned inductance (H) — Unsaturated stator inductance when rotor is in aligned position
`23.6e-3` (default)

Unsaturated stator inductance when the rotor is in aligned position L_d (H).

Dependencies

To enable this parameter, set **Machine model** to `Generic model`.

Saturated aligned inductance (H) — Saturated stator inductance when rotor is in aligned position
`0.15e-3` (default)

Saturated stator inductance when the rotor is in aligned position L_{dsat} (H).

Dependencies

To enable this parameter, set **Machine model** to `Generic model`.

Maximum current (A) — Stator maximum current
`450` (default)

Stator maximum current I_m (A).

Dependencies

To enable this parameter, set **Machine model** to `Generic model`.

Maximum flux linkage (V.s) — Maximum flux linkage
`0.486` (default)

Maximum flux linkage ψ_m (Wb or V.s) corresponding to I_m .

Dependencies

To enable this parameter, set **Machine model** to `Generic model`.

Plot magnetization curves — Option to plot magnetization curves
`off` (default) | `on`

If selected, the mask plots the magnetization curves corresponding to the lookup table provided. The magnetization curves represent the machine flux linkage versus the stator current with the rotor position as a parameter.

Source — Source option

Dialog (default) | MAT-file

Select **Dialog** to specify the magnetization characteristic directly in the mask of the block. Select **Mat file** (default) to specify the magnetization characteristic from data in a MAT file.

Dependencies

To enable this parameter, set **Machine model** to **Specific model**.

Magnetization characteristic table — Magnetization characteristic table

'srm64_60kw.mat' (default)

When the **Source** parameter is set to **Dialog**, enter the 2-D lookup table containing the flux linkage as a function of stator current and rotor position.

When the **Source** parameter is set to **MAT-file**, enter the name of the MAT-file that contains the 2-D lookup table flux linkage table, the rotor angle vector, and the stator current vector. The MAT-file must contain these three variable names: FTBL, RotorAngles, and StatorCurrents.

Dependencies

To enable this parameter, set **Machine model** to **Generic model**.

Rotor angle vector (degrees) — Rotor position for flux linkage

[0 10 20 30 40 45] (default)

Rotor position θ (deg) for which the flux linkage is specified.

Dependencies

To enable this parameter, set **Machine model** to **Generic model** and **Source** to **Dialog**.

Stator current vector (A) — Stator current for flux linkage

[0:25:450] (default)

The stator current I_s (A) for which the flux linkage is specified.

Dependencies

To enable this parameter, set **Machine model** to **Generic model** and **Source** to **Dialog**.

Version History

Introduced before R2006a

References

- [1] T.J.E. Miller, *Switched Reluctance Motors and Their Control*, Clarendon Press, Oxford, 1993.
- [2] R. Krishnan, *Switched Reluctance Motor Drives*, CRC Press, 2001.

- [3] D.A. Torrey, X.M. Niu, E.J. Unkauf, "Analytical modelling of variable-reluctance machine magnetisation characteristics," IEE Proceedings - Electric Power Applications, Vol. 142, No. 1, January 1995, pp. 14-22.
- [4] H. Le-Huy, P. Brunelle, "Design and Implementation of a Switched Reluctance Motor Generic Model for Simulink SimPowerSystems," Electrimacs 2005 Conference.

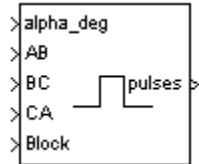
Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

Synchronized 6-Pulse Generator

Implement synchronized pulse generator to fire thyristors of six-pulse converter



Library

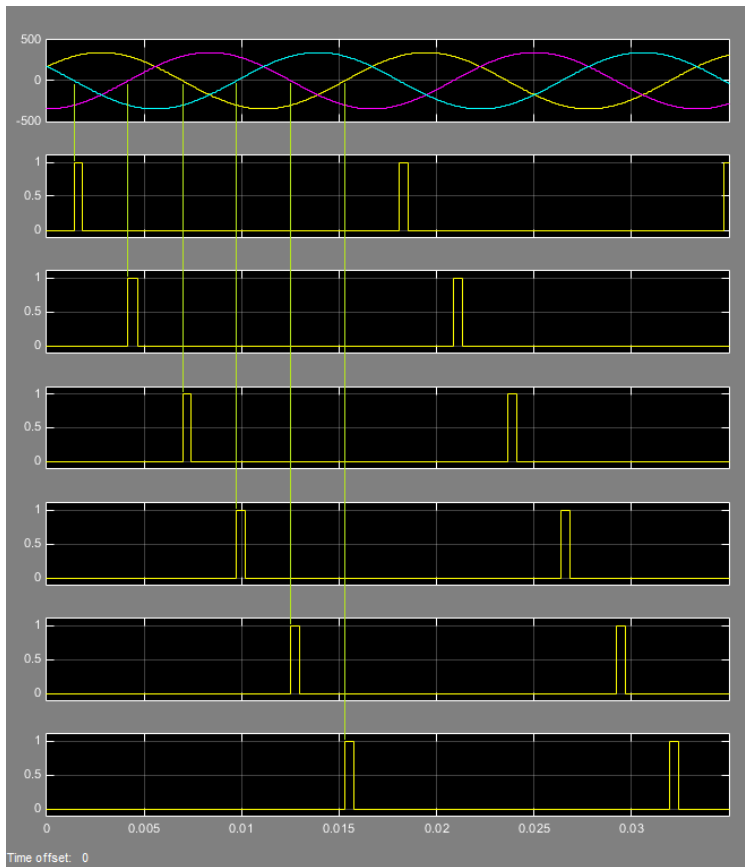
powerlib_extras/Control Blocks, powerlib_extras/Discrete Control Blocks

Note The Pulse & Signal Generators section of the Control and Measurements library contains the Pulse Generator (Thyristor, 6-Pulse) block. This is an improved version of the Synchronized 6-Pulse Generator block. The new block features a mechanism that eliminates duplicate continuous and discrete versions of the same block by basing the block configuration on the simulation mode. If your legacy models contain the Synchronized 6-Pulse Generator block, they will continue to work. However, for best performance, use the Pulse Generator (Thyristor, 6-Pulse) block in your new models.

Description

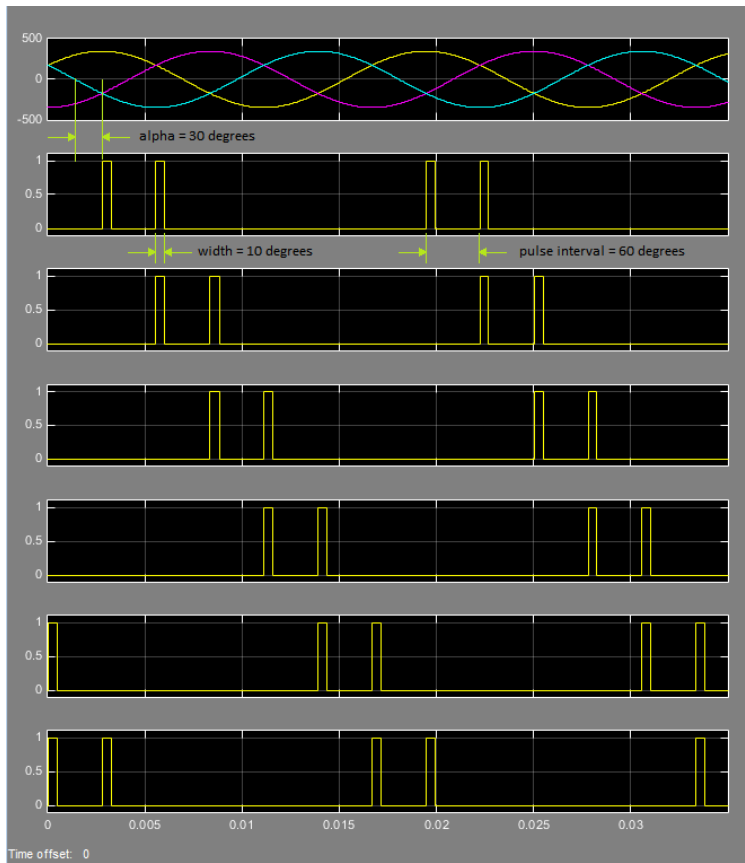
The Synchronized 6-Pulse Generator block can be used to fire the six thyristors of a six-pulse converter. The output of the block is a vector of six pulses individually synchronized on the six thyristor voltages. The pulses are generated alpha degrees after the increasing zero crossings of the thyristor commutation voltages.

The figures below display the synchronization of the six pulses for an alpha angle of 0 degrees. The pulses are generated exactly at the zero crossings of the three line-to-line synchronization voltages.

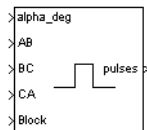


The Synchronized 6-Pulse Generator block can be configured to work in double-pulsing mode. In this mode two pulses are sent to each thyristor: a first pulse when the alpha angle is reached, then a second pulse 60 degrees later, when the next thyristor is fired.

The figures below display the synchronization of the six pulses for an alpha angle of 30 degrees and with double-pulsing mode. Notice that the pulses are generated 30 degrees after the zero crossings of the line-to-line.



The pulse ordering at the output of the block corresponds to the natural order of commutation of a three-phase thyristor bridge. When you connect the Synchronized 6-Pulse Generator block to the pulses input of the Universal Bridge block (with the thyristors as the power electronic device), the pulses are sent to the thyristors in the following order:



When you build your own three-phase thyristor bridge with single thyristor blocks, you need to connect the pulse signals of the Synchronized 6-Pulse Generator block to the gate inputs of the corresponding thyristors.

Parameters

Frequency of synchronization voltages

The frequency, in hertz, of the synchronization voltages. It usually corresponds to the frequency of the network.

Pulse width

The width of the pulses, in degrees.

Double pulsing

If selected, the generator sends to each thyristor a first pulse when the alpha angle is reached, and then a second pulse 60 degrees later when the next thyristor in the sequence is fired.

Inputs and Outputs**alpha_deg**

Input 1 is the alpha firing signal, in degrees. This input can be connected to a Constant block, or it can be connected to a controller system to control the pulses of the generator.

AB, BC, CA

Inputs 2, 3, and 4 are the phase-to-phase synchronization voltages V_{ab} , V_{bc} , and V_{ca} . The synchronization voltages should be in phase with the three phase-phase voltages at the converter AC terminals. Synchronization voltages are normally derived at the primary windings of the converter transformer. If the converter is connected to the delta winding of a Wye/Delta transformer, the synchronization voltages should be the phase-to-ground voltages of the primary windings.

Freq

Available only with the discrete version of the Synchronized 6-Pulse Generator. This input should be connected to a Constant block containing the fundamental frequency, in hertz, or to a PLL tracking the frequency of the system.

block

Input 5 allows you to block the operation of the generator. The pulses are disabled when the applied signal is greater than zero.

pulses

The output contains the six pulse signals.

Examples

The `power_sixpulses` example uses a Pulse Generator (Thyristor, 6-Pulse) block (the improved version of the Synchronized 6-Pulse Generator block) to fire the thyristors of a six-pulse thyristor bridge. The bridge is fed by a three-phase voltage source (200 V peak line-to-ground or 245 V RMS line-to-line) and it is connected to a resistive load.

A first simulation is performed with an alpha angle of 0 degrees. Open the Constant block connected at input 1 of the Pulse Generator (Thyristor) block and set its value to 0. Start the simulation. The average voltage is

$$V_{dc} = \frac{3\sqrt{2}}{\pi} E = \frac{3\sqrt{2}}{\pi} 245 = 331 \text{ volts.}$$

Now change the value of the alpha angle to 30 degrees and start the simulation. Notice that the waveforms of the thyristor voltages look different from the previous case. The thyristors start conducting 30 degrees after their commutation voltage becomes positive and the resulting DC voltage at the output of the rectifier is lower. Its average value is now

$$V_{dc} = \frac{3\sqrt{2}}{\pi} E \cos \alpha = \frac{3\sqrt{2}}{\pi} 245 \cos(30^\circ) = 286 \text{ volts.}$$

Version History

Introduced before R2006a

See Also

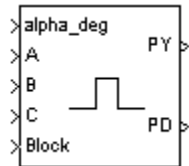
Pulse Generator (Thyristor) | Synchronized 12-Pulse Generator

Topics

“Simple 6-Pulse HVDC Transmission System”

Synchronized 12-Pulse Generator

Implement synchronized pulse generator to fire thyristors of twelve-pulse converter



Library

powerlib_extras/Control Blocks, powerlib_extras/Discrete Control Blocks

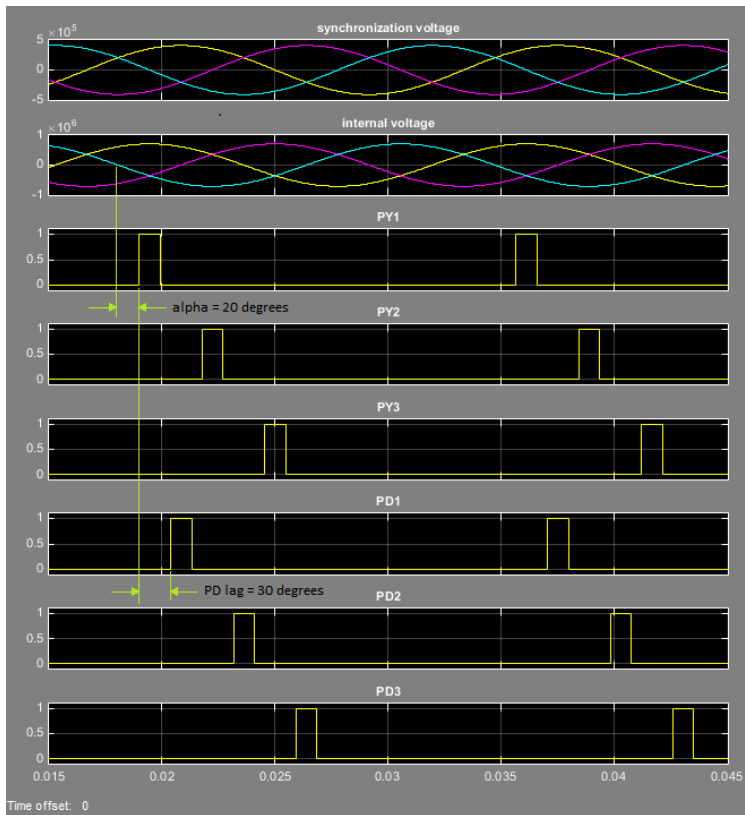
Note The Pulse & Signal Generators section of the Control and Measurements library contains the Pulse Generator (Thyristor, 12-Pulse) block. This is an improved version of the Synchronized 12-Pulse Generator block. The new block features a mechanism that eliminates duplicate continuous and discrete versions of the same block by basing the block configuration on the simulation mode. If your legacy models contain the Synchronized 12-Pulse Generator block, they will continue to work. However, for best performance, use the Pulse Generator (Thyristor, 12-Pulse) block in your new models.

Description

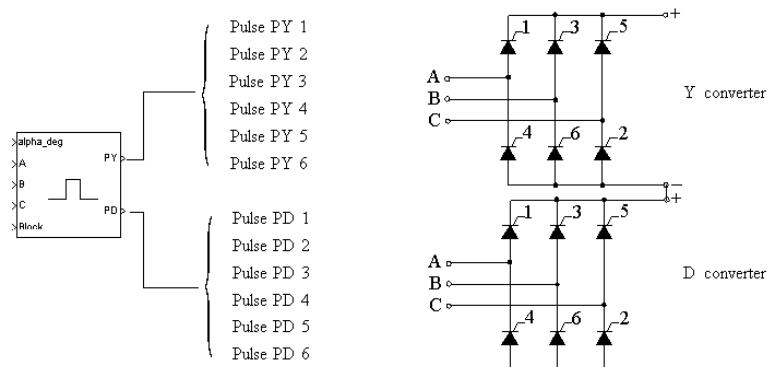
The Synchronized 12-Pulse Generator block generates two vectors of six pulses synchronized on the twelve thyristor commutation voltages. The first set of pulses, denoted PY, is sent to the six-pulse bridge connected to the wye (Y) secondary winding of the Y/Y/Delta converter transformer. The second set of pulses, denoted PD, is sent to the six-pulse bridge connected to the delta secondary winding of the converter transformer.

The synchronization voltages provided to the generator are the three phase-to-ground voltages V_a , V_b , V_c measured on the primary side (Y) of the Y/Y/Delta converter transformer. The PY pulses are generated alpha degrees after the zero crossing of the phase-to-phase voltages generated from the synchronization voltages. The PD pulses can be set to lead or lag the PY pulses by 30 degrees.

The following figure shows the synchronization and internal phase-to-phase voltages plus the first three pulses of the PY and PD output vectors. In this example the PY pulses are generated 20 degrees (alpha angle) after the zero crossing of the phase-to-phase voltages and the PD pulses lag the PY pulses by 30 degrees.



The ordering of the pulses in the two outputs of the block corresponds to the natural order of commutation of a three-phase thyristor bridge. When you connect the Synchronized 12-Pulse Generator block outputs to the pulse inputs of the Universal Bridge blocks (with the thyristor device), the pulses are sent to the thyristors in the following way:



Parameters

Delta winding connection

If set to D1 (lagging), the PD pulses lag the PY pulses by 30 degrees. If set to D11 (leading), the PD pulses lead the PY pulses by 30 degrees.

Frequency of synchronization voltages

The frequency, in hertz, of the synchronization voltages. It usually corresponds to the frequency of the network.

Pulse width

The width of the pulses, in degrees.

Double pulsing

If selected, the generator sends to each thyristor a first pulse when the alpha angle is reached, and then a second pulse 60 degrees later when the next thyristor in the sequence is fired. The double pulsing is applied separately on the two vectors of pulses.

Inputs and Outputs**alpha_deg**

Input 1 is the alpha firing signal, in degrees. This input can be connected to a Constant block, or it can be connected to a controller system to control the pulses of the generator.

A, B, C

Inputs 2, 3, and 4 are the phase-to-ground synchronization voltages V_a , V_b , and V_c . The synchronization voltages should be measured at the primary side of the converter transformer.

Freq

Available only with the discrete version of the Synchronized 6-Pulse Generator block. This input should be connected to a Constant block containing the fundamental frequency, in hertz, or to a PLL tracking the frequency of the system.

block

Input 5 allows you to block the operation of the generator. The pulses are disabled when the applied signal is greater than zero.

PY

Output 1 contains the six-pulse signals to be sent to the six-pulse thyristor converter connected to the Y secondary winding of the converter transformer.

PD

Output 2 contains the six-pulse signals to be sent to the six-pulse thyristor converter connected to the Delta (D) secondary winding of the converter transformer.

Examples

In the `power_twelvepulses` example a Pulse Generator (Thyristor) block (the improved version of the Synchronized 12-Pulse Generator block) is used to fire the thyristors of a twelve-pulse thyristor bridge built with two six-pulse bridges. The bridge is fed by a three-winding three-phase transformer (500 kV / 200 kV / 200 kV). The Y-connected secondary feeds the first six-pulse bridge. The Delta secondary feeds the second bridge. The transformer is assumed to be ideal (no leakage reactances, no resistance). The expected DC voltage obtained for $\alpha = 0$ is

$$V_{dc} = 2 \frac{3\sqrt{2}}{\pi} 200 \text{ kV} = 540 \text{ kV}.$$

The two bridge rectifiers are connected in series and a 300 km DC line is connected to the rectifier.

A first simulation is performed with an alpha angle of 0 degrees. Open the Constant block connected at input 1 of the Pulse Generator (Thyristor, 12-Pulse) block and set its value to 0. Start the

simulation. The voltages of the thyristors of the D Thyristor Converter block are displayed in the next figure.

Compare the DC voltage generated by the Generator (Thyristor, 12-Pulse) block with the DC voltage you obtained with the Generator (Thyristor, 6-Pulse) block. Notice that the ripple in the DC voltage waveform is lower. The rectifier voltage contains the harmonics $12 \cdot k$ ($k = 1, 2, \dots$).

Version History

Introduced before R2006a

See Also

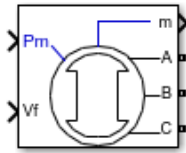
Pulse Generator (Thyristor) | Synchronized 6-Pulse Generator

Topics

“Thyristor-Based HVDC Transmission System”

Synchronous Machine pu Fundamental

Model dynamics of three-phase round-rotor or salient-pole synchronous machine using fundamental parameters in pu units



Libraries:

Simscape / Electrical / Specialized Power Systems / Electrical Machines

Description

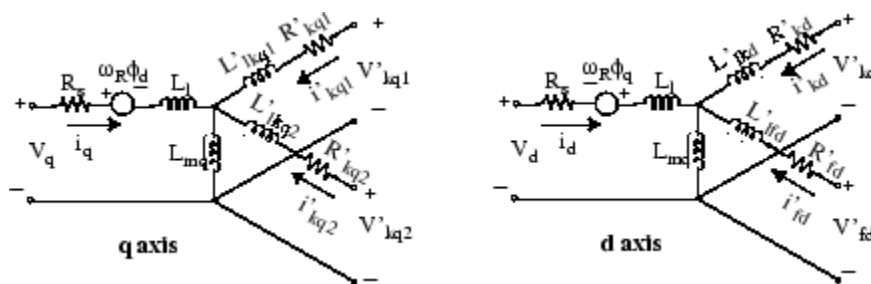
The Synchronous Machine pu Fundamental block models a synchronous machine in generator or motor mode using fundamental parameters in pu units. The operating mode is dictated by the sign of the mechanical power (positive for generator mode or negative for motor mode). The electrical part of the machine is represented by a sixth-order state-space model and the mechanical part is the same as in the Simplified Synchronous Machine block.

For more information on the pu units system, see “Per-Unit System of Units” on page A-2.

The model takes into account the dynamics of the stator, field, and damper windings. The equivalent circuit of the model is represented in the rotor reference frame (the qd frame). Stator windings are connected in wye to an internal neutral point. All rotor parameters and electrical quantities are viewed from the stator and identified by primed variables. The subscripts are:

- d, q — d - and q -axis quantity
- R, s — Rotor and stator quantity
- l, m — Leakage and magnetizing inductance
- f, k — Field and damper winding quantity

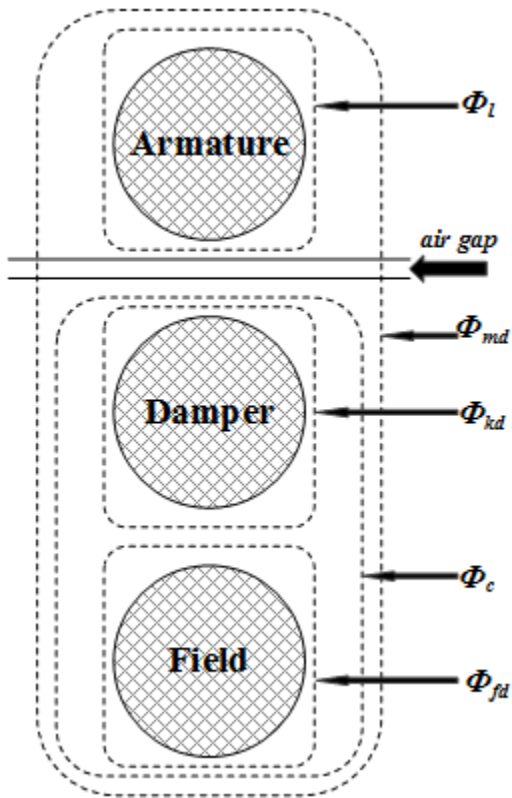
The electrical model of the machine is shown in these diagrams.



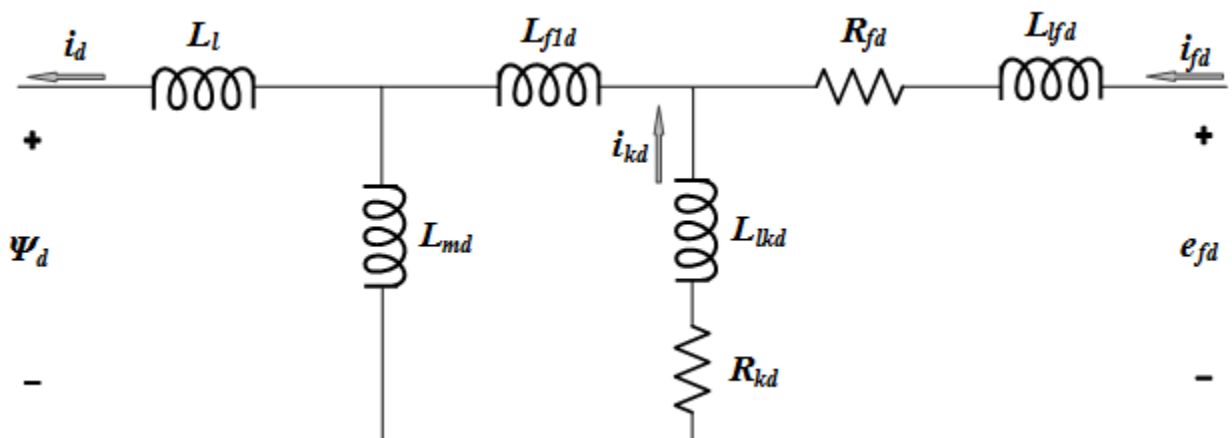
Dynamic Model with Unequal Mutual Inductance

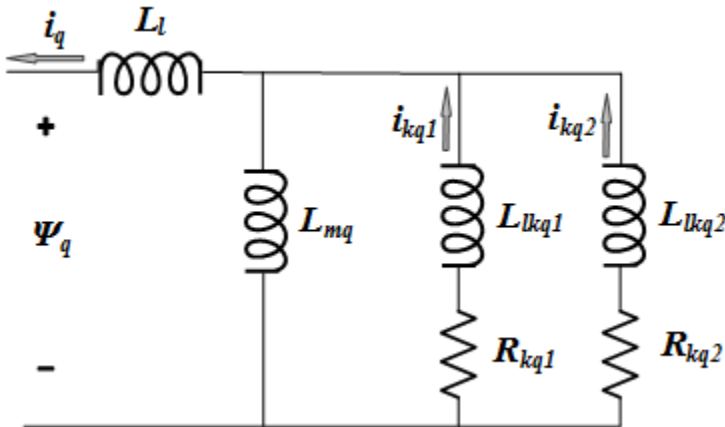
The conventional theory of synchronous machine modeling for stability analysis assumes that the mutual inductances between the armature, damper, and field on direct-axis windings are identical. Generally, damper windings are near the air gap, and as a result the flux linking damper circuits are almost equal to the flux linking armature. This hypothesis produces acceptable outcomes for a wide

range of stability studies, especially those on the side of the network. However, when it comes to field current studies, there is considerable error. The equivalent circuit dynamic model of a synchronous machine may include an additional inductance representing the difference between field-damper and field-armature mutual inductances on the D -axis [1]. This inductance is typically called Canay inductance. Canay inductance corresponds to the leakage flux, Φ_c , in the following figure and is interpreted as a corrective element in an equivalent model that may have a negative value [2].



The IEEE standard 1110-2002 [3] presents the direct and quadratic axes of the synchronous machine dynamic model as shown in the diagrams.





The relevant equations are:

$$V_d = -i_d R_s - \omega \psi_q + \frac{d\psi_d}{dt}$$

$$V_q = -i_q R_s + \omega \psi_d + \frac{d\psi_q}{dt}$$

$$V_0 = -i_0 R_0 + \frac{d\psi_0}{dt}$$

$$V_{fd} = \frac{d\psi_{fd}}{dt} + R_{fd} i_{fd}$$

$$0 = \frac{d\psi_{kd}}{dt} + R_{kd} i_{kd}$$

$$0 = \frac{d\psi_{kq1}}{dt} + R_{kq1} i_{kq1}$$

$$0 = \frac{d\psi_{kq2}}{dt} + R_{kq2} i_{kq2}$$

$$\begin{bmatrix} \psi_d \\ \psi_{kd} \\ \psi_{fd} \end{bmatrix} = \begin{bmatrix} L_{md} + L_l & L_{md} & L_{md} \\ L_{md} & L_{lkd} + L_{f1d} + L_{md} & L_{f1d} + L_{md} \\ L_{md} & L_{f1d} + L_{md} & L_{lfd} + L_{f1d} + L_{md} \end{bmatrix} \begin{bmatrix} -i_d \\ i_{kd} \\ i_{fd} \end{bmatrix}$$

$$\begin{bmatrix} \psi_q \\ \psi_{kq1} \\ \psi_{kq2} \end{bmatrix} = \begin{bmatrix} L_{mq} + L_l & L_{mq} & L_{mq} \\ L_{mq} & L_{mq} + L_{kq1} & L_{mq} \\ L_{mq} & L_{mq} & L_{mq} + L_{kq2} \end{bmatrix} \begin{bmatrix} -i_q \\ i_{kq1} \\ i_{kq2} \end{bmatrix}$$

Base Values and Transformation Ratio

The Synchronous Machine pu Fundamental block allows you to specify the fundamental parameters of a synchronous machine. You enter field and damper winding parameters (resistances, leakage inductances, and mutual inductances) in pu.

Stator Base Values

This table displays the stator base values.

$$V_{sbase} = \frac{V_n \sqrt{2}}{\sqrt{3}} \quad \text{Base stator voltage} = \text{peak nominal line-to-neutral voltage (V)}$$

$$I_{sbase} = \frac{P_n \sqrt{2}}{V_n \sqrt{3}} \quad \text{Base stator current (A)}$$

$$Z_{sbase} = \frac{V_{sbase}}{I_{sbase}} = \frac{V_n^2}{P_n} \quad \text{Base stator impedance } (\Omega)$$

$$\omega_{base} = 2\pi f_n \quad \text{Base angular frequency (rad/s)}$$

$$L_{sbase} = \frac{Z_{sbase}}{\omega_{base}} \quad \text{Base stator inductance (H)}$$

where:

- P_n is the three-phase nominal power (VA).
- V_n is the nominal line-to-line voltage (Vrms).
- f_n is the nominal frequency (Hz).
- i_{fn} is the nominal field current producing nominal stator voltage at no load (A).

The stator parameters to specify in the block are:

$$R_{s\_pu} = \frac{R_s}{Z_{sbase}} \quad \text{Stator resistance per phase (pu)}$$

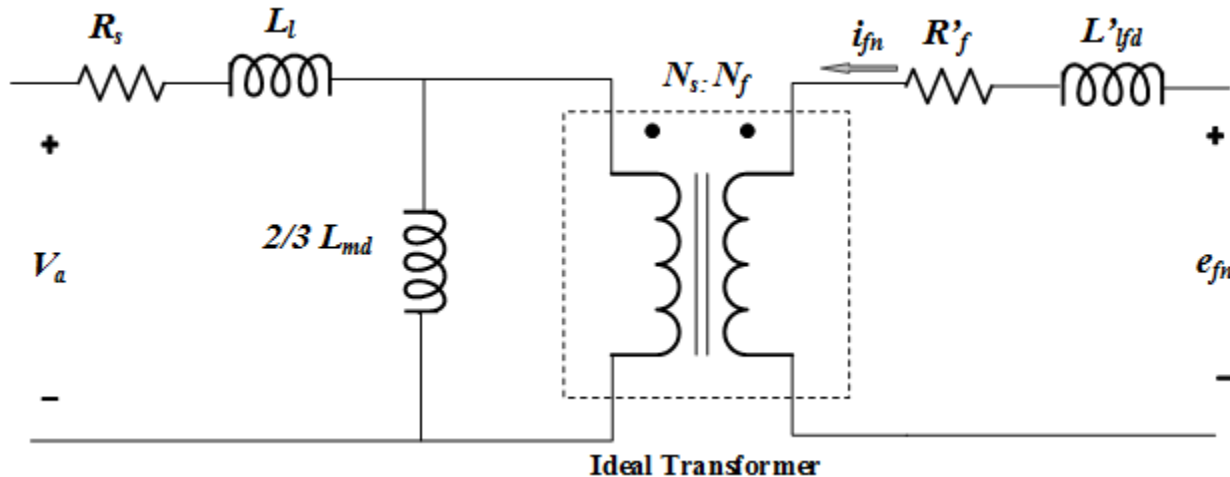
$$L_{l\_pu} = \frac{L_l}{L_{sbase}} \quad \text{Stator leakage inductance (pu)}$$

$$L_{md\_pu} = \frac{L_{md}}{L_{sbase}} \quad \text{Direct-axis magnetizing inductance (pu)}$$

$$L_{mq\_pu} = \frac{L_{mq}}{L_{sbase}} \quad \text{Quadrature-axis magnetizing inductance (pu)}$$

Stator/Field Transformation Ratio

This figure shows one phase of the stator winding coupled with the field winding.



In the diagram:

- N_s and N_f are the equivalent number of sinusoidally distributed turns of the stator winding and of the field winding, respectively.
- R_s and L_l are stator resistance and leakage inductance.
- R_f' and L_{lf}' are field resistance and leakage inductance.

When the three stator windings are energized with a three-phase positive-sequence voltage and the field winding is open, the stator magnetizing inductance is L_{md} . However, when only one phase is energized and the field winding is open, the magnetizing inductance is $2/3 L_{md}$, as shown in the figure.

At no load, when the field winding is rotating at nominal speed and bears the nominal field DC current i_{fn} , the amplitude of AC voltage (peak value) induced on one phase of the stator is V_{sbase} .

The maximum mutual inductance between one stator winding and the field winding is obtained when the two windings are aligned and is given by:

$$L_{sfd} = \frac{2}{3} L_{md} \frac{N_f}{N_s} = \frac{V_{sbase}}{i_{fn} \omega_{base}}$$

from which we deduce the transformation ratio:

$$\frac{N_s}{N_f} = \frac{2}{3} L_{md} \frac{i_{fn} \omega_{base}}{V_{sbase}}.$$

The transformation ratio can be also expressed as:

$$\frac{N_s}{N_f} = \frac{2}{3} \frac{I_{fbase}}{I_{sbase}},$$

where I_{fbase} is the base field current. The table shows how to compute field base values.

$$\begin{aligned} I_{fbase} &= i_{fn} L_{md\_pu} \quad \text{Base field current (A)} \\ V_{fbase} &= \frac{P_n}{I_{fbase}} \quad \text{Base field voltage (V)} \\ Z_{fbase} &= \frac{V_{fbase}}{I_{fbase}} \quad \text{Base field impedance (\Omega)} \\ L_{fbase} &= \frac{Z_{fbase}}{\omega_{base}} \quad \text{Base field inductance (H)} \end{aligned}$$

The actual field parameters are:

- R_f' is field resistance (Ω)
- L_{lf}' is field leakage inductance (H)

The field parameters to enter in the block menu are:

$$R_{f\_pu} = \frac{R_f'}{Z_{fbase}} \quad \text{Field resistance (pu)}$$

$$L_{lfd\_pu} = \frac{L_{lfd'}}{L_{fbase}} \quad \text{Field leakage inductance (pu)}$$

Field Voltage and Current

The voltage at the **Vf** input of the block is normalized with respect to nominal field voltage so that a 1 pu input produces a 1 pu stator voltage at no load. The **If** current returned by the measurement output of the block is also normalized with respect to the nominal field current so that a 1 pu input produces $I_f = 1$ pu.

Examples

Model 555 MVA, 24 kV, 60 Hz, 3600 rpm Synchronous Generator

The `power_SM_Fundamental` example uses the Synchronous Machine SI Fundamental block and Synchronous Machine pu Fundamental block to model a 555 MVA, 24 kV, 60 Hz, 3600 rpm synchronous generator. It shows how to specify SI and pu parameters and explains how to compute field and damper rotor winding parameters that are referred to the stator. In addition to the field winding, the round rotor of this machine has three damper windings: one damper in the direct axis and two dampers in the quadrature axis.

Three circuits simulate the same synchronous machine:

- Circuit 1: Fundamental parameters are specified in SI. The nominal field current is specified ($ifn = 1300$ A).
- Circuit 2: Fundamental parameters are specified in SI. The nominal field current is not specified ($ifn = 0$).
- Circuit 3: Fundamental parameters are specified in pu.

Machine parameters are taken from an example in Kundur [5].

Look at the **Model Properties/PreLoad Fcn** callback section of the model to see machine specifications and the computation of stator and field bases, RL rotor parameters referred to the stator, transformation ratio, and nominal field voltage and current.

The machines initially operate in steady state at virtually no load (load = 0.1% of nominal power) with constant field voltage and mechanical power. A phase-to-phase six-cycle fault is applied at $t = 0.1$ sec. The Scope block shows the comparison between line-to-line AB voltage, phase A stator current, and field current of the three machines.

To simulate the discrete model, in the `powergui` block, set **Simulation type** to **Discrete**. The model is discretized with a sample time $T_s = 50$ μ s. To obtain a stable model with such a small load (0.1% of nominal power), in the **Advanced** tab of each Synchronous Machine block, set **Discrete solver model** to Trapezoidal iterative (alg. loop).

Use Synchronous Machine Block in Motor Mode

The `power_syncmachine` example uses the Synchronous Machine block in motor mode. The simulated system consists of a 150 HP (112 kVA), 762 V industrial-grade synchronous motor connected to a network with a 10 MVA short-circuit level. The machine is initialized for an output

electrical power of -50 kW (negative value for motor mode), corresponding to a mechanical power of -48.9 kW. The corresponding values of mechanical power and field voltage are specified by the Pm Step block and the Vf Constant block. The Pm Step block applies a sudden increase of mechanical power from -48.9 kW to -60 kW at time $t = 0.1$ s.

Run the simulation.

After the load has increased from 48.9 kW to 60 kW at $t = 0.1$ s, the machine speed oscillates before stabilizing to 1800 rpm. The load angle (angle between terminal voltage and internal voltage) increases from -21 degrees to -53 degrees.

Assumptions and Limitations

In discrete systems, when you set the **Discrete solver model** parameter of a Synchronous Machine block to **Trapezoidal non iterative**, you might have to connect a small parasitic resistive load at the machine terminals to avoid numerical oscillations. Large sample times require larger loads. The minimum resistive load is proportional to the sample time. As a rule of thumb, remember that with a $25\text{ }\mu\text{s}$ time step on a 60 Hz system, the minimum load is approximately 2.5% of the machine nominal power. For example, a 200 MVA synchronous machine in a power system discretized with a $50\text{ }\mu\text{s}$ sample time requires approximately 5% of resistive load or 10 MW. If the sample time is reduced to $20\text{ }\mu\text{s}$, a resistive load of 4 MW should be sufficient.

However, if you set the **Discrete solver model** parameter of a Synchronous Machine block to **Trapezoidal iterative (alg. loop)**, you can use a negligible parasitic load (below 0.1% of nominal power) while preserving numerical stability. This iterative model produces an algebraic loop and results in slower simulation speed.

Ports

Input

Pm — Mechanical power
scalar

Mechanical power at the machine's shaft, in pu. In generator mode, this input can be a positive constant or function or the output of a prime mover block (see the Hydraulic Turbine and Governor or Steam Turbine and Governor blocks). In motor mode, this input is usually a negative constant or function.

Dependencies

To enable this port, set the **Mechanical input** parameter (**Configuration** tab) to Mechanical power Pm.

w — Machine speed
scalar

Machine speed, in rad/s.

Dependencies

To enable this port, set the **Mechanical input** parameter (**Configuration** tab) to Speed w.

Vf — Field voltage
scalar

Field voltage, in pu. This voltage can be supplied by a voltage regulator in generator mode (see the Excitation System block). In motor mode, this value of this input is usually a constant.

Output

m — Measurement signal
24-element vector

Measurement signals, returned as a 24-element vector. You can demultiplex these signals by using the Bus Selector block.

Name	Definition	Units
ias	Stator current is_a	pu
ibs	Stator current is_b	pu
ics	Stator current is_c	pu
iq	Stator current iq	pu
id	Stator current id	pu
ifd	Field current ifd	pu
ikq1	Damper winding current ikq1	pu
Ikq2	Damper winding current ikq2	pu
ikd	Damper winding current ikd	pu
phimq	Mutual flux phimq	pu
phimd	Mutual flux phimd	pu
vq	Stator voltage vq	pu
vd	Stator voltage vd	pu
lmq	Lmq saturated inductance	pu
lmd	Lmd saturated inductance	pu
dtheta	Rotor angle deviation d_theta	rad
w	Rotor speed wm	pu
Pe	Electrical power Pe	pu
dw	Rotor speed deviation dw	pu
theta	Rotor mechanical angle theta	degree
Te	Electromagnetic torque Te	pu
delta	Load angle delta	degree
Pe0	Output active power Peo	pu
Qe0	Output reactive power Qeo	pu

The angular deviation of the rotor angle theta represents the electrical angle giving the instantaneous position of the rotor with respect to a common reference rotating at synchronous speed. This angle is useful in stability studies to measure the relative positions of the rotors of different machines in a network. The positions of the rotors are then measured with respect to the theta position of a given machine chosen as a reference.

Conserving

A — Phase A electric terminal
specialized electrical

Specialized electrical conserving port associated with the phase A electrical terminal.

B — Phase B electric terminal
specialized electrical

Specialized electrical conserving port associated with the phase B electrical terminal.

C — Phase C electric terminal
specialized electrical

Specialized electrical conserving port associated with the phase C electrical terminal.

S — Machine rotor shaft
mechanical

Mechanical rotational conserving port associated with the machine rotor shaft.

Dependencies

To enable this port, set the **Mechanical input** parameter (**Configuration** tab) to Mechanical rotational port.

Parameters

Configuration

Preset model — Electrical and mechanical parameters

No (default) | 01: 50Hz 400V 8.1kVA 1500RPM | 02: 50Hz 400V 16kVA 1500RPM | ...

Set of predetermined electrical and mechanical parameters for various synchronous machine ratings of power (kVA), phase-to-phase voltage (V), frequency (Hz), and rated speed (rpm).

Select one of the preset models to load the corresponding electrical and mechanical parameters. Choices are:

- 01: 50Hz 400V 8.1kVA 1500RPM
- 02: 50Hz 400V 16kVA 1500RPM
- 03: 50Hz 400V 31.3kVA 1500RPM
- 04: 50Hz 400V 42.5kVA 1500RPM
- 05: 50Hz 400V 60kVA 1500RPM
- 06: 50Hz 400V 85kVA 1500RPM
- 07: 50Hz 400V 250kVA 1500RPM
- 08: 50Hz 400V 325kVA 1500RPM
- 09: 50Hz 400V 670kVA 1500RPM
- 10: 50Hz 400V 910kVA 1500RPM
- 11: 50Hz 400V 1320kVA 1500RPM

- 12: 50Hz 400V 2000kVA 1500RPM
- 13: 60Hz 460V 10.2kVA 1800RPM
- 14: 60Hz 460V 20kVA 1800RPM
- 15: 60Hz 460V 37.5kVA 1800RPM
- 16: 60Hz 460V 52.5kVA 1800RPM
- 17: 60Hz 460V 72.5kVA 1800RPM
- 18: 60Hz 460V 100kVA 1800RPM
- 19: 60Hz 460V 300kVA 1800RPM
- 20: 60Hz 460V 406kVA 1800RPM
- 21: 60Hz 460V 800kVA 1800RPM
- 22: 60Hz 460V 1075kVA 1800RPM
- 23: 60Hz 460V 1588kVA 1800RPM
- 24: 60Hz 460V 2500kVA 1800RPM

Select **No** (default) if you do not want to use a preset model or if you want to modify some of the parameters of a preset model, as described below.

When you select a preset model, the electrical and mechanical parameters in the **Parameters** tab are dimmed. To start from a preset model and then modify the machine parameters:

- 1 Select the preset model for which you want to initialize the parameters.
- 2 Change the **Preset model** parameter to **No**. This action does not change the machine parameters, but breaks the connection with the preset model.
- 3 Modify the machine parameters as you want.

Mechanical input — Mechanical input

Mechanical power P_m (default) | Speed w | Mechanical rotational port

Whether to represent the mechanical power applied to the shaft or the rotor speed as a Simulink input of the block, or to represent the machine shaft with a Simscape rotational mechanical port.

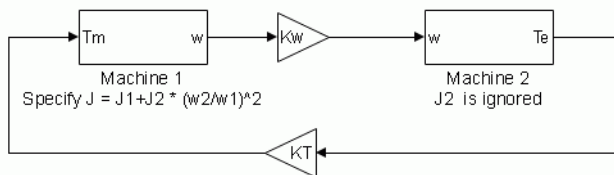
Select **Mechanical power P_m** to specify a mechanical power input, in pu, and expose the **P_m** port. The machine speed is determined by the inertia constant H and by the difference between the mechanical torque T_m , which results from the applied mechanical power P_m , and the internal electromagnetic torque T_e . When the speed is positive, a positive mechanical power signal indicates generator mode and a negative signal indicates motor mode.

Select **Speed w** to specify a speed input in pu and expose the **w** port. The machine speed is imposed and the mechanical part of the model (the inertia constant H) is ignored. Using the speed as the mechanical input allows you to model a mechanical coupling between two machines.

The next figure indicates how to model a stiff shaft interconnection in a motor-generator set where both machines are synchronous machines.

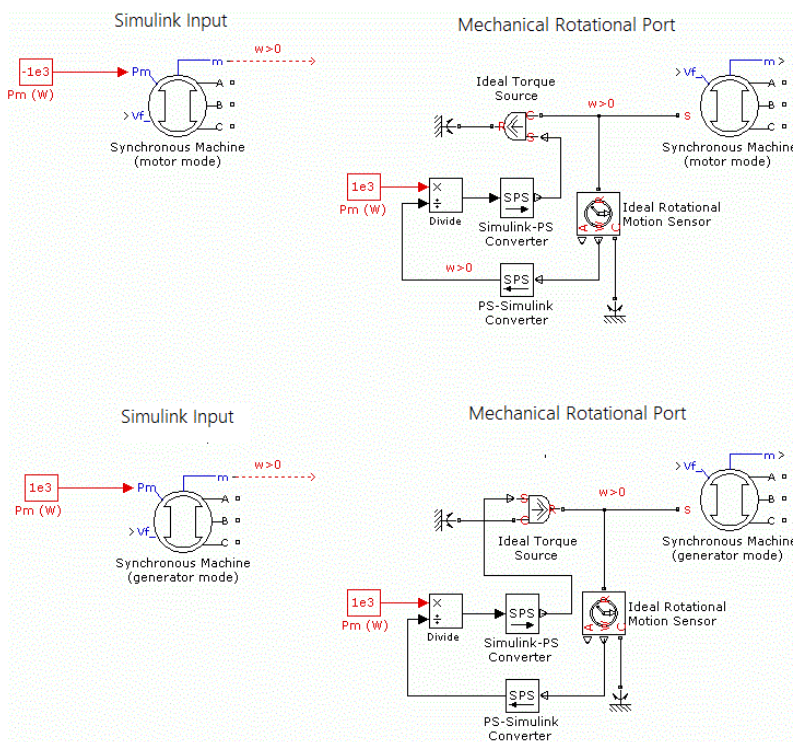
The speed output of machine 1 (the motor) is connected to the speed input of machine 2 (the generator). In this figure, friction torque is ignored in machine 2. Therefore, its electromagnetic torque output **T_e** corresponds to the mechanical torque T_m applied to the shaft of machine 1. The corresponding mechanical input power of machine 1 is computed as $P_m = T_m * w$. The K_w factor takes into account the speed units of both machines (pu) and the gear box ratio w_2/w_1 . The K_T factor takes

into account the torque units of both machines (pu) and the machine ratings. Also, because inertia J_2 is ignored in machine 2, J_2 refers to the speed of machine 1, and speed must be added to machine 1 inertia, J_1 .



Select **Mechanical rotational port** to expose a Simscape mechanical rotational port, **S**, that allows you to connect the machine shaft to another machine shaft or to other Simscape blocks that have mechanical rotational ports.

The figure indicates how to connect an Ideal Torque Source block from the Simscape library to the machine shaft port to represent the machine in motor mode or generator mode when the rotor speed is positive.



Rotor type — Rotor type
Salient-pole (default) | Round

Specify the rotor type as either **Salient-pole** or **Round** (cylindrical). The setting affects the number of rotor circuits in the q -axis (damper windings).

Use signal names to identify bus labels — How to identify bus labels
off (default) | on

When this check box is selected, the measurement output uses the signal names to identify the bus labels. Select this option for applications that require bus signal labels to have only alphanumeric characters.

When this check box is cleared, the measurement output uses the signal definition to identify the bus labels. The labels contain nonalphanumeric characters that are incompatible with some Simulink applications.

Parameters

Nominal power, line-to-line voltage and frequency [$P_n(\text{VA})$ $V_n(\text{Vrms})$ $f_n(\text{Hz})$] — Nominal power, line-to-line voltage, and frequency

[187E6 13800 60] (default) | three-element vector

Total three-phase apparent power P_n (VA), RMS line-to-line voltage V_n (V), and frequency f_n (Hz).

Stator [R_s L_l L_{md} L_{mq} (L_c)] (pu) — Stator resistance and inductances

[2.85E-03 0.114 1.19 0.36] (default) | four-element vector

Resistance R_s (pu), leakage inductance L_l (pu), d -axis and q -axis magnetizing inductances L_{md} (pu) and L_{mq} (pu), and Canay inductance. If the value of the Canay inductance is not known, you must enter 0 or leave it blank.

Field [R_f L_{lfd}] (pu) — Field resistance and leakage inductance

[5.79E-04 0.114] (default) | two-element vector

Field resistance R_f (pu) and leakage inductance L_{lfd} (pu), both referred to the stator.

Dampers [R_{kd} L_{lkd} R_{kq1} L_{lkq1}] (pu) — Resistance and leakage inductance

[1.17E-02 0.182 1.97E-02 0.384] (default) | four-element vector

d -axis resistance R_{kd} (pu) and leakage inductance L_{lkd} (pu), and q -axis resistance R_{kq1} (pu) and leakage inductance L_{lkq1} (pu). All of these values are referred to the stator.

Dependencies

To enable this parameter, set **Rotor type** to Salient-pole.

Dampers [R_{kd} L_{lkd} R_{kq1} L_{lkq1} R_{kq2} L_{lkq2}] (pu) — Resistance and leakage inductance

[0.0652, 0.5134, 0.0287, 0.2553, 7.765e-3, 0.9167] (default) | six-element vector

d -axis resistance R_{kd} (pu) and leakage inductance L_{lkd} (pu), q -axis resistance R_{kq1} (pu) and leakage inductance L_{lkq1} (pu), and q -axis resistance R_{kq2} (pu) and leakage inductance L_{lkq2} (pu). All of these values are referred to the stator.

Dependencies

To enable this parameter, set **Rotor type** to Round.

Inertia coefficient, friction factor, pole pairs [$H(\text{s})$ $F(\text{pu})$ $p()$] — Inertia coefficient, friction factor, and pole pairs

[3.7 0 20] (default) | three-element vector

Inertia constant H (s), which is the ratio of energy stored in the rotor at nominal speed over the nominal power of the machine, friction factor F (pu torque/pu speed), and the number of pole pairs p .

Friction torque T_f is proportional to rotor speed ω ($T_f = F\omega$, where all quantities are expressed in pu).

Dependencies

To enable this parameter, set **Mechanical input** to Mechanical power P_m or Mechanical rotational port.

Pole pairs p () — Number of pole pairs
20 (default) | positive integer

Number of pole pairs of the synchronous machine.

Dependencies

To enable this parameter, set **Mechanical input** to Speed w .

Initial conditions [$\Delta w(\%)$ θ (deg) i_a, i_b, i_c (pu) ϕ_a, ϕ_b, ϕ_c (deg) V_f (pu)] — Initial conditions
[0 0 0 0 0 0 0 0 1] (default) | nine-element vector

Initial speed deviation $\Delta\omega$ (percent of nominal speed), electrical angle of the rotor θ_e (degrees), line current magnitudes i_a, i_b, i_c (pu) and phase angles ϕ_a, ϕ_b, ϕ_c (degrees), and the initial field voltage V_f (pu). You can compute these values automatically by using the Load Flow tool of the powergui block.

Simulate saturation — Simulate magnetic saturation
off (default) | on

Whether the magnetic saturation of the rotor and stator iron is simulated.

Select this check box to provide the matrix of parameters for simulating the saturation.

Clear this check box to not model saturation in your simulation. In this case, the relationship between the i_{fd} and V_t is linear (no saturation).

[$i_{fd}; v_t$] (pu) — No-load saturation curve parameters
[0.6404,0.7127,0.8441,0.9214,0.9956,1.082,1.19,1.316,1.457;0.7,0.7698,0.8872,0.9466,0.9969,1.046,1.1,1.151,1.201] (default) | vector

No-load saturation curve parameters. The magnetic saturation of the stator and rotor iron is modeled by a piecewise linear relationship specifying points on the no-load saturation curve. The first row of this matrix contains the values of field currents (in pu). The second row contains values of corresponding terminal voltages (in pu). The first point (first column of the matrix) must be different from [0,0]. This point corresponds to the point where the effect of saturation begins. For saturation, the nominal field current and nominal RMS line-to-line voltage are the base values for the field current and terminal voltage, respectively.

Click **Plot** to view the no-load saturation curve.

Dependencies

To enable this parameter, select **Simulate saturation**.

Advanced

To enable the **Advanced** tab, in the powergui block, set **Simulation type** to Discrete and clear **Automatically handle discrete solver**.

Sample time (-1 for inherited) — Sample time for block

-1 (default) | scalar

Sample time used by the block. To inherit the sample time specified in the powergui block, set this parameter to -1.

Discrete solver model — Integration method

Trapezoidal non iterative (default) | Trapezoidal iterative (alg. loop) |
Trapezoidal robust | Backward Euler robust

Integration method used by the block when the **Solver type** parameter of the powergui block is set to Discrete.

The **Discrete solver model** is automatically set to Trapezoidal robust when you select the **Automatically handle Discrete solver and Advanced tab solver settings of blocks** parameter of the powergui block.

The Trapezoidal non iterative and Trapezoidal iterative (alg. loop) methods are no longer recommended for discretizing the Synchronous Machine pu Fundamental block. Trapezoidal non iterative requires you to add non-negligible shunt loads at the machine terminals to guarantee simulation stability, and Trapezoidal iterative (alg. loop) may fail to converge and cause the simulation to stop when the number of machines increases in the model.

The Trapezoidal robust and Backward Euler robust methods allow you to eliminate the need to use parasitic loads and simulate a machine without loads. To eliminate topological errors of machines connected to an inductive circuit (for example, a circuit breaker connected in series with the machine); the machine models a negligible internal load of 0.01% of nominal power.

The Trapezoidal robust method is slightly more accurate than the Backward Euler robust method, especially when the model is simulated at larger sample times. The Trapezoidal robust method may produce slight damped numerical oscillations on machine voltage at no-load conditions, while the Backward Euler robust method prevents oscillations and maintains good accuracy.

For more information on what method to use in your application, see Simulating Discretized Electrical Systems.

Load Flow

The load flow parameters are used to define block parameters for use with the Load Flow tool of the powergui block. These load flow parameters are used for model initialization only. They have no impact on the block model or on the simulation performance.

Generator type — Machine generator type

PV (default) | swing | PQ

Generator type of the machine. Choice are:

- **swing** — Implements a generator controlling magnitude and phase angle of its terminal voltage. The reference voltage magnitude and angle are specified by the **Swing bus or PV bus voltage** and **Swing bus voltage angle** parameters of the Load Flow Bus block connected to the machine terminals.
- **PV** — Implements a generator controlling its output active power P and voltage magnitude V . P is specified by the **Active power generation P (W)** parameter of the block. V is specified by the **Swing bus or PV bus voltage** parameter of the Load Flow Bus block connected to the machine

terminals. You can control the minimum and maximum reactive power generated by the block by using the **Minimum reactive power Qmin (var)** and **Maximum reactive power Qmax (var)** parameters.

- PQ — Implements a generator controlling its output active power P and reactive power Q . P and Q are specified by the **Active power generation P (W)** and **Reactive power generation Q (var)** parameters of the block, respectively.

Active power generation P (W) — Active power generated by machine

0 (default) | scalar

Active power that you want generated by the machine, in watts. When the machine operates in motor mode, you specify a negative value.

Dependencies

To enable this parameter, set **Generator type** to PV or PQ.

Reactive power generation Q (var) — Reactive power generated by machine

0 (default) | scalar

Reactive power that you want generated by the machine, in vars. A negative value indicates that the reactive power is absorbed by the machine.

Dependencies

To enable this parameter, set **Generator type** to PQ.

Minimum reactive power Qmin (var) — Minimum reactive power generated by machine

-inf (default) | scalar

Minimum reactive power that can be generated by the machine while keeping the terminal voltage at its reference value. This reference voltage is specified by the **Swing bus or PV bus voltage** parameter of the Load Flow Bus block connected to the machine terminals. The default value is -inf, which means that there is no lower limit on the reactive power output.

Dependencies

To enable this parameter, set **Generator type** to PV.

Maximum reactive power Qmax (var) — Maximum reactive power generated by machine

inf (default) | scalar

Maximum reactive power that can be generated by the machine while keeping the terminal voltage at its reference value. This reference voltage is specified by the **Swing bus or PV bus voltage** parameter of the Load Flow Bus block connected to the machine terminals. The default value is inf, which means that there is no upper limit on the reactive power output.

Dependencies

To enable this parameter, set **Generator type** to PV.

Version History

Introduced before R2006a

References

- [1] Canay, I.M. "Causes of Discrepancies on Calculation of Rotor Quantities and Exact Equivalent Diagrams of the Synchronous Machine." *IEEE Transactions on Power Apparatus and Systems*. PAS-88, no. 7 (1969): 1114-1120.
- [2] Moeini, A., et al. "Synchronous Machine Stability model, an Update to IEEE Std 1110-2002 Data Translation Technique." *IEEE standards panel sessions*. 2018.
- [3] *IEEE Guide for Synchronous Generator Modeling Practices and Applications in Power System Stability Analyses*. IEEE Std 1110-2002 (Revision of IEEE Std 1110-1991 [2003]): 1-72.
- [4] Krause, P.C. *Analysis of Electric Machinery*. Section 12.5. New York: McGraw-Hill, 1986.
- [5] Kundur, P. *Power System Stability and Control*. New York, McGraw-Hill, 1994.

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

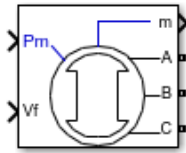
Excitation System | Hydraulic Turbine and Governor | powergui | Simplified Synchronous Machine | Steam Turbine and Governor | Synchronous Machine SI Fundamental | Synchronous Machine pu Standard

Topics

"Per-Unit System of Units" on page A-2

Synchronous Machine pu Standard

Model dynamics of three-phase round-rotor or salient-pole synchronous machine using standard parameters in pu units



Libraries:

Simscape / Electrical / Specialized Power Systems / Electrical Machines

Description

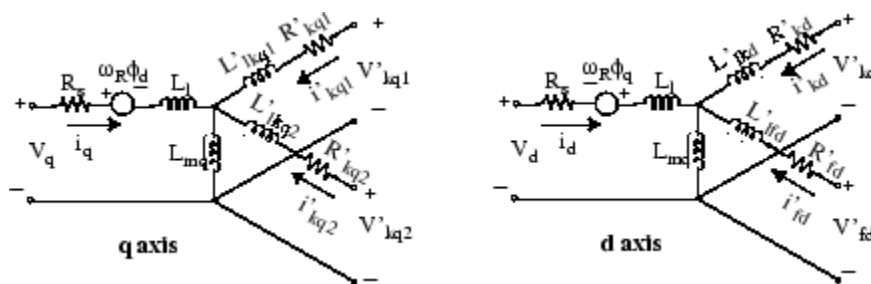
The Synchronous Machine pu Standard block models a synchronous machine in generator or motor mode using standard parameters in pu units. The operating mode is dictated by the sign of the mechanical power (positive for generator mode or negative for motor mode). The electrical part of the machine is represented by a sixth-order state-space model and the mechanical part is the same as in the Simplified Synchronous Machine block.

For more information on the pu units system, see “Per-Unit System of Units” on page A-2.

The model takes into account the dynamics of the stator, field, and damper windings. The equivalent circuit of the model is represented in the rotor reference frame (qd frame). Stator windings are connected in wye to an internal neutral point. All rotor parameters and electrical quantities are viewed from the stator and identified by primed variables. The subscripts are:

- d, q — d - and q -axis quantity
- R, s — Rotor and stator quantity
- l, m — Leakage and magnetizing inductance
- f, k — Field and damper winding quantity

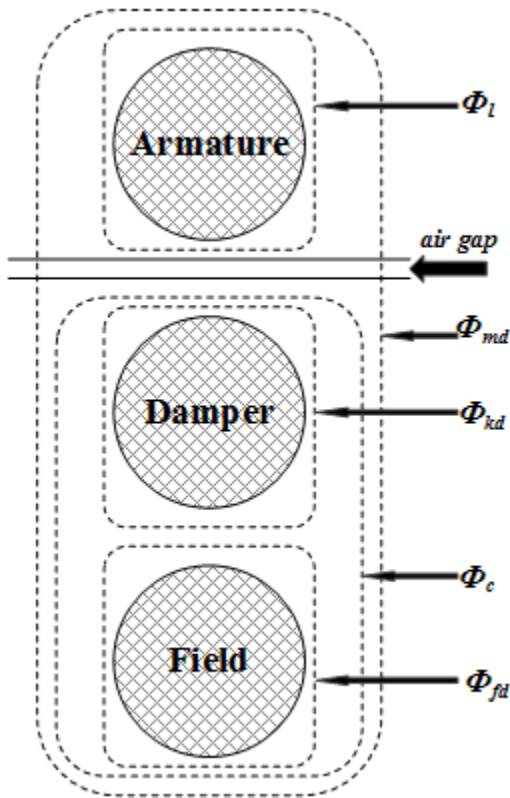
The electrical model of the machine is shown in these diagrams.



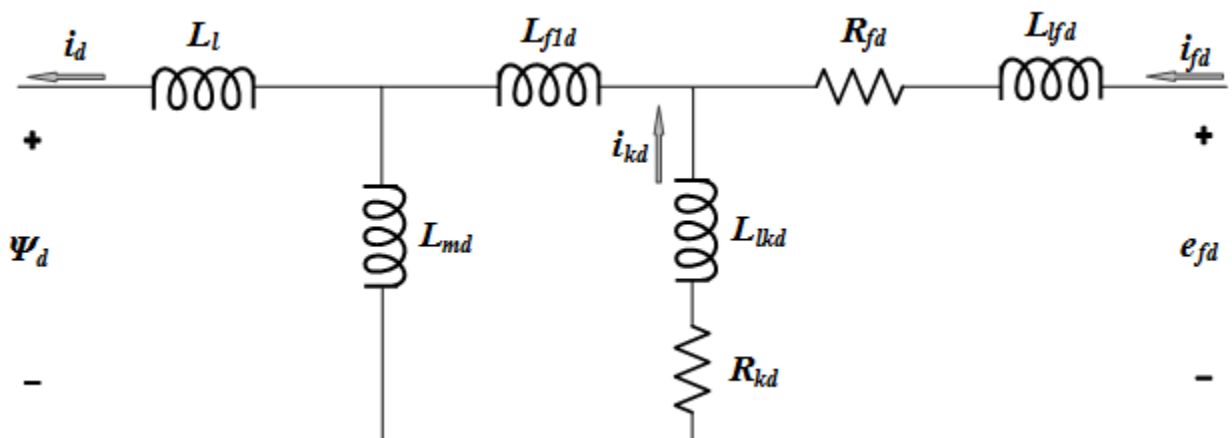
Dynamic Model with Unequal Mutual Inductance

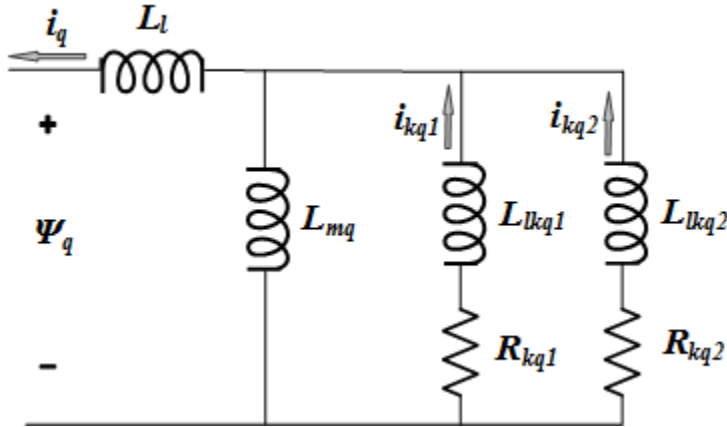
The conventional theory of synchronous machine modeling for stability analysis assumes that the mutual inductances between the armature, damper, and field on direct-axis windings are identical. Generally, damper windings are near the air gap, and as a result the flux linking damper circuits are almost equal to the flux linking armature. This hypothesis produces acceptable outcomes for a wide

range of stability studies, especially those on the side of the network. However, when it comes to field current studies, there is considerable error. The equivalent circuit dynamic model of a synchronous machine may include an additional inductance representing the difference between field-damper and field-armature mutual inductances on the D -axis [1]. This inductance is typically called Canay inductance. Canay inductance corresponds to the leakage flux, Φ_c , in the following figure and is interpreted as a corrective element in an equivalent model that may have a negative value [2].



The IEEE standard 1110-2002 [3] presents the direct and quadratic axes of the synchronous machine dynamic model as shown in the diagrams.





The relevant equations are:

$$V_d = -i_d R_s - \omega \psi_q + \frac{d\psi_d}{dt}$$

$$V_q = -i_q R_s + \omega \psi_d + \frac{d\psi_q}{dt}$$

$$V_0 = -i_0 R_0 + \frac{d\psi_0}{dt}$$

$$V_{fd} = \frac{d\psi_{fd}}{dt} + R_{fd} i_{fd}$$

$$0 = \frac{d\psi_{kd}}{dt} + R_{kd} i_{kd}$$

$$0 = \frac{d\psi_{kq1}}{dt} + R_{kq1} i_{kq1}$$

$$0 = \frac{d\psi_{kq2}}{dt} + R_{kq2} i_{kq2}$$

$$\begin{bmatrix} \psi_d \\ \psi_{kd} \\ \psi_{fd} \end{bmatrix} = \begin{bmatrix} L_{md} + L_l & L_{md} & L_{md} \\ L_{md} & L_{lkd} + L_{f1d} + L_{md} & L_{f1d} + L_{md} \\ L_{md} & L_{f1d} + L_{md} & L_{lfd} + L_{f1d} + L_{md} \end{bmatrix} \begin{bmatrix} -i_d \\ i_{kd} \\ i_{fd} \end{bmatrix}$$

$$\begin{bmatrix} \psi_q \\ \psi_{kq1} \\ \psi_{kq2} \end{bmatrix} = \begin{bmatrix} L_{mq} + L_l & L_{mq} & L_{mq} \\ L_{mq} & L_{mq} + L_{kq1} & L_{mq} \\ L_{mq} & L_{mq} & L_{mq} + L_{kq2} \end{bmatrix} \begin{bmatrix} -i_q \\ i_{kq1} \\ i_{kq2} \end{bmatrix}$$

Examples

Model Sub-Synchronous Resonance in Steam Turbine and Governor on Series-Compensated Network.

The “Subsynchronous Resonance in Steam Turbine and Governor System” example uses the Synchronous Machine pu Standard block to model sub-synchronous resonance (SSR) in a steam turbine and governor on a series-compensated network.

Assumptions and Limitations

In discrete systems, when you set the **Discrete solver model** parameter of a Synchronous Machine block to `Trapezoidal non_iterative`, you might have to connect a small parasitic resistive load at the machine terminals to avoid numerical oscillations. Large sample times require larger loads. The minimum resistive load is proportional to the sample time. As a rule of thumb, remember that with a 25 μ s time step on a 60 Hz system, the minimum load is approximately 2.5% of the machine nominal power. For example, a 200 MVA synchronous machine in a power system discretized with a 50 μ s sample time requires approximately 5% of resistive load or 10 MW. If the sample time is reduced to 20 μ s, a resistive load of 4 MW should be sufficient.

However, if you set the **Discrete solver model** parameter of a Synchronous Machine block to `Trapezoidal iterative (alg. loop)`, you can use a negligible parasitic load (below 0.1% of nominal power) while preserving numerical stability. This iterative model produces an algebraic loop and results in slower simulation speed.

Ports

Input

Pm — Mechanical power
scalar

Mechanical power at the machine's shaft, in pu. In generator mode, this input can be a positive constant or function or the output of a prime mover block (see the Hydraulic Turbine and Governor or Steam Turbine and Governor blocks). In motor mode, this input is usually a negative constant or function.

Dependencies

To enable this port, set the **Mechanical input** parameter (**Configuration** tab) to Mechanical power Pm.

w — Machine speed
scalar

Machine speed, in rad/s.

Dependencies

To enable this port, set the **Mechanical input** parameter (**Configuration** tab) to Speed w.

Vf — Field voltage
scalar

Field voltage, in pu. This voltage can be supplied by a voltage regulator in generator mode (see the Excitation System block). In motor mode, this value of this input is usually a constant.

Output

m — Measurement signal
24-element vector

Measurement signals, returned as a 24-element vector. You can demultiplex these signals by using the Bus Selector block.

Name	Definition	Units
ias	Stator current is_a	pu
ibs	Stator current is_b	pu
ics	Stator current is_c	pu
iq	Stator current iq	pu
id	Stator current id	pu
ifd	Field current ifd	pu
ikq1	Damper winding current ikq1	pu
lkq2	Damper winding current ikq2	pu
ikd	Damper winding current ikd	pu
phimq	Mutual flux phimq	pu
phimd	Mutual flux phimd	pu
vq	Stator voltage vq	pu
vd	Stator voltage vd	pu
lmq	Lmq saturated inductance	pu
lmd	Lmd saturated inductance	pu
dtheta	Rotor angle deviation d_theta	rad
w	Rotor speed wm	pu
Pe	Electrical power Pe	pu
dw	Rotor speed deviation dw	pu
theta	Rotor mechanical angle theta	degree
Te	Electromagnetic torque Te	pu
delta	Load angle delta	degree
Pe0	Output active power Peo	pu
Qe0	Output reactive power Qeo	pu

The angular deviation of the rotor angle theta represents the electrical angle giving the instantaneous position of the rotor with respect to a common reference rotating at synchronous speed. This angle is useful in stability studies to measure the relative positions of the rotors of different machines in a network. The positions of the rotors are then measured with respect to the theta position of a given machine chosen as a reference.

Conserving

A — Phase A electric terminal
specialized electrical

Specialized electrical conserving port associated with the phase A electrical terminal.

B — Phase B electric terminal
specialized electrical

Specialized electrical conserving port associated with the phase B electrical terminal.

C — Phase C electric terminal
specialized electrical

Specialized electrical conserving port associated with the phase C electrical terminal.

S — Machine rotor shaft
mechanical

Mechanical rotational conserving port associated with the machine rotor shaft.

Dependencies

To enable this port, set the **Mechanical input** parameter (**Configuration** tab) to Mechanical rotational port.

Parameters

Specify standard parameters, also known as operational parameters, of the synchronous machine (such as steady-state, transient, and/or subtransient reactances and time constants) as inputs to the block. Typically, machine manufacturers provide operational parameters.

In power system analysis software, the synchronous machine equations are commonly solved using the direct-quadrature-zero transformation technique. A data translation technique calculates fundamental parameters from standard parameters [2], [3].

Configuration

Preset model — Electrical and mechanical parameters

No (default) | 01: 50Hz 400V 8.1kVA 1500RPM | 02: 50Hz 400V 16kVA 1500RPM | ...

Set of predetermined electrical and mechanical parameters for various synchronous machine ratings of power (kVA), phase-to-phase voltage (V), frequency (Hz), and rated speed (rpm).

Select one of the preset models to load the corresponding electrical and mechanical parameters. Choices are:

- 01: 50Hz 400V 8.1kVA 1500RPM
- 02: 50Hz 400V 16kVA 1500RPM
- 03: 50Hz 400V 31.3kVA 1500RPM
- 04: 50Hz 400V 42.5kVA 1500RPM
- 05: 50Hz 400V 60kVA 1500RPM
- 06: 50Hz 400V 85kVA 1500RPM
- 07: 50Hz 400V 250kVA 1500RPM
- 08: 50Hz 400V 325kVA 1500RPM
- 09: 50Hz 400V 670kVA 1500RPM
- 10: 50Hz 400V 910kVA 1500RPM

- 11: 50Hz 400V 1320kVA 1500RPM
- 12: 50Hz 400V 2000kVA 1500RPM
- 13: 60Hz 460V 10.2kVA 1800RPM
- 14: 60Hz 460V 20kVA 1800RPM
- 15: 60Hz 460V 37.5kVA 1800RPM
- 16: 60Hz 460V 52.5kVA 1800RPM
- 17: 60Hz 460V 72.5kVA 1800RPM
- 18: 60Hz 460V 100kVA 1800RPM
- 19: 60Hz 460V 300kVA 1800RPM
- 20: 60Hz 460V 406kVA 1800RPM
- 21: 60Hz 460V 800kVA 1800RPM
- 22: 60Hz 460V 1075kVA 1800RPM
- 23: 60Hz 460V 1588kVA 1800RPM
- 24: 60Hz 460V 2500kVA 1800RPM

Select **No** (default) if you do not want to use a preset model or if you want to modify some of the parameters of a preset model, as described below.

When you select a preset model, the electrical and mechanical parameters in the **Parameters** tab are dimmed. To start from a preset model and then modify the machine parameters:

- 1 Select the preset model you want to initialize the parameters.
- 2 Change the **Preset model** parameter to **No**. This action does not change the machine parameters, but breaks the connection with the preset model.
- 3 Modify the machine parameters as you want.

Mechanical input — Mechanical input

Mechanical power P_m (default) | Speed w | Mechanical rotational port

Whether to represent the mechanical power applied to the shaft or the rotor speed as a Simulink input of the block, or to represent the machine shaft with a Simscape rotational mechanical port.

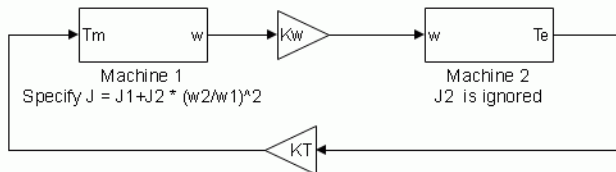
Select **Mechanical power P_m** to specify a mechanical power input, in pu, and expose the **P_m** port. The machine speed is determined by the inertia constant H and by the difference between the mechanical torque T_m , which results from the applied mechanical power P_m , and the internal electromagnetic torque T_e . When the speed is positive, a positive mechanical power signal indicates generator mode and a negative signal indicates motor mode.

Select **Speed w** to specify a speed input in pu and expose the **w** port. The machine speed is imposed and the mechanical part of the model (the inertia constant H) is ignored. Using the speed as the mechanical input allows you to model a mechanical coupling between two machines.

The next figure indicates how to model a stiff shaft interconnection in a motor-generator set where both machines are synchronous machines.

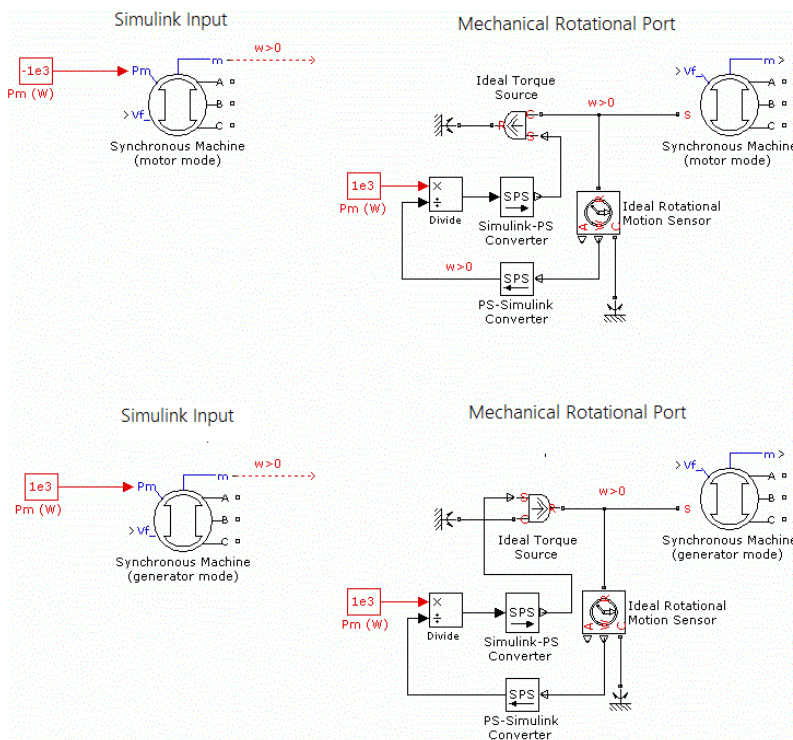
The speed output of machine 1 (the motor) is connected to the speed input of machine 2 (the generator). In this figure, friction torque is ignored in machine 2. Therefore, its electromagnetic torque output **T_e** corresponds to the mechanical torque T_m applied to the shaft of machine 1. The corresponding mechanical input power of machine 1 is computed as $P_m = T_m \cdot w$. The K_w factor takes

into account the speed units of both machines (pu) and the gear box ratio w_2/w_1 . The K_T factor takes into account the torque units of both machines (pu) and the machine ratings. Also, because inertia J_2 is ignored in machine 2, J_2 refers to the speed of machine 1 and must be added to machine 1 inertia, J_1 .



Select **Mechanical rotational port** to expose a Simscape mechanical rotational port, **S**, that allows you to connect of the machine shaft to another machine shaft or to other Simscape blocks that have mechanical rotational ports.

The figure indicates how to connect an Ideal Torque Source block from the Simscape library to the machine shaft port to represent the machine in motor mode or generator mode when the rotor speed is positive.



Rotor type — Rotor type
Salient-pole (default) | Round

Specify the rotor type as either **Salient-pole** or **Round** (cylindrical). The setting affects the number of rotor circuits in the q -axis (damper windings).

Use signal names to identify bus labels — How to identify bus labels
off (default) | on

When this check box is selected, the measurement output uses the signal names to identify the bus labels. Select this option for applications that require bus signal labels to have only alphanumeric characters.

When this check box is cleared, the measurement output uses the signal definition to identify the bus labels. The labels contain nonalphanumeric characters that are incompatible with some Simulink applications.

Parameters

Nominal power, line-to-line voltage, frequency [Pn(VA) Vn(Vrms) fn(Hz)] — Nominal power, line-to-line voltage, and frequency

[6e+04 400 50] (default) | three-element vector

Total three-phase apparent power (VA), RMS line-to-line voltage (pu), frequency (Hz), and field current (pu).

This parameter is identical to the **Nominal power, voltage, frequency, field current [Pn(VA) Vn(Vrms) fn(Hz) ifn(A)]** parameter of the Synchronous Machine SI Fundamental block, except that you do not specify a nominal field current. This value is not required because the Synchronous Machine pu Standard block does not need the transformation ratio. Because rotor quantities are viewed from the stator, they are converted to pu using the stator base quantities derived from the preceding three nominal parameters.

Reactances [Xd Xd' Xd'' Xq Xq'' Xl] (pu) — Reactances

[2.24 0.17 0.12 1.02 0.13 0.08] (default) | six-element vector

d -axis synchronous reactance X_d , transient reactance X_d' , and subtransient reactance X_d'' ; the q -axis synchronous reactance X_q and subtransient reactance X_q'' ; and the leakage reactance X_l (all in pu).

Dependencies

To enable this parameter, set **Rotor type** to Salient-pole.

Reactances [Xd Xd' Xd'' Xq Xq' Xq'' Xl] (pu) — Reactances

[1.81, 0.3, 0.23, 1.76, 0.65, 0.25, 0.15] (default)

d -axis synchronous reactance X_d , transient reactance X_d' , and subtransient reactance X_d'' ; the q -axis synchronous reactance X_q , transient reactance X_q' , and subtransient reactance X_q'' ; and the leakage reactance X_l (all in pu).

Dependencies

To enable this parameter, set **Rotor type** to Round.

d axis — d -axis time constant

Short-circuit (default) | Open-circuit

Time constant for the d axis.

q axis — q -axis time constant

Short-circuit (default) | Open-circuit

Time constant for the q axis.

[Tdo' Tdo'' Tqo''] (s) — Time constants in s
[4.4849 0.0681 0.1] (default) | three-element vector

Time constants, all in s, for the following:

- *d*-axis transient open-circuit (Tdo')
- *d*-axis subtransient open-circuit (Tdo'')
- *q*-axis subtransient open-circuit (Tqo'')

Dependencies

To enable this parameter, set **Rotor type** to Salient-pole, **d axis** to Open-circuit, and **q axis** to Open-circuit.

[Tdo' Tdo'' Tqo' Tqo''] (s) — Time constants in s
[8.0, 0.03, 1.0, 0.07] (default) | four-element vector

Time constants, all in s, for the following:

- *d*-axis transient open-circuit (Tdo')
- *d*-axis subtransient open-circuit (Tdo'')
- *q*-axis transient open-circuit (Tqo')
- *q*-axis subtransient open-circuit (Tqo'')

Dependencies

To enable this parameter, set **Rotor type** to Round, **d axis** to Open-circuit, and **q axis** to Open-circuit.

[Tdo' Tdo'' Tq''] (s) — Time constants in s
[4.49 0.0681 0.0513] (default) | three-element vector

Time constants, all in s, for the following:

- *d*-axis transient open-circuit (Tdo')
- *d*-axis subtransient open-circuit (Tdo'')
- *q*-axis subtransient short-circuit (Tq'')

Dependencies

To enable this parameter, set **Rotor type** to Salient-pole, **d axis** to Open-circuit, and **q axis** to Short-circuit.

[Tdo' Tdo'' Tq' Tq''] (s) — Time constants in s
[8.0, 0.03, 0.3371, 0.0295] (default) | four-element vector

Time constants, all in s, for the following:

- *d*-axis transient open-circuit (Tdo')
- *d*-axis subtransient open-circuit (Tdo'')
- *q*-axis transient short-circuit (Tq')
- *q*-axis subtransient short-circuit (Tq'')

Dependencies

To enable this parameter, set **Rotor type** to Round, **d axis** to Open-circuit, and **q axis** to Short-circuit.

[Td' Td'' Tqo''] (s) — Time constants in s
 [1.01, 0.053, 0.1] (default) | three-element vector

Time constants, all in s, for the following:

- *d*-axis transient short-circuit (Td')
- *d*-axis subtransient short-circuit (Td'')
- *q*-axis subtransient open-circuit (Tqo'')

Dependencies

To enable this parameter, set **Rotor type** to Salient-pole, **d axis** to Short-circuit, and **q axis** to Open-circuit.

[Td' Td'' Tqo' Tqo''] (s) — Time constants in s
 [1.3201, 0.023, 1.0, 0.07] (default) | four-element vector

Time constants, all in s, for the following:

- *d*-axis transient short-circuit (Td')
- *d*-axis subtransient short-circuit (Td'')
- *q*-axis transient open-circuit (Tqo')
- *q*-axis subtransient open-circuit (Tqo'')

Dependencies

To enable this parameter, set **Rotor type** to Round, **d axis** to Short-circuit, and **q axis** to Open-circuit.

[Td' Td'' Tq''] (s) — Time constants in s
 [0.012 0.003 0.003] (default) | three-element vector

Time constants, all in s, for the following:

- *d*-axis transient short-circuit (Td')
- *d*-axis subtransient short-circuit (Td'')
- *q*-axis subtransient short-circuit (Tq'')

Dependencies

To enable this parameter, set **Rotor type** to Salient-pole, **d axis** to Short-circuit, and **q axis** to Short-circuit.

[Td' Td'' Tq' Tq''] (s) — Time constants in s
 [1.3201, 0.0231, 0.3371, 0.0295] (default) | four-element vector

Time constants, all in s, for the following:

- *d*-axis transient short-circuit (Td')

- d -axis subtransient short-circuit (T_d'')
- q -axis transient short-circuit (T_q')
- q -axis subtransient short-circuit (T_q'')

Dependencies

To enable this parameter, set **Rotor type** to Round, **d axis** to Short-circuit, and **q axis** to Short-circuit.

Stator resistance R_s (pu) — Stator resistance

0.037875 (default) | positive scalar

Stator resistance R_s , in pu.

Inertia coefficient, friction factor, pole pairs [$H(s)$ $F(pu)$ $p()$] — Inertia coefficient, friction factor, and pole pairs

[0.1028 0.02056 2] (default) | three-element vector

Inertia constant H (s), which is the ratio of energy stored in the rotor at nominal speed over the nominal power of the machine, friction factor F (pu torque/pu speed), and the number of pole pairs p . Friction torque T_f is proportional to rotor speed ω ($T_f = F\omega$, where all quantities are expressed in pu).

Dependencies

To enable this parameter, set **Mechanical input** to Mechanical power P_m or Mechanical rotational port.

Pole pairs p () — Pole pairs

20 (default) | positive integer

Number of pole pairs of the synchronous machine.

Dependencies

To enable this parameter, set **Mechanical input** to Speed w .

Initial conditions [$\Delta\omega(\%)$ $\theta(\text{deg})$ $i_a, i_b, i_c(\text{pu})$ $\phi_a, \phi_b, \phi_c(\text{deg})$ $V_f(\text{pu})$] — Initial conditions

[0 0 0 0 0 0 0 0 1] (default) | nine-element vector

Initial speed deviation $\Delta\omega$ (percent of nominal speed), electrical angle of the rotor θ_e (degrees), line current magnitudes i_a , i_b , i_c (pu) and phase angles ϕ_a , ϕ_b , ϕ_c (degrees), and the initial field voltage V_f (pu). You can compute these values automatically by using the Load Flow tool or the Machine Initialization tool of the powergui block.

Simulate saturation — Simulate magnetic saturation

on (default) | off

Whether the magnetic saturation of the rotor and stator iron is simulated.

Select this check box to provide the matrix of parameters for simulating the saturation.

Clear this check box to not model saturation in your simulation. In this case, the relationship between the $i_f d$ and V_t is linear (no saturation).

[ifd; vt] (pu) — No-load saturation curve parameters

[0.6404,0.7127,0.8441,0.9214,0.9956,1.082,1.19,1.316,1.457;0.7,0.7698,0.8872,0.9466,0.9969,1.046,1.1,1.151,1.201] (default) | vector

No-load saturation curve parameters. The magnetic saturation of the stator and rotor iron is modeled by a piecewise linear relationship specifying points on the no-load saturation curve. The first row of this matrix contains the values of field currents (in pu). The second row contains values of corresponding terminal voltages (in pu). The first point (first column of the matrix) must be different from [0,0]. This point corresponds to the point where the effect of saturation begins. For saturation, the nominal field current and nominal RMS line-to-line voltage are the base values for the field current and terminal voltage, respectively.

Click **Plot** to view the no-load saturation curve.

Dependencies

To enable this parameter, select **Simulate saturation**.

Advanced

To enable the **Advanced** tab, in the powergui block, set **Simulation type** to Discrete and clear **Automatically handle discrete solver**.

Sample time (-1 for inherited) — Sample time for block

-1 (default) | scalar

Sample time used by the block. To inherit the sample time specified in the powergui block, set this parameter to -1.

Discrete solver model — Integration method

Trapezoidal non iterative (default) | Trapezoidal iterative (alg. loop) | Trapezoidal robust | Backward Euler robust

Integration method used by the block when the **Solver type** parameter of the powergui block is set to Discrete.

The **Discrete solver model** is automatically set to Trapezoidal robust when you select the **Automatically handle Discrete solver and Advanced tab solver settings of blocks** parameter of the powergui block.

The Trapezoidal non iterative and Trapezoidal iterative (alg. loop) methods are no longer recommended for discretizing the Synchronous Machine pu Standard block. Trapezoidal non iterative requires you to add non-negligible shunt loads at the machine terminals to guarantee simulation stability, and Trapezoidal iterative (alg. loop) may fail to converge and cause the simulation to stop when the number of machines increases in the model.

The Trapezoidal robust and Backward Euler robust methods allow you to eliminate the need to use parasitic loads and simulate a machine without loads. To eliminate topological errors of machines connected to an inductive circuit (for example, a circuit breaker connected in series with the machine); the machine models a negligible internal load of 0.01% of nominal power.

The Trapezoidal robust method is slightly more accurate than the Backward Euler robust method, especially when the model is simulated at larger sample times. The Trapezoidal robust method may produce slight damped numerical oscillations on machine voltage at no-load conditions, while the Backward Euler robust method prevents oscillations and maintains good accuracy.

For more information on what method to use in your application, see *Simulating Discretized Electrical Systems*.

Load Flow

The load flow parameters are used to define block parameters for use with the Load Flow tool of the powergui block. These load flow parameters are used for model initialization only. They have no impact on the block model or on the simulation performance.

Generator type — Machine generator type

PV (default) | swing | PQ

Generator type of the machine. Choice are:

- **swing** — Implements a generator controlling magnitude and phase angle of its terminal voltage. The reference voltage magnitude and angle are specified by the **Swing bus or PV bus voltage** and **Swing bus voltage angle** parameters of the Load Flow Bus block connected to the machine terminals.
- **PV** — Implements a generator controlling its output active power P and voltage magnitude V . P is specified by the **Active power generation P (W)** parameter of the block. V is specified by the **Swing bus or PV bus voltage** parameter of the Load Flow Bus block connected to the machine terminals. You can control the minimum and maximum reactive power generated by the block by using the **Minimum reactive power Qmin (var)** and **Maximum reactive power Qmax (var)** parameters.
- **PQ** — Implements a generator controlling its output active power P and reactive power Q . P and Q are specified by the **Active power generation P (W)** and **Reactive power generation Q (var)** parameters of the block, respectively.

Active power generation P (W) — Active power generated by machine

0 (default) | scalar

Active power that you want generated by the machine, in watts. When the machine operates in motor mode, you specify a negative value.

Dependencies

To enable this parameter, set **Generator type** to PV or PQ.

Reactive power generation Q (var) — Reactive power generated by machine

0 (default) | scalar

Reactive power that you want generated by the machine, in vars. A negative value indicates that the reactive power is absorbed by the machine.

Dependencies

To enable this parameter, set **Generator type** to PQ.

Minimum reactive power Qmin (var) — Minimum reactive power generated by machine

-inf (default) | scalar

Minimum reactive power that can be generated by the machine while keeping the terminal voltage at its reference value. This reference voltage is specified by the **Swing bus or PV bus voltage** parameter of the Load Flow Bus block connected to the machine terminals. The default value is -inf, which means that there is no lower limit on the reactive power output.

Dependencies

To enable this parameter, set **Generator type** to PV.

Maximum reactive power Qmax (var) — Maximum reactive power generated by machine

`inf` (default) | scalar

Maximum reactive power that can be generated by the machine while keeping the terminal voltage at its reference value. This reference voltage is specified by the **Swing bus or PV bus voltage** parameter of the Load Flow Bus block connected to the machine terminals. The default value is `inf`, which means that there is no upper limit on the reactive power output.

Dependencies

To enable this parameter, set **Generator type** to PV.

Version History

Introduced before R2006a

References

- [1] Canay, I.M. "Causes of Discrepancies on Calculation of Rotor Quantities and Exact Equivalent Diagrams of the Synchronous Machine." *IEEE Transactions on Power Apparatus and Systems*. PAS-88, no. 7 (1969): 1114–1120.
- [2] Moeini, A., et al. "Synchronous Machine Stability model, an Update to IEEE Std 1110-2002 Data Translation Technique." *IEEE standards panel sessions*. 2018.
- [3] *IEEE Guide for Synchronous Generator Modeling Practices and Applications in Power System Stability Analyses*. IEEE Std 1110-2002 (Revision of IEEE Std 1110-1991 [2003]): 1–72.
- [4] Krause, P.C. *Analysis of Electric Machinery*. Section 12.5. New York: McGraw-Hill, 1986.
- [5] Kundur, P. *Power System Stability and Control*. New York, McGraw-Hill, 1994.

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

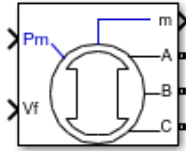
Excitation System | Hydraulic Turbine and Governor | powergui | Simplified Synchronous Machine | Steam Turbine and Governor | Synchronous Machine SI Fundamental | Synchronous Machine pu Fundamental

Topics

"Per-Unit System of Units" on page A-2

Synchronous Machine SI Fundamental

Model dynamics of three-phase round-rotor or salient-pole synchronous machine using fundamental parameters in SI units



Libraries:

Simscape / Electrical / Specialized Power Systems / Electrical Machines

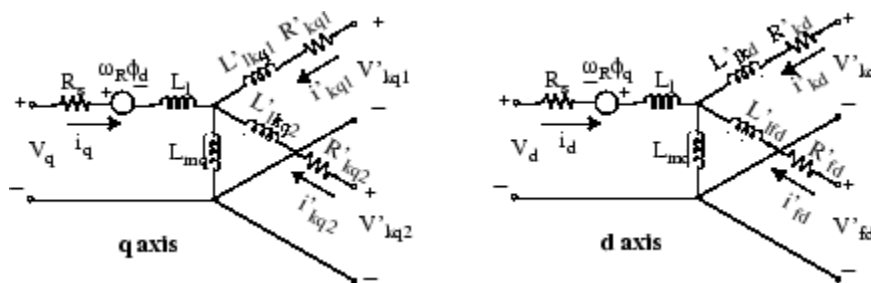
Description

The Synchronous Machine SI Fundamental block models a synchronous machine in generator or motor mode using fundamental parameters in SI units. The operating mode is dictated by the sign of the mechanical power (positive for generator mode or negative for motor mode). The electrical part of the machine is represented by a sixth-order state-space model and the mechanical part is the same as in the Simplified Synchronous Machine block.

The model takes into account the dynamics of the stator, field, and damper windings. The equivalent circuit of the model is represented in the rotor reference frame (qd frame). Stator windings are connected in wye to an internal neutral point. All rotor parameters and electrical quantities are viewed from the stator and identified by primed variables. The subscripts are:

- d, q — d - and q -axis quantity
- R, s — Rotor and stator quantity
- l, m — Leakage and magnetizing inductance
- f, k — Field and damper winding quantity

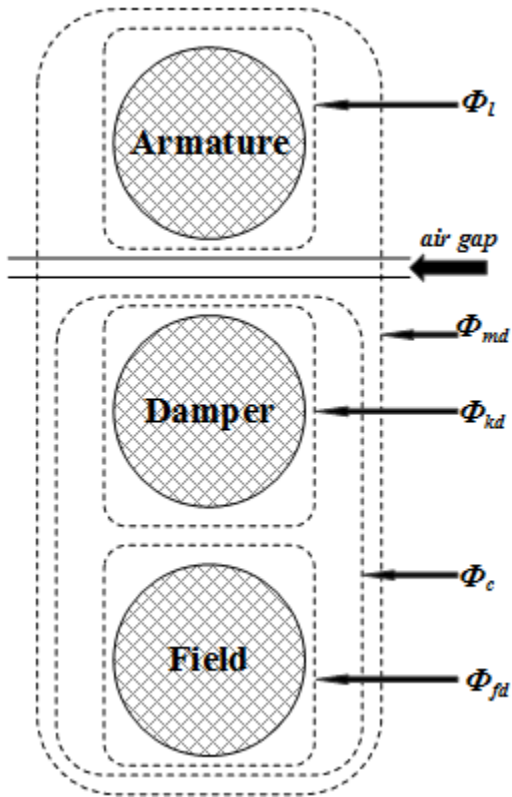
The electrical model of the machine is shown in these diagrams.



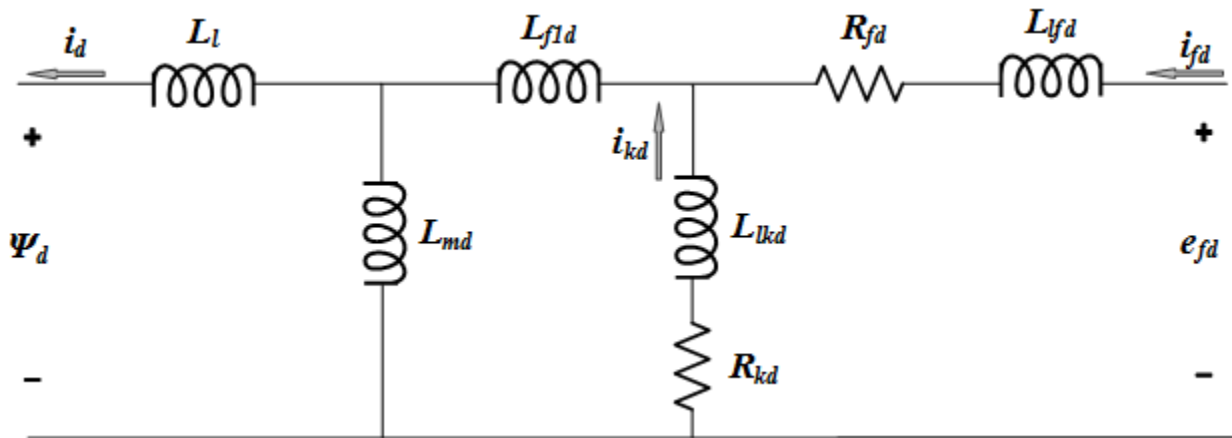
Dynamic Model with Unequal Mutual Inductance

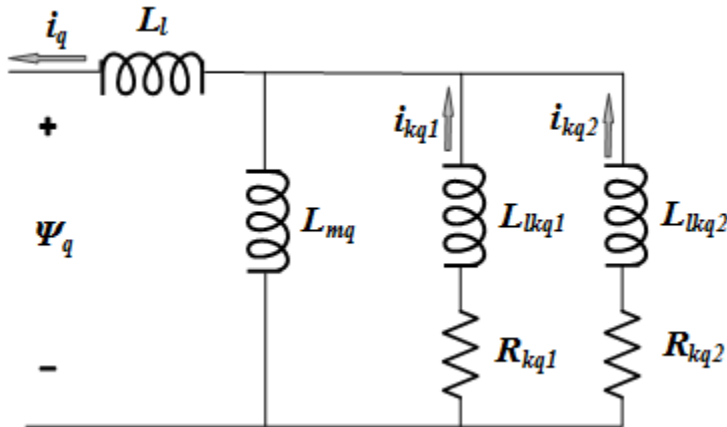
The conventional theory of synchronous machine modeling for stability analysis assumes that the mutual inductances between the armature, damper, and field on direct-axis windings are identical. Generally, damper windings are near the air gap, and as a result the flux linking damper circuits are almost equal to the flux linking armature. This hypothesis produces acceptable outcomes for a wide range of stability studies, especially those on the side of the network. However, when it comes to field current studies, there is considerable error. The equivalent circuit dynamic model of a synchronous

machine may include an additional inductance representing the difference between field-damper and field-armature mutual inductances on the D -axis [1]. This inductance is typically called Canay inductance. Canay inductance corresponds to the leakage flux, Φ_c , in the following figure and is interpreted as a corrective element in an equivalent model that may have a negative value [2].



The IEEE standard 1110-2002 [3] presents the direct and quadratic axes of the synchronous machine dynamic model as shown in the diagrams.





The relevant equations are:

$$V_d = -i_d R_s - \omega \psi_q + \frac{d\psi_d}{dt}$$

$$V_q = -i_q R_s + \omega \psi_d + \frac{d\psi_q}{dt}$$

$$V_0 = -i_0 R_0 + \frac{d\psi_0}{dt}$$

$$V_{fd} = \frac{d\psi_{fd}}{dt} + R_{fd} i_{fd}$$

$$0 = \frac{d\psi_{kd}}{dt} + R_{kd} i_{kd}$$

$$0 = \frac{d\psi_{kq1}}{dt} + R_{kq1} i_{kq1}$$

$$0 = \frac{d\psi_{kq2}}{dt} + R_{kq2} i_{kq2}$$

$$\begin{bmatrix} \psi_d \\ \psi_{kd} \\ \psi_{fd} \end{bmatrix} = \begin{bmatrix} L_{md} + L_l & L_{md} & L_{md} \\ L_{md} & L_{lkd} + L_{f1d} + L_{md} & L_{f1d} + L_{md} \\ L_{md} & L_{f1d} + L_{md} & L_{lfd} + L_{f1d} + L_{md} \end{bmatrix} \begin{bmatrix} -i_d \\ i_{kd} \\ i_{fd} \end{bmatrix}$$

$$\begin{bmatrix} \psi_q \\ \psi_{kq1} \\ \psi_{kq2} \end{bmatrix} = \begin{bmatrix} L_{mq} + L_l & L_{mq} & L_{mq} \\ L_{mq} & L_{mq} + L_{kq1} & L_{mq} \\ L_{mq} & L_{mq} & L_{mq} + L_{kq2} \end{bmatrix} \begin{bmatrix} -i_q \\ i_{kq1} \\ i_{kq2} \end{bmatrix}$$

Base Values, Transformation Ratio, and Rotor Parameters Referred to the Stator

The Synchronous Machine SI Fundamental block allows you to specify the fundamental parameters of a synchronous machine. You enter field and damper winding parameters (resistances, leakage inductances, and mutual inductances) in SI (Ω , H). The RL parameters of the field and damper windings are not the actual field RL values of the machine but the RL values referred to the stator.

Stator Base Values

This table displays the stator base values.

$$V_{sbase} = \frac{V_n \sqrt{2}}{\sqrt{3}} \quad \text{Base stator voltage} = \text{peak nominal line-to-neutral voltage (V)}$$

$$I_{sbase} = \frac{P_n \sqrt{2}}{V_n \sqrt{3}} \quad \text{Base stator current (A)}$$

$$Z_{sbase} = \frac{V_{sbase}}{I_{sbase}} = \frac{V_n^2}{P_n} \quad \text{Base stator impedance } (\Omega)$$

$$\omega_{base} = 2\pi f_n \quad \text{Base angular frequency (rad/s)}$$

$$L_{sbase} = \frac{Z_{sbase}}{\omega_{base}} \quad \text{Base stator inductance (H)}$$

where:

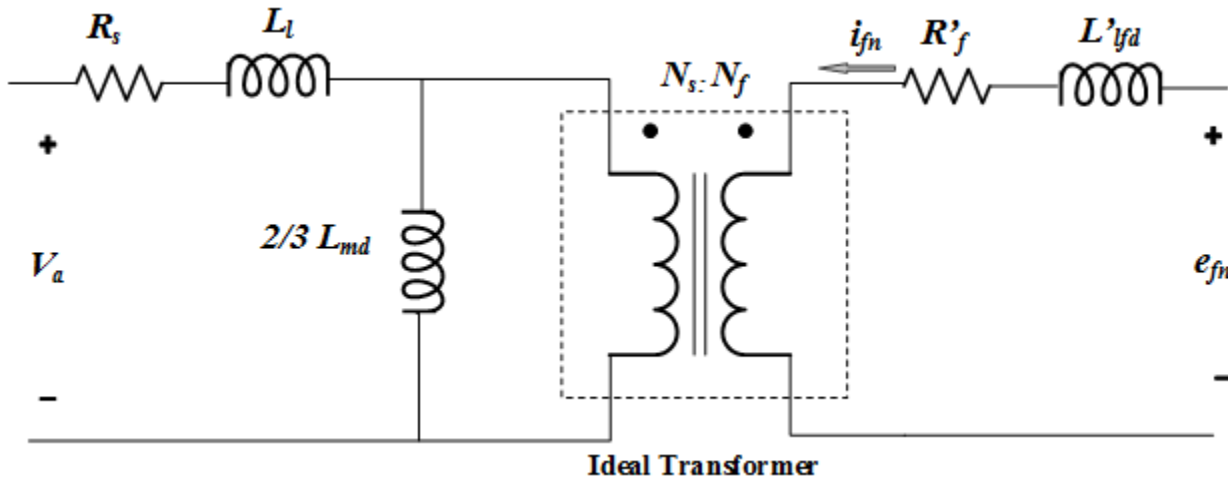
- P_n is the three-phase nominal power (VA).
- V_n is the nominal line-to-line voltage (Vrms).
- f_n is the nominal frequency (Hz).
- i_{fn} is the nominal field current producing nominal stator voltage at no load (A).

The stator parameters to specify in the block are:

R_s	Stator resistance per phase (Ω)
L_l	Stator leakage inductance (H)
L_{md}	Direct-axis magnetizing inductance viewed from stator (H)
L_{mq}	Quadrature-axis magnetizing inductance viewed from stator (H)

Stator/Field Transformation Ratio

This figure shows one phase of the stator winding coupled with the field winding.



In the diagram:

- N_s and N_f are the equivalent number of sinusoidally distributed turns of the stator winding and of the field winding, respectively.
- R_s and L_l are stator resistance and leakage inductance.
- R_f' and L_{lfd}' are field resistance and leakage inductance.

When the three stator windings are energized with a three-phase positive-sequence voltage and the field winding is open, the stator magnetizing inductance is L_{md} . However, when only one phase is energized and the field winding is open, the magnetizing inductance is $2/3 L_{md}$, as shown in the figure.

At no load, when the field winding is rotating at nominal speed and bears the nominal field DC current i_{fn} , the amplitude of AC voltage (peak value) induced on one phase of the stator is V_{sbase} .

The maximum mutual inductance between one stator winding and the field winding is obtained when the two windings are aligned and is given by:

$$L_{sfd} = \frac{2}{3} L_{md} \frac{N_f}{N_s} = \frac{V_{sbase}}{i_{fn} \omega_{base}}$$

from which we deduce the transformation ratio:

$$\frac{N_s}{N_f} = \frac{2}{3} L_{md} \frac{i_{fn} \omega_{base}}{V_{sbase}}.$$

The transformation ratio can be also expressed as:

$$\frac{N_s}{N_f} = \frac{2}{3} \frac{I_{fbase}}{I_{sbase}},$$

where I_{fbase} is the base field current. The table shows how to compute field base values.

$$\begin{aligned} I_{fbase} &= i_{fn} L_{md\_pu} && \text{Base field current (A)} \\ V_{fbase} &= \frac{P_n}{I_{fbase}} && \text{Base field voltage (V)} \\ Z_{fbase} &= \frac{V_{fbase}}{I_{fbase}} && \text{Base field impedance } (\Omega) \\ L_{fbase} &= \frac{Z_{fbase}}{\omega_{base}} && \text{Base field inductance (H)} \end{aligned}$$

The actual field parameters are:

- R_f' is field resistance (Ω)
- L_{lfd}' is field leakage inductance (H)

Field Voltage, Current, and RL Values Referred to the Stator

You can specify field parameters for the field resistance and leakage inductance referred to the stator (R_f , L_{lfd}).

If the nominal field current i_{fn} is known, the transformation ratio N_s / N_f is calculated using the same equation as for the stator/field transformation ratio.

According to Krause [4], the field voltage and current referred to the stator (V_f , I_f) are deduced from actual V_f' , I_f' values as follows:

$$V_f = \frac{N_s}{N_f} V_f'$$

$$I_f = \frac{2}{3} \frac{N_f}{N_s} I_f'$$

When the actual field resistance R_f' and leakage inductances L_{lfd}' (seen from the rotor) are known, the values referred to the stator are:

$$R_f = \frac{3}{2} R_f' \left(\frac{N_s}{N_f} \right)^2$$

$$L_{lfd} = \frac{3}{2} L_{lfd}' \left(\frac{N_s}{N_f} \right)^2$$

When the nominal field current is not known, and if the pu values of field resistance and leakage inductances are known (R_{f_pu} , L_{lfd_pu}), the corresponding SI values referred to the stator are computed as follows:

$$R_f = R_{f\_pu} \times Z_{sbase}$$

$$L_{lfd} = L_{lfd\_pu} \times L_{sbase}$$

The same conversions are used for RL parameters of damper windings.

Field Voltage and Current

When you specify the nominal field current, the signal at the **Vf** input port corresponds to the actual field voltage, as in real life. The field current returned by the measurement output also corresponds to the actual field current I_f .

The nominal field voltage producing nominal stator voltage at no load is given by:

$$e_{fn} = R_f' \times i_{fn}$$

When you do not specify the nominal field current, the signal at the **Vf** input port corresponds to the actual field voltage referred to the stator. In this case, the nominal field voltage referred to the stator producing nominal stator voltage at no load is:

$$e_{fn}(\text{stator side}) = \frac{R_{f\_pu}}{L_{md\_pu}} V_{sbase} = \frac{R_f}{L_{md} \omega_{base}} V_{sbase}.$$

The field current returned by the measurement output is the field current referred to the stator. The nominal field current referred to the stator is:

$$i_{fn}(\text{stator side}) = \frac{I_{sbase}}{L_{md\_pu}} = \frac{I_{sbase}}{I_{fbase}} i_{fn}.$$

Examples

Model 555 MVA, 24 kV, 60 Hz, 3600 rpm Synchronous Generator

The `power_SM_Fundamental` example uses the Synchronous Machine SI Fundamental block and Synchronous Machine pu Fundamental block to model a 555 MVA, 24 kV, 60 Hz, 3600 rpm synchronous generator. It shows how to specify SI and pu parameters and explains how to compute field and damper rotor winding parameters that are referred to the stator. In addition to the field winding, the round rotor of this machine has three damper windings: one damper in the direct axis and two dampers in the quadrature axis.

Three circuits simulate the same synchronous machine:

- Circuit 1: Fundamental parameters are specified in SI. The nominal field current is specified ($ifn = 1300$ A).
- Circuit 2: Fundamental parameters are specified in SI. The nominal field current is not specified ($ifn = 0$).
- Circuit 3: Fundamental parameters are specified in pu.

Machine parameters are taken from an example in Kundur [5].

Look at the **Model Properties/PreLoad Fcn** callback section of the model to see machine specifications and the computation of stator and field bases, RL rotor parameters referred to the stator, transformation ratio, and nominal field voltage and current.

The machines initially operate in steady state at virtually no load (load = 0.1% of nominal power) with constant field voltage and mechanical power. A phase-to-phase six-cycle fault is applied at $t = 0.1$ sec. The Scope block shows the comparison between line-to-line AB voltage, phase A stator current, and field current of the three machines.

To simulate the discrete model, in the `powergui` block, set **Simulation type** to **Discrete**. The model is discretized with a sample time $T_s = 50$ μ s. To obtain a stable model with such a small load (0.1% of nominal power), in the **Advanced** tab of each Synchronous Machine block, set **Discrete solver model** to **Trapezoidal iterative (alg. loop)**.

Use Synchronous Machine Block in Motor Mode

The `power_syncmachine` example uses the Synchronous Machine block in motor mode. The simulated system consists of a 150 HP (112 kVA), 762 V industrial-grade synchronous motor connected to a network with a 10 MVA short-circuit level. The machine is initialized for an output electrical power of -50 kW (negative value for motor mode), corresponding to a mechanical power of -48.9 kW. The corresponding values of mechanical power and field voltage are specified by the Pm Step block and the Vf Constant block. The Pm Step block applies a sudden increase of mechanical power from -48.9 kW to -60 kW at time $t = 0.1$ s.

Run the simulation.

After the load has increased from 48.9 kW to 60 kW at $t = 0.1$ s, the machine speed oscillates before stabilizing to 1800 rpm. The load angle (angle between terminal voltage and internal voltage) increases from -21 degrees to -53 degrees.

Assumptions and Limitations

In discrete systems, when you set the **Discrete solver model** parameter of a Synchronous Machine block to **Trapezoidal non iterative**, you might have to connect a small parasitic resistive load

at the machine terminals to avoid numerical oscillations. Large sample times require larger loads. The minimum resistive load is proportional to the sample time. As a rule of thumb, remember that with a 25 μ s time step on a 60 Hz system, the minimum load is approximately 2.5% of the machine nominal power. For example, a 200 MVA synchronous machine in a power system discretized with a 50 μ s sample time requires approximately 5% of resistive load or 10 MW. If the sample time is reduced to 20 μ s, a resistive load of 4 MW should be sufficient.

However, if you set the **Discrete solver model** parameter of a Synchronous Machine block to **Trapezoidal iterative (alg. loop)**, you can use a negligible parasitic load (below 0.1% of nominal power) while preserving numerical stability. This iterative model produces an algebraic loop and results in slower simulation speed.

Ports

Input

Pm — Mechanical power
scalar

Mechanical power at the machine's shaft, in W. In generator mode, this input can be a positive constant or function or the output of a prime mover block (see the Hydraulic Turbine and Governor or Steam Turbine and Governor blocks). In motor mode, this input is usually a negative constant or function.

Dependencies

To enable this port, set the **Mechanical input** parameter (**Configuration** tab) to Mechanical power Pm.

w — Machine speed
scalar

Machine speed, in rad/s.

Dependencies

To enable this port, set the **Mechanical input** parameter (**Configuration** tab) to Speed w.

Vf — Field voltage
scalar

Field voltage. This voltage can be supplied by a voltage regulator in generator mode (see the Excitation System block). In motor mode, this value of this input is usually a constant.

The field voltage at this port must be entered in volts DC if the nominal field current I_{fn} is specified in the **Nominal power, voltage, frequency, field current [Pn(VA) Vn(Vrms) fn(Hz) ifn(A)]** parameter, or in volts referred to stator if I_{fn} is not specified. To obtain the value of the field voltage producing a 1 pu stator voltage, select the **Display nominal field current and voltage producing 1 pu stator voltage** check box in the **Advanced** tab.

Output

m — Measurement signal
24-element vector

Measurement signals, returned as a 24-element vector. You can demultiplex these signals by using the Bus Selector block.

Name	Definition	Units
ias	Stator current is_a	A
ibs	Stator current is_b	A
ics	Stator current is_c	A
iq	Stator current iq	A
id	Stator current id	A
ifd	Field current ifd	A
ikq1	Damper winding current ikq1	A
Ikq2	Damper winding current ikq2	A
ikd	Damper winding current ikd	A
phimq	Mutual flux phimq	V.s
phimd	Mutual flux phimd	V.s
vq	Stator voltage vq	V
vd	Stator voltage vd	V
lmq	Lmq saturated inductance	H
lmd	Lmd saturated inductance	H
dtheta	Rotor angle deviation d_theta	rad
w	Rotor speed wm	rad/s
Pe	Electrical power Pe	W
dw	Rotor speed deviation dw	pu
theta	Rotor mechanical angle theta	degree
Te	Electromagnetic torque Te	N.m
delta	Load angle delta	degree
Pe0	Output active power Peo	W
Qe0	Output reactive power Qeo	VAR

The angular deviation of the rotor angle theta represents the electrical angle giving the instantaneous position of the rotor with respect to a common reference rotating at synchronous speed. This angle is useful in stability studies to measure the relative positions of the rotors of different machines in a network. The positions of the rotors are then measured with respect to the theta position of a given machine chosen as a reference.

Conserving

A — Phase A electric terminal
specialized electrical

Specialized electrical conserving port associated with the phase A electrical terminal.

B — Phase B electric terminal
specialized electrical

Specialized electrical conserving port associated with the phase B electrical terminal.

C — Phase C electric terminal
specialized electrical

Specialized electrical conserving port associated with the phase C electrical terminal.

S — Machine rotor shaft
mechanical

Mechanical rotational conserving port associated with the machine rotor shaft.

Dependencies

To enable this port, set the **Mechanical input** parameter (**Configuration** tab) to Mechanical rotational port.

Parameters

In power system analysis software, the synchronous machine equations are commonly solved using the direct-quadrature-zero transformation technique. The RL parameters of field and damper windings are not the actual field RL values of the machine but the RL values referred to the stator. The rotor side parameters are transferred to stator side and normalized before entering the final processing.

Configuration

Preset model — Electrical and mechanical parameters

No (default) | 01: 50Hz 400V 8.1kVA 1500RPM | 02: 50Hz 400V 16kVA 1500RPM | ...

Set of predetermined electrical and mechanical parameters for various synchronous machine ratings of power (kVA), phase-to-phase voltage (V), frequency (Hz), and rated speed (rpm).

Select one of the preset models to load the corresponding electrical and mechanical parameters. Choices are:

- 01: 50Hz 400V 8.1kVA 1500RPM
- 02: 50Hz 400V 16kVA 1500RPM
- 03: 50Hz 400V 31.3kVA 1500RPM
- 04: 50Hz 400V 42.5kVA 1500RPM
- 05: 50Hz 400V 60kVA 1500RPM
- 06: 50Hz 400V 85kVA 1500RPM
- 07: 50Hz 400V 250kVA 1500RPM
- 08: 50Hz 400V 325kVA 1500RPM
- 09: 50Hz 400V 670kVA 1500RPM
- 10: 50Hz 400V 910kVA 1500RPM
- 11: 50Hz 400V 1320kVA 1500RPM
- 12: 50Hz 400V 2000kVA 1500RPM
- 13: 60Hz 460V 10.2kVA 1800RPM
- 14: 60Hz 460V 20kVA 1800RPM
- 15: 60Hz 460V 37.5kVA 1800RPM

- 16: 60Hz 460V 52.5kVA 1800RPM
- 17: 60Hz 460V 72.5kVA 1800RPM
- 18: 60Hz 460V 100kVA 1800RPM
- 19: 60Hz 460V 300kVA 1800RPM
- 20: 60Hz 460V 406kVA 1800RPM
- 21: 60Hz 460V 800kVA 1800RPM
- 22: 60Hz 460V 1075kVA 1800RPM
- 23: 60Hz 460V 1588kVA 1800RPM
- 24: 60Hz 460V 2500kVA 1800RPM

Select **No** (default) if you do not want to use a preset model or if you want to modify some of the parameters of a preset model, as described below.

When you select a preset model, the electrical and mechanical parameters in the **Parameters** tab are dimmed). To start from a preset model and then modify the machine parameters:

- 1 Select the preset model for which you want to initialize the parameters.
- 2 Change the **Preset model** parameter to **No**. This action does not change the machine parameters, but breaks the connection with the preset model.
- 3 Modify the machine parameters as you want.

Mechanical input — Mechanical input

Mechanical power P_m (default) | Speed w | Mechanical rotational port

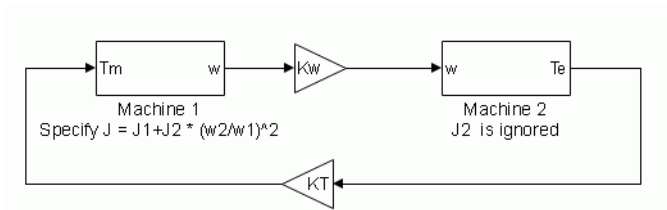
Whether to represent the mechanical power applied to the shaft or the rotor speed as a Simulink input of the block, or to represent the machine shaft with a Simscape rotational mechanical port.

Select **Mechanical power P_m** to specify a mechanical power input, in W, and expose the **P_m** port. The machine speed is determined by the machine inertia J and by the difference between the mechanical torque T_m , which results from the applied mechanical power P_m , and the internal electromagnetic torque T_e . When the speed is positive, a positive mechanical power signal indicates generator mode and a negative signal indicates motor mode.

Select **Speed w** to specify a speed input in rad/s and expose the **w** port. The machine speed is imposed and the mechanical part of the model (the machine inertia J) is ignored. Using the speed as the mechanical input allows you to model a mechanical coupling between two machines.

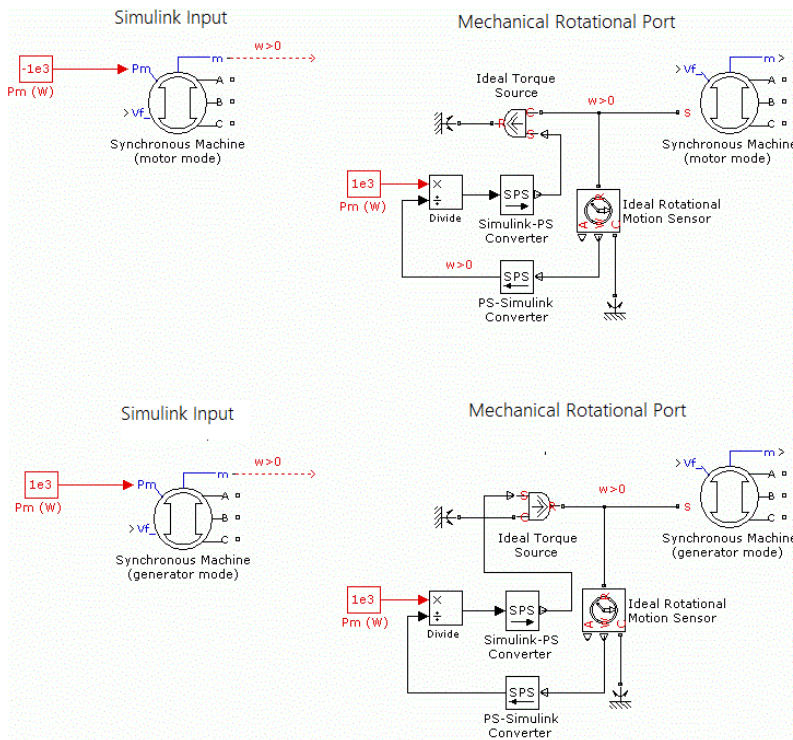
The next figure indicates how to model a stiff shaft interconnection in a motor-generator set where both machines are synchronous machines.

The speed output of machine 1 (the motor) is connected to the speed input of machine 2 (the generator). In this figure, friction torque is ignored in machine 2. Therefore, its electromagnetic torque output **T_e** corresponds to the mechanical torque T_m applied to the shaft of machine 1. The corresponding mechanical input power of machine 1 is computed as $P_m = T_m * w$. The K_w factor takes into account the speed units of both machines (rad/s) and the gear box ratio w_2/w_1 . The K_T factor takes into account the torque units of both machines (N.m) and the machine ratings. Also, because inertia J_2 is ignored in machine 2, J_2 refers to the speed of machine 1 and must be added to machine 1 inertia J_1 .



Select **Mechanical rotational port** to expose a Simscape mechanical rotational port, **S**, that allows you to connect of the machine shaft to another machine shaft or to other Simscape blocks that have mechanical rotational ports.

The figure indicates how to connect an Ideal Torque Source block from the Simscape library to the machine shaft port to represent the machine in motor mode or generator mode when the rotor speed is positive.



Rotor type — Rotor type

Salient-pole (default) | Round

Specify the rotor type as either **Salient-pole** or **Round** (cylindrical). The setting affects the number of rotor circuits in the q -axis (damper windings).

Use signal names to identify bus labels — How to identify bus labels

off (default) | on

When this check box is selected, the measurement output uses the signal names to identify the bus labels. Select this option for applications that require bus signal labels to have only alphanumeric characters.

When this check box is cleared, the measurement output uses the signal definition to identify the bus labels. The labels contain nonalphanumeric characters that are incompatible with some Simulink applications.

Parameters

Nominal power, voltage, frequency, field current [$P_n(VA)$ $V_n(V_{rms})$ $f_n(Hz)$ $if_n(A)$] — Nominal power, voltage, frequency, and field current

[187E6 13800 60 1087] (default) | four-element vector

Total three-phase apparent power P_n (VA), RMS line-to-line voltage V_n (V), frequency f_n (Hz), and field current if_n (A).

The nominal field current is the current that produces the nominal terminal voltage under no-load conditions. This model was developed with all quantities viewed from the stator, as explained in Krause [4]. The nominal field current makes it possible to compute the transformation ratio of the machine, which allows you to apply the field voltage viewed from the rotor, as in real life. It also allows the field current, which is a variable in the output vector of the model, to be viewed from the rotor.

If the value of the nominal field current is not known, you must enter 0 or leave it blank. Because the transformation ratio cannot be determined in this case, you have to apply the field voltage as viewed from the stator. The field current in the output vector is also viewed from the stator.

Stator [$R_s(\text{ohm})$ $L_l, L_{md}, L_{mq}(H)$ ($L_c(H)$)] — Stator resistance and inductances

[2.9069E-03 3.0892E-04 3.2164E-03 9.7153E-04] (default) | four-element vector

Resistance R_s (Ω), leakage inductance L_l (H), d -axis and q -axis magnetizing inductances L_{md} (H) and L_{mq} (H), and Canay inductance. If the value of the Canay inductance is not known, you must enter 0 or leave it blank.

Field [$R_f'(\text{ohm})$ $L_{lfd}'(H)$] — Field resistance and leakage inductance

[5.9013E-04 3.0712E-04] (default) | two-element vector

Field resistance R_f (Ω) and leakage inductance L_{lfd} (H), both referred to the stator.

Dampers [R_{kd}', L_{lkd}' R_{kq1}', L_{lkq1}'] ($R=\text{ohm}, L=H$) — Resistance and leakage inductance

[1.1900E-02 4.9076E-04 2.0081E-02 1.0365E-03] (default) | four-element vector

d -axis resistance R_{kd} (Ω) and leakage inductance L_{lkd} (H), and q -axis resistance R_{kq1} (Ω) and leakage inductance L_{lkq1} (H). All of these values are referred to the stator.

Dependencies

To enable this parameter, set **Rotor type** to Salient-pole.

Dampers [R_{kd}', L_{lkd}' R_{kq1}', L_{lkq1}' R_{kq2}', L_{lkq2}'] ($R=\text{ohm}, L=H$) — Resistance and leakage inductance

[0.0664, 1.387e-3, 0.0292, 6.896e-4, 7.907e-3, 2.477e-3] (default) | six-element vector

d -axis resistance R_{kd} (Ω) and leakage inductance L_{lkd} (H), q -axis resistance R_{kq1} (Ω) and leakage inductance L_{lkq1} (H), and q -axis resistance R_{kq2} (Ω) and leakage inductance L_{lkq2} (H). All of these values are referred to the stator.

Dependencies

To enable this parameter, set **Rotor type** to Round.

Inertia, friction factor, pole pairs [$J(\text{kg.m}^2)$ $F(\text{N.m.s})$ $p()$] — Inertia, friction factor, and pole pairs

[3.895e6 0 20] (default) | three-element vector

Inertia coefficient J (kg.m^2), friction factor F (N.m.s), and the number of pole pairs p . The friction torque T_f is proportional to the rotor speed ω ($T_f = F\omega$.) T_f is expressed in N.m, F in N.m.s , and ω in rad/s.

Dependencies

To enable this parameter, set **Mechanical input** to Mechanical power P_m or Mechanical rotational port.

Pole pairs — Number of pole pairs

20 (default) | positive integer

Number of pole pairs of the synchronous machine.

Dependencies

To enable this parameter, set **Mechanical input** to Speed w .

Initial conditions [$\Delta\omega(\%)$ $\theta_e(\text{deg})$ $i_a, i_b, i_c(\text{A})$ $\phi_a, \phi_b, \phi_c(\text{deg})$ $V_f(\text{V})$] — Initial conditions

[0 0 0 0 0 0 0 0 70.3192] (default) | nine-element vector

Initial speed deviation $\Delta\omega$ (percent of nominal speed), electrical angle of the rotor θ_e (degrees), line current magnitudes i_a , i_b , i_c (A) and phase angles ϕ_a , ϕ_b , ϕ_c (degrees), and the initial field voltage V_f (V). You can compute these values automatically by using the Load Flow tool or the Machine Initialization tool of the powergui block.

You can specify the initial field voltage in one of two ways. If you know the nominal field current (first line, last parameter), enter the initial field voltage in volts DC referred to the rotor. Otherwise, enter a zero as the nominal field current and specify the initial field voltage in volts DC referred to the stator. You can determine the nominal field voltage viewed from the stator by selecting the **Display nominal field current and voltage producing 1 pu stator voltage** check box on the **Advanced** tab.

Simulate saturation — Simulate magnetic saturation

off (default) | on

Whether magnetic saturation of the rotor and stator iron is simulated.

Select this check box to provide the matrix of parameters for simulating the saturation.

Clear this check box to not model saturation in your simulation. In this case, the relationship between the $i_f d$ and V_t is linear (no saturation).

[$i_{fd}(\text{A})$; $V_t(\text{VLL rms})$] — No-load saturation curve parameters

[695.64, 774.7, 917.5, 1001.6, 1082.2, 1175.9, 1293.6, 1430.2, 1583.7; 9660, 10623, 12243, 13063, 13757, 14437, 15180, 15890, 16567] (default) | vector

No-load saturation curve parameters. The magnetic saturation of the stator and rotor iron is modeled by a piecewise linear relationship specifying points on the no-load saturation curve. The first row of

this matrix contains the values of field currents. The second row contains values of corresponding terminal voltages. The first point (first column of the matrix) must be different from $[0, 0]$. This point corresponds to the point where the effect of saturation begins. For saturation, the nominal field current and nominal RMS line-to-line voltage are the base values for the field current and terminal voltage, respectively.

Click **Plot** to view the no-load saturation curve.

Dependencies

To enable this parameter, select **Simulate saturation**.

Advanced

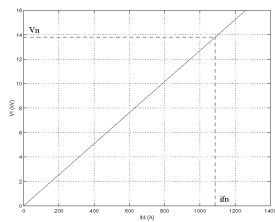
To enable the **Advanced** tab, in the powergui block, set **Simulation type** to Discrete and clear **Automatically handle discrete solver**.

Display nominal field current and voltage producing 1 pu stator voltage — Determine nominal field current and voltage

off (default) | on

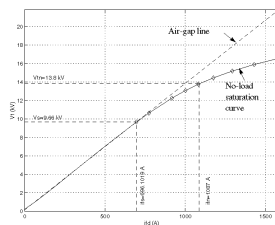
Select to determine the nominal field current and voltage viewed from the stator.

For example, without saturation, the figure shows a typical curve. *ifn* is 1087 A and *Vn* is 13800 V RMS line-to-line, which is also 11268 V peak line-to-neutral.



The saturation is modeled as a piecewise linear saturation curve by using two lookup tables implementing variations of *Lmd* and *Lmq* magnetizing inductances.

The next figure illustrates the good fit graphically (the diamonds are the actual points entered in the dialog box).



In this case, the following values are used:

<i>ifn</i>	1087 A
<i>ifd</i>	[695.64, 774.7, 917.5, 1001.6, 1082.2, 1175.9, 1293.6, 1430.2, 1583.7] A
<i>Vt</i>	[9660, 10623, 12243, 13063, 13757, 14437, 15180, 15890, 16567] V

Sample time (-1 for inherited) — Sample time for block

-1 (default) | scalar

Sample time used by the block. To inherit the sample time specified in the powergui block, set this parameter to -1.

Discrete solver model — Integration method

Trapezoidal non iterative (default) | Trapezoidal iterative (alg. loop) | Trapezoidal robust | Backward Euler robust

Integration method used by the block when the **Solver type** parameter of the powergui block is set to Discrete.

The **Discrete solver model** is automatically set to Trapezoidal robust when you select the **Automatically handle Discrete solver and Advanced tab solver settings of blocks** parameter of the powergui block.

The Trapezoidal non iterative and Trapezoidal iterative (alg. loop) methods are no longer recommended for discretizing the Synchronous Machine SI Fundamental block. Trapezoidal non iterative requires you to add non-negligible shunt loads at the machine terminals to guarantee simulation stability, and Trapezoidal iterative (alg. loop) may fail to converge and cause the simulation to stop when the number of machines increases in the model.

The Trapezoidal robust and Backward Euler robust methods allow you to eliminate the need to use parasitic loads and simulate a machine without loads. To eliminate topological errors of machines connected to an inductive circuit (for example, a circuit breaker connected in series with the machine); the machine models a negligible internal load of 0.01% of nominal power.

The Trapezoidal robust method is slightly more accurate than the Backward Euler robust method, especially when the model is simulated at larger sample times. The Trapezoidal robust method may produce slight damped numerical oscillations on machine voltage at no-load conditions, while the Backward Euler robust method prevents oscillations and maintains good accuracy.

For more information on what method to use in your application, see Simulating Discretized Electrical Systems.

Load Flow

The load flow parameters are used to define block parameters for use with the Load Flow tool of the powergui block. These load flow parameters are used for model initialization only. They have no impact on the block model or on the simulation performance.

Generator type — Machine generator type

PV (default) | swing | PQ

Generator type of the machine. Choice are:

- **swing** — Implements a generator controlling magnitude and phase angle of its terminal voltage. The reference voltage magnitude and angle are specified by the **Swing bus or PV bus voltage** and **Swing bus voltage angle** parameters of the Load Flow Bus block connected to the machine terminals.
- **PV** — Implements a generator controlling its output active power P and voltage magnitude V . P is specified by the **Active power generation P (W)** parameter of the block. V is specified by the **Swing bus or PV bus voltage** parameter of the Load Flow Bus block connected to the machine

terminals. You can control the minimum and maximum reactive power generated by the block by using the **Minimum reactive power Qmin (var)** and **Maximum reactive power Qmax (var)** parameters.

- PQ — Implements a generator controlling its output active power P and reactive power Q . P and Q are specified by the **Active power generation P (W)** and **Reactive power generation Q (var)** parameters of the block, respectively.

Active power generation P (W) — Active power generated by machine

0 (default) | scalar

Active power that you want generated by the machine, in watts. When the machine operates in motor mode, you specify a negative value.

Dependencies

To enable this parameter, set **Generator type** to PV or PQ.

Reactive power generation Q (var) — Reactive power generated by machine

0 (default) | scalar

Reactive power that you want generated by the machine, in vars. A negative value indicates that the reactive power is absorbed by the machine.

Dependencies

To enable this parameter, set **Generator type** to PQ.

Minimum reactive power Qmin (var) — Minimum reactive power generated by machine

-inf (default) | scalar

Minimum reactive power that can be generated by the machine while keeping the terminal voltage at its reference value. This reference voltage is specified by the **Swing bus or PV bus voltage** parameter of the Load Flow Bus block connected to the machine terminals. The default value is -inf, which means that there is no lower limit on the reactive power output.

Dependencies

To enable this parameter, set **Generator type** to PV.

Maximum reactive power Qmax (var) — Maximum reactive power generated by machine

inf (default) | scalar

Maximum reactive power that can be generated by the machine while keeping the terminal voltage at its reference value. This reference voltage is specified by the **Swing bus or PV bus voltage** parameter of the Load Flow Bus block connected to the machine terminals. The default value is inf, which means that there is no upper limit on the reactive power output.

Dependencies

To enable this parameter, set **Generator type** to PV.

Version History

Introduced before R2006a

References

- [1] Canay, I.M. "Causes of Discrepancies on Calculation of Rotor Quantities and Exact Equivalent Diagrams of the Synchronous Machine." *IEEE Transactions on Power Apparatus and Systems*. PAS-88, no. 7 (1969): 1114-1120.
- [2] Moeini, A., et al. "Synchronous Machine Stability model, an Update to IEEE Std 1110-2002 Data Translation Technique." *IEEE standards panel sessions*. 2018.
- [3] *IEEE Guide for Synchronous Generator Modeling Practices and Applications in Power System Stability Analyses*. IEEE Std 1110-2002 (Revision of IEEE Std 1110-1991 [2003]): 1-72.
- [4] Krause, P.C. *Analysis of Electric Machinery*. Section 12.5. New York: McGraw-Hill, 1986.
- [5] Kundur, P. *Power System Stability and Control*. New York, McGraw-Hill, 1994.

Extended Capabilities

C/C++ Code Generation

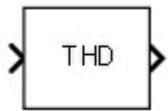
Generate C and C++ code using Simulink® Coder™.

See Also

Excitation System | Hydraulic Turbine and Governor | powergui | Simplified Synchronous Machine | Steam Turbine and Governor | Synchronous Machine pu Fundamental | Synchronous Machine pu Standard

THD

Compute total harmonic distortion (THD) of signal



Libraries:
Simscape / Electrical / Specialized Power Systems / Sensors and Measurements

Description

The THD block computes the total harmonic distortion (THD) of a periodic distorted signal. The signal can be a measured voltage or current.

The THD is defined as the root mean square (RMS) value of the total harmonics of the signal, divided by the RMS value of its fundamental signal. For example, for currents, the THD is defined as

$$\text{total harmonic distortion (THD)} = \frac{I_H}{I_F}$$
$$I_H = \sqrt{I_2^2 + I_3^2 + \dots + I_n^2} \quad I_n: \text{RMS value of the harmonic } n$$
$$I_F: \text{RMS value of the fundamental current}$$

The THD has a null value for a pure sinusoidal voltage or current.

Characteristics

Sample Time	Specified in the Sample Time parameter Continuous if Sample Time = 0
Scalar Expansion	Yes, of the parameters
Dimensionalized	Yes

Examples

The power_RMS_THD model uses a THD block to compute the THD of a 60 Hz signal with second harmonic and a DC bias.

The model sample time is parameterized by the Ts variable set to a default value of 50e-6 s. Set Ts to 0 in the command window to simulate the model in continuous mode.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Fundamental frequency of input signal (Hz) — Frequency of fundamental signal
60 (default) | scalar

Frequency, in hertz, of the fundamental signal.

Sample time — Block sample time

0 (default) | nonnegative scalar

Sample time of the block, in seconds. Set to 0 to implement a continuous block.

Version History

Introduced in R2013a

Extended Capabilities

C/C++ Code Generation

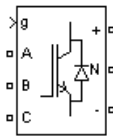
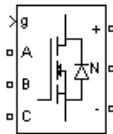
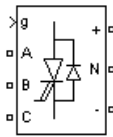
Generate C and C++ code using Simulink® Coder™.

See Also

RMS

Three-Level Bridge

Implement three-level neutral point clamped (NPC) power converter with selectable topologies and power switching devices

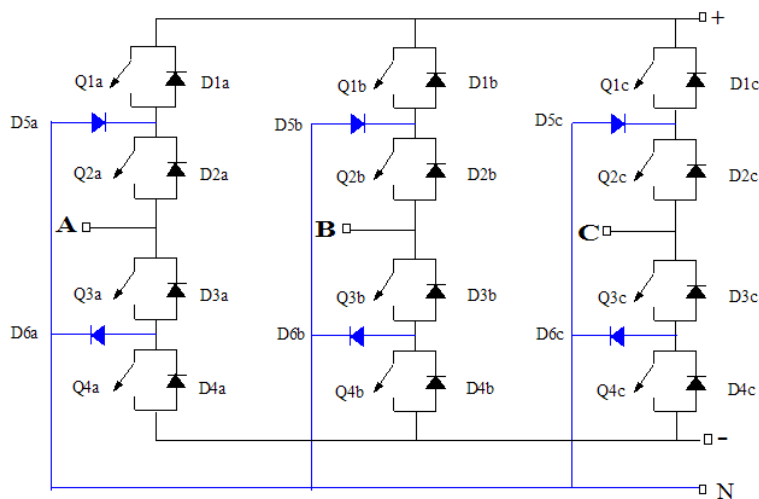


Libraries:

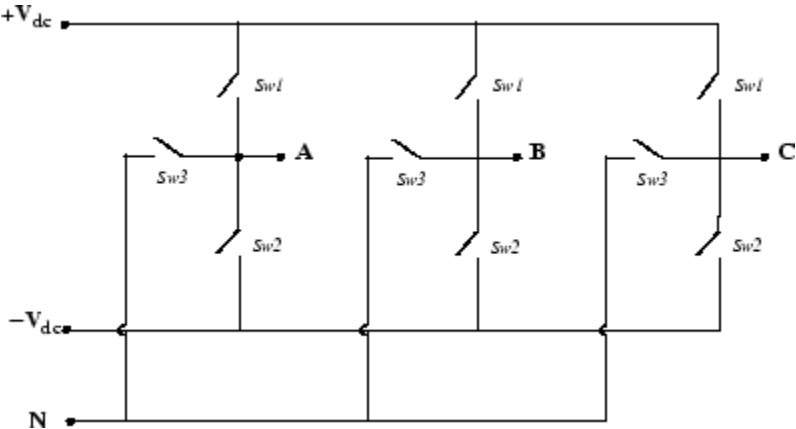
Simscape / Electrical / Specialized Power Systems / Power Electronics

Description

The Three-Level Bridge block implements a three-level power converter that consists of one, two, or three arms of power switching devices. Each arm consists of four switching devices along with their antiparallel diodes and two neutral clamping diodes as shown in the figure below.



The type of power switching device (IGBT, GTO, MOSFET, or ideal switch) and the number of arms (one, two, or three) are selectable from the dialog box. When the ideal switch is used as the switching device, the Three-Level Bridge block implements an ideal switch bridge having a three-level topology as shown in the following figure.



Assumptions and Limitations

This block does not model turn-on and turn-off times (Fall time, Tail time) of power switching devices.

Ports

Input

g — Gating signal to control converter
vector of 4, 8, or 12 elements

Vectorized gating signal that contains pulses to control the power electronic devices of the bridge. The length of the input vector depends on the number of arm you specified for the bridge topology.

Number of bridge arms	Pulse Vector of Input g
1	[Q1a,Q2a,Q3a,Q4a]
2	[Q1a,Q2a,Q3a,Q4a,Q1b,Q2b,Q3b,Q4b]
3	[Q1a,Q2a,Q3a,Q4a,Q1b,Q2b,Q3b,Q4b,Q1c,Q2c,Q3c,Q4c]

Note In the case of the ideal switch converter, the Q1 pulse is sent to Sw1, the Q4 pulse to Sw2, and a logical AND operation is performed on the Q2 and Q3 pulses and the result sent to Sw3.

Conserving

A — Phase A terminal
specialized electrical

Specialized electrical conserving port associated with the phase A terminal.

B — Phase B terminal
specialized electrical

Specialized electrical conserving port associated with the phase B terminal.

C — Phase C terminal
specialized electrical

Specialized electrical conserving port associated with the phase C terminal.

N — Neutral terminal
specialized electrical

Specialized electrical conserving port associated with the neutral terminal.

+ — Positive terminal
specialized electrical

Specialized electrical conserving port associated with the positive terminal.

- — Negative terminal
specialized electrical

Specialized electrical conserving port associated with the negative terminal.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Number of bridge arms — Number of bridge arms
3 (default) | 1 | 2

Number of bridge arms.

Snubber resistance Rs (Ohms) — Snubber resistance
1e5 (default) | positive scalar

Snubber resistance, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Snubber capacitance Cs (F) — Snubber capacitance
`inf` (default) | nonnegative scalar

Snubber capacitance, in farads. Set the **Snubber capacitance Cs (F)** parameter to `0` to eliminate the snubbers, or to `inf` to get a resistive snubber.

To eliminate snubbers in all power electronic devices when using a continuous solver, select the **Disable snubbers in switching devices** option in the Powergui **Preferences** tab.

When your system is discretized, you can simulate power electronic devices with virtually no snubbers by specifying purely resistive snubbers with a very large resistance, thus producing negligible leakage currents. The bridge operates satisfactorily with purely resistive snubbers.

Power electronic device — Power electronic device in bridge
GTO / Diodes (default) | MOSFET / Diodes | IGBT / Diodes | Ideal switches

Type of power electronic device to use in the bridge

Internal resistance Ron (Ohms) — Forward voltage across diode

1e-3 (default) | nonnegative scalar

Internal resistance of the selected devices and diodes, in ohms (Ω).

Forward voltages [Device Vf(V), Diode Vfd(V)] — Forward voltage across diode

[0 0] (default) | vector of two nonnegative values

Forward voltage of the selected devices (for GTO or IGBT only) and of the anti-parallel and clamping diodes, in volts

Dependencies

To enable this parameter, set **Power electronic device** to GTO/Diodes or IGBT / Diodes.

Measurements — Forward voltage across diode

None (default) | All device currents | Phase-to-Neutral and DC voltages | All voltages and currents

Select **All device currents** to measure the current flowing through all the components (switching devices and diodes). If the snubber devices are defined, the measured currents are those flowing through the power electronic devices only.

Select **Phase-to-Neutral** and **DC voltages** to measure the terminal voltages (AC and DC) of the Three-Level Bridge block.

Select **All voltages** and **currents** to measure all voltages and currents defined for the Three-Level Bridge block.

Place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurement** list box of the Multimeter block, the measurement is identified by a label followed by the block name.

Measurement	Label (for GTO, IGBT, MOSFET Devices)
Device currents	IQ1a, IQ2a, IQ3a, IQ4a, IQ1b, IQ2b, IQ3b, IQ4b, IQ1c, IQ2c, IQ3c, IQ4c, ID1a, ID2a, ID3a, ID4a, ID5a, ID6a, ID1b, ID2b, ID3b, ID4b, ID5b, ID6b, ID1c, ID2c, ID3c, ID4c, ID5c, ID6c
Terminal voltages	Uan:, Ubn:, Ucn:, Udc+:, Udc-:
Measurement	Label (for Ideal Switch Device)
Device currents	Isw1a, Isw2a, Isw3a, Isw1b, Isw2b, Isw3b, Isw1c, Isw2c, Isw3c
Terminal voltages	Uan:, Ubn:, Ucn:, Udc+:, Udc-:

Version History

Introduced before R2006a

Extended Capabilities

C/C++ Code Generation

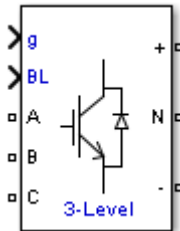
Generate C and C++ code using Simulink® Coder™.

See Also

Full-Bridge Converter | Two-Level Converter

Three-Level NPC Converter

Implement three-phase, three-level Neutral-Point Clamped (NPC) power converter



Libraries:

Simscape / Electrical / Specialized Power Systems / Power Electronics

Description

The Three-Level NPC Converter block implements a three-phase, three-level neutral-point clamped power converter. You can choose from four model types:

- **Switching devices** — The converter is modeled with IGBT/diode pairs controlled by firing pulses produced by a PWM generator. This model provides the most accurate simulation results.
- **Switching function** — The converter is modeled by a switching-function model. The switches are replaced with two voltage sources and two diodes on the AC side and with two current sources on the DC side.

The converter is controlled by firing pulses produced by a PWM generator (0/1 signals) or by firing pulses averaged over a specified period (PWM averaging: signals between 0 and 1). Both modes of operation produce harmonics normally generated by a PWM-controlled converter and also correctly simulate rectifying operation as well as blanking time. This model type is well-suited for real-time simulation.

- **Average model (Uref-controlled)** — The converter is modeled using a switching-function model directly controlled by the reference voltage signals (Uref). A PWM generator is not required. It does not model neutral point voltage balancing.
- **Average model (No rectifier mode)** — The block uses the voltage source directly controlled by the reference voltage to model the converter. The model does not require a PWM generator and does not simulate the rectifier mode. This setting provides the fastest simulations.

When doing real-time simulation, use the **Switching function** setting with the firing pulses averaged, or the **Average model (Uref-controlled)** or **Average model (No rectifier mode)** settings.

Ports

Input

g — Gating signal to control converter
vector of 12 elements

Vectorized gating signal to control the converter. The gating signal contains 12 firing pulses. The first four pulses control the Q1a to Q4a switching devices (phase A of the converter), pulses five to eight

control the Q1b to Q4b switching devices (phase B of the converter), and the last four pulses control the Q1c to Q4c switching devices (phase C of the converter).

Dependencies

To enable this port, set **Model type** to `Switching devices` or `Switching function`.

Uref — Signal to control converter

vector of three elements in the range [-1,1]

Vectorized signal to control the converter. The signal contains three reference modulating signals (one for each phase).

Dependencies

To enable this port, set **Model type** to `Average model (Uref-controlled)` or `Average model (No rectifier mode)`.

BL — Input to block firing pulses

scalar

Input port to block all firing pulses to the converter. To block all firing pulses to the converter, apply a signal value of 1 to this port.

Conserving

A — Phase A terminal

specialized electrical

Specialized electrical conserving port associated with the phase A terminal.

B — Phase B terminal

specialized electrical

Specialized electrical conserving port associated with the phase B terminal.

C — Phase C terminal

specialized electrical

Specialized electrical conserving port associated with the phase C terminal.

N — Neutral terminal

specialized electrical

Specialized electrical conserving port associated with the neutral terminal.

+ — Positive terminal

specialized electrical

Specialized electrical conserving port associated with the positive terminal.

- — Negative terminal

specialized electrical

Specialized electrical conserving port associated with the negative terminal.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Model type — Model type

Switching devices (default) | Switching function | Average model (Uref-controlled) | Average model (No rectifier mode)

Select one of these values:

- **Switching devices** — The converter is modeled with IGBT/diode pairs controlled by firing pulses produced by a PWM generator.
- **Switching function** — The converter is modeled by a switching-function model. The switches are replaced with two voltage sources and two diodes on the AC side and with two current sources on the DC side.
- **Average model (Uref-controlled)** — The converter is modeled using a switching-function model directly controlled by the reference voltage signals (Uref). A PWM generator is not required. It does not model neutral point voltage balancing.
- **Average model (No rectifier mode)** — The block uses the voltage source directly controlled by the reference voltage to model the converter. The model does not require a PWM generator and does not simulate the rectifier mode.

Device on-state resistance (Ohms) — Internal resistance of switching devices

1e-3 (default) | positive scalar

Internal resistance of the switching devices, in ohms.

Dependencies

To enable this parameter, set **Model type** to **Switching devices**.

Snubber resistance (Ohms) — Snubber resistance

1e6 (default) | positive scalar

Snubber resistance, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to **Switching devices**.

Snubber capacitance (F) — Snubber capacitance

inf (default) | nonnegative scalar

Snubber capacitance, in farads. Set the snubber capacitance to `0` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to **Switching devices**.

Diode on-state resistance (Ohms) — Internal resistance of diodes

1e-3 (default) | positive scalar

Internal resistance of the diodes, in ohms.

Dependencies

To enable this parameter, set **Model type** to `Switching function` or `Average model (Uref-controlled)`.

Diode snubber resistance (Ohms) — Snubber resistance of diodes

`1e6` (default) | positive scalar

Snubber resistance of the diodes, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to `Switching function` or `Average model (Uref-controlled)`.

Diode snubber capacitance (F) — Snubber capacitance of diodes

`inf` (default) | nonnegative scalar

Snubber capacitance of the diodes, in farads. Set the snubber capacitance to `0` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to `Switching function` or `Average model (Uref-controlled)`.

Diode forward voltage (V) — Forward voltage across diode

`1e-3` (default) | nonnegative scalar

Forward voltage, in volts, across the diode when it is conducting.

Dependencies

To enable this parameter, set **Model type** to `Switching function` or `Average model (Uref-controlled)`.

DC side snubber resistance (Ohms) — Snubber resistance across current sources

`inf` (default) | positive scalar

Snubber resistance across the two current sources, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to `Switching function`, `Average model (Uref-controlled)`, or `Average model (No rectifier mode)`.

AC side source resistance (Ohms) — Source resistance on AC side

`1e-3` (default) | positive scalar

Source resistance on the AC side, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to `Average model (No rectifier mode)`.

Version History

Introduced in R2015b

Extended Capabilities

C/C++ Code Generation

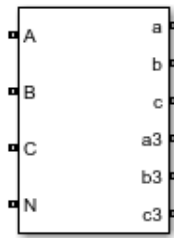
Generate C and C++ code using Simulink® Coder™.

See Also

Full-Bridge Converter | Two-Level Converter

Three-Phase Autotransformer with Tertiary Winding

Three-phase autotransformer with tertiary winding



Libraries:

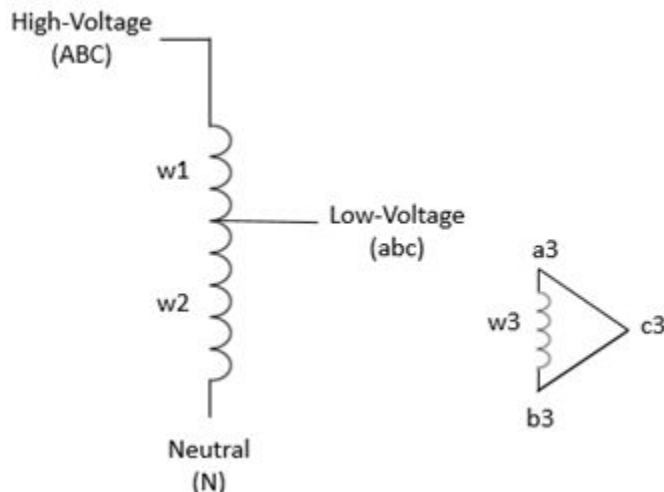
Simscape / Electrical / Specialized Power Systems / Power Grid Elements

Description

The Three-Phase Autotransformer with Tertiary Winding block represents a three-phase autotransformer. The high-voltage side is identified by the **A**, **B**, and **C** ports, the low-voltage side by the **a**, **b**, and **c** ports, and the tertiary-winding by the **a3**, **b3**, and **c3** ports.

Equivalent Circuit

The equivalent circuit of one phase is shown in the diagram. Each phase consists of three coupled windings: a series winding between the high-voltage terminals and low-voltage terminals, a common winding between the low-voltage terminals and the neutral terminal, and a tertiary winding connected in Delta D1.



Standard Model for Winding Resistances and Inductances

The resistances and leakage inductances of the three windings are determined from short-circuit test parameters using the following standard equations:

$$R1 = (R12*(1+k)/(1-k) + R13 - R23\_pu)/(1-k)/2 \quad R2 = (R12 - R13 + R23\_pu)/(1-k)/2 \quad R3 = (-R12 + R13 + R23*(1-2*k))/(1-k)/2$$

$$L1 = (L12*(1+k)/(1-k) + L13 - L23)/(1-k)/2 \quad L2 = (L12 - L13 + L23)/(1-k)/2 \quad L3 = (-L12 + L13 + L23*(1-2*k))/(1-k)/2$$

where:

- $R12$, $R13$, and $R23$ are the RHL, RHT, and RLT short-circuit test resistances.
- $L12$, $L13$, and $L23$ are the LHL, LHT, and LLT short-circuit test inductances.
- k is the voltage ratio between the high-voltage side and low-voltage side nominal voltages.

All parameters are in pu based on the nominal power and nominal voltage of the windings.

The standard equations listed above may produce negative winding resistances and inductances. Although negative values are permitted in phasor models (at 50 Hz or 60 Hz) using algebraic equations, these negative parameters may result in numerical instability in EMT models that use differential equations. In this case, a warning message suggests you modify the $R23$ or the $L23$ parameter and proposes a range of values that produce positive resistance and inductance values.

Alternate Model for Winding Resistances

To avoid the limitations of the standard model that may result in negative resistances or very uneven sharing of losses between winding 1 and winding 2, you may choose an alternate model for computing winding resistances. This alternate model assumes that the Joules losses corresponding to $R12$ are equally shared between winding w1 and winding w2 ($R2 = R1$), which is close to real life. $R1$ and $R2$ are in pu and are computed as follows:

$$R1 = R12/2/(1-k)^2 \quad R2 = R1$$

$R3$ is adjusted to obtain the specified $R13$ value, as given by the following equation:

$$R3 = R13 - R1*(1-k)^2 - R2*k^2.$$

Although this model returns an error on $R23$, this has a limited impact because it affects only the tertiary winding, which is frequently unloaded or feeds a maximum of 10% of autotransformer nominal power. This model more accurately represents the sharing of currents between winding 1 and winding 2 for DC or very low-frequency phenomena. For example, during a geomagnetic disturbance that produces very low-frequency earth electrical fields, the resulting geomagnetically induced currents (GICs) throughout the network are dependent on the DC autotransformer model.

Ports

Conserving

A - High-voltage phase A terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the phase A terminal on the high-voltage side of the transformer.

B - High-voltage phase B terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the phase B terminal on the high-voltage side of the transformer.

C - High-voltage phase C terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the phase C terminal on the high-voltage side of the transformer.

N - Neutral terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the neutral terminal of the transformer.

a - Low-voltage phase A terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the phase A terminal on the low-voltage side of the transformer.

b - Low-voltage phase B terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the phase B terminal on the low-voltage side of the transformer.

c - Low-voltage phase C terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the phase C terminal on the low-voltage side of the transformer.

a3 - Tertiary winding phase A terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the phase A terminal of the tertiary winding.

b3 - Tertiary winding phase B terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the phase B terminal of the tertiary winding.

c3 - Tertiary winding phase C terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the phase C terminal of the tertiary winding.

Parameters

Pnom (VA) - Transformer nominal power — Nominal power rating
300e6 (default) | positive number

Nominal power rating of the transformer, in volt-amperes (VA).

Fnom (Hz) - Transformer nominal frequency — Nominal frequency

60 (default) | positive number

Nominal frequency of the transformer, in hertz.

Vnom high-voltage side (VLL rms) - High-voltage side nominal voltage — Line-to-line nominal voltage

315e3 (default) | positive number

Line-to-line nominal voltage, in volts RMS, on the high-voltage side of the transformer.

Vnom low-voltage side (VLL rms) - Low-voltage side nominal voltage — Line-to-line nominal voltage

230e3 (default) | positive number

Line-to-line nominal voltage, in volts RMS, on the low-voltage side of the transformer.

Vnom tertiary-winding (VLL rms) - Tertiary winding nominal voltage — Line-to-line nominal voltage

12e3 (default) | positive number

Line-to-line nominal voltage, in volts RMS, of the tertiary winding.

RHL (pu/Pnom) - Short-circuit test resistance — Short-circuit resistance

0.0016 (default) | positive number

Short-circuit resistance, in pu, from the high-voltage terminals when the low-voltage terminals are short-circuited.

RHT (pu/Pnom) - Short-circuit test resistance — Short-circuit resistance

0.0088 (default) | positive number

Short-circuit resistance, in pu, from the high-voltage terminals when the tertiary terminals are short-circuited.

RLT (pu/Pnom) - Short-circuit test resistance — Short-circuit resistance

0.0088 (default) | positive number

Short-circuit resistance, in pu, from the low-voltage terminals when the tertiary terminals are short-circuited.

Calculation method for resistances - Winding resistances calculation method — Determine winding resistances

evaluated from RHL, RHT, RLT (default) | Evaluated from RHL, RHT ; same losses in R1 & R2

Select evaluated from RHL, RHT, RLT to use the standard model to determine the winding resistances.

Select Evaluated from RHL, RHT ; same losses in R1 & R2 to use the alternate model to determine the winding resistances.

LHL (pu/Pnom) - Short-circuit test inductance — Short-circuit inductance

0.11 (default) | positive number

Short-circuit inductance, in pu, from the high-voltage terminals when the low-voltage terminals are short-circuited.

LHT (pu/Pnom) - Short-circuit test inductance — Short-circuit inductance

0.22 (default) | positive number

Short-circuit inductance, in pu, from the high-voltage terminals when the tertiary terminals are short-circuited.

LLT (pu/Pnom) - Short-circuit test inductance — Short-circuit inductance

0.12 (default) | positive number

Short-circuit inductance, in pu, from the low-voltage terminals when the tertiary terminals are short-circuited.

Core type - Transformer core type — Core type

Three single-phase transformers (default) | Three-limb core (core-type)

Select `Three single-phase transformers` to implement a three-phase transformer using three single-phase transformer models. This core type is used in very large power transformers (hundreds of MW) found in utility grids.

Select `Three-limb core (core-type)` to implement a three-limb core three-phase transformer. In most applications, three-phase transformers use a three-limb core (core-type transformer). This option produces accurate simulation results during an asymmetrical fault for both linear and nonlinear models, including saturation. During asymmetrical voltage conditions, the zero-sequence flux of a core-type transformer returns outside the core, through an air gap, structural steel, and a tank. Thus, the natural zero-sequence inductance L_0 (without delta winding) of such a core-type transformer is usually very low (typically $0.3 \text{ pu} < L_0 < 2 \text{ pu}$) compared with a three-phase transformer using three single-phase units ($L_0 > 100 \text{ pu}$). This low L_0 value affects voltages, currents, and flux unbalances during linear and saturated operation.

Inductance L_0 of zero-sequence flux path return (pu) - Three-limb core transformer inductance — Inductance L_0

0.5 (default) | scalar number

Inductance L_0 , in pu, of the three-limb core transformer.

Dependencies

To enable this parameter, set **Core Type** to `Three single-phase transformers`.

Magnetization resistance R_m (pu/Pnom) - Magnetization resistance — Magnetization resistance

500 (default) | scalar number

Magnetization resistance R_m , in pu.

Magnetization inductance L_m (pu/Pnom) - Magnetization inductance — Magnetization inductance

500 (default) | scalar number

Magnetization inductance L_m , in pu, for a nonsaturable core.

Dependencies

To enable this parameter, clear **Simulate saturation**.

Simulate saturation - Activate transformer saturation — Implement saturable core
off (default) | on

Select to implement a saturable core.

Saturation characteristic [i1 phi1; i2 phi2; ...] - Saturation characteristic — Saturation characteristic
[0,0 ; 0.0024,1.2 ; 1.0,1.52] (default) | matrix

Saturation characteristic for the saturable core. Specify a series of current and flux pairs, in pu, starting with the pair (0, 0).

Dependencies

To enable this parameter, select **Simulate saturation**.

Specify initial fluxes - Activate initial flux specification — Define initial fluxes
off (default) | on

Select to define the initial fluxes with the **Initial fluxes [phi0A, phi0B, phi0C]** parameter.

When the **Specify initial fluxes** parameter is not selected upon simulation, the Simscape Electrical Specialized Power Systems software automatically computes the initial fluxes to start the simulation in steady state. The computed values are saved in the **Initial Fluxes** parameter and overwrite any previous values.

Dependencies

To enable this parameter, select **Simulate saturation**.

Initial fluxes [phi0A , phi0B , phi0C] - Initial fluxes — Specifies initial fluxes
[0, 0, 0] (default) | vector of 3 values

Specifies initial fluxes for each phase of the transformer.

When the **Specify initial fluxes** parameter is not selected upon simulation, the Simscape Electrical Specialized Power Systems software automatically computes the initial fluxes to start the simulation in steady state. The computed values are saved in the **Initial Fluxes** parameter and overwrite any previous values.

Dependencies

To enable this parameter, select **Simulate saturation** and **Specify initial fluxes**.

Measurements - Measurements — Measurements

None (default) | Winding voltages | Fluxes and excitation currents ($I_{mag} + I_{Rm}$) |
Fluxes and magnetization currents (I_{mag}) | All measurements (V I Flux)

Select **Winding voltages** to measure the voltage across the winding terminals.

Select **Fluxes and excitation currents ($I_{mag} + I_{Rm}$)** to measure the flux linkage, in volt seconds (Vs), and the total excitation current including iron losses modeled by R_m .

Select **Fluxes** and magnetization currents (**Imag**) to measure the flux linkage, in volt seconds (Vs), and the magnetization current, in amperes (A), not including iron losses modeled by **Rm**.

Select **All measurements** (**V I Flux**) to measure the winding voltages, magnetization currents, and the flux linkages.

Place a **Multimeter** block in your model to display the selected measurements during the simulation. In the **Available Measurements** parameter of the **Multimeter** block, the measurements are identified by a label followed by the block name.

Advanced

To enable these settings, in the **powergui** block, set **Simulation type** to **Discrete**.

Break Algebraic loop in discrete saturation model - Activate Break algebraic loop in discrete saturation model — Insert delay

`off (default) | on`

Select to insert a delay at the output of the saturation model that computes the magnetization current as a function of flux linkage (the integral of input voltage computed by a trapezoidal method). This delay eliminates the algebraic loop that results from trapezoidal discretization methods and speeds up the simulation of the model. However, the delay introduces one time step time delay in the model and can cause numerical oscillations if the sample time is too large. The algebraic loop is required in most cases to get an accurate solution.

When this parameter is cleared, the discretization method of the saturation model is specified by the **Discrete solver model** parameter.

Discrete solver model - Discrete solver model — Resolve algebraic loop

`Trapezoidal robust (default) | Trapezoidal iterative | Backward Euler robust`

Select one of these methods to resolve the algebraic loop.

- **Trapezoidal iterative** — Although this method produces correct results, it is not recommended because Simulink tends to slow down and may fail to converge and stop simulation, especially if the model has a large number of transformers. Also, because of the Simulink algebraic loop constraint, this method cannot be used in real time. In R2018b and previous releases, this is setting is used when **Break Algebraic loop in discrete saturation model** parameter is cleared.
- **Trapezoidal robust** — This method is slightly more accurate than the **Backward Euler robust** method. However, it may produce slightly damped numerical oscillations on transformer voltages when the transformer is at no load.
- **Backward Euler robust** — This method provides good accuracy and prevents oscillations when the transformer is at no load.

The maximum number of iterations for the robust methods is specified in the **Preferences** tab of the **powergui** block, in the **Solver details for nonlinear elements** section. For real-time applications, you may need to limit the number of iterations. The two robust solvers are the recommended methods for discretizing the saturation model of the transformer.

For more information on which method to use in your application, see **Simulating Discretized Electrical Systems**.

Dependencies

To enable this parameter, clear **Break Algebraic loop in discrete saturation model**.

Version History

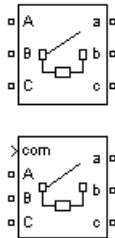
Introduced in R2021b

Extended Capabilities**C/C++ Code Generation**

Generate C and C++ code using Simulink® Coder™.

Three-Phase Breaker

Implement three-phase circuit breaker opening at current zero crossing



Libraries:

Simscape / Electrical / Specialized Power Systems / Power Grid Elements

Description

The Three-Phase Breaker block implements a three-phase circuit breaker where the opening and closing times can be controlled either from an external Simulink signal (external control mode), or from an internal control timer (internal control mode).

The Three-Phase Breaker block uses three Breaker blocks connected between the inputs and the outputs of the block. You can use this block in series with the three-phase element you want to switch. The arc extinction process of the Three-Phase Fault block is the same as for the Breaker block. See the Breaker block reference pages for details on the modeling of the single-phase breakers.

If the Three-Phase Breaker block is set in external control mode, a control input appears in the block icon. The control signal connected to the Simulink input must be either 0, which opens the breakers, or any positive value, which closes the breakers. For clarity, a 1 signal is commonly used to close the breakers. If the Three-Phase Breaker block is set in internal control mode, the switching times are specified in the dialog box of the block. The three individual breakers are controlled with the same signal.

Series Rs-Cs snubber circuit are included in the model. They can be optionally connected to the three individual breakers. If the Three-Phase Breaker block happens to be in series with an inductive circuit, an open circuit or a current source, you must use the snubbers.

Ports

Input

com — Control input
Simulink signal

Control input for the breaker.

Dependencies

To enable this port, select the **External** parameter.

Conserving

A — Phase A at input of breaker

specialized electrical

Specialized electrical conserving port associated with the phase A at the input of the breaker.

B — Phase B at input of breaker

specialized electrical

Specialized electrical conserving port associated with the phase B at the input of the breaker.

C — Phase C at input of breaker

specialized electrical

Specialized electrical conserving port associated with the phase C at the input of the breaker.

a — Phase A at output of breaker

specialized electrical

Specialized electrical conserving port associated with the phase A at the output of the breaker.

b — Phase B at output of breaker

specialized electrical

Specialized electrical conserving port associated with the phase B at the output of the breaker.

c — Phase C at output of breaker

specialized electrical

Specialized electrical conserving port associated with the phase C at the output of the breaker.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Initial status — Initial status of breakers

open (default) | closed

Initial status of the breakers. The initial status is the same for the three breakers. Depending on the initial status, the icon shows a closed contact or an open contact.

Switching of: Phase A — Switching of phase A

on (default) | off

If selected, the switching of phase A is activated. If not selected, the breaker of phase A stays in its initial status specified in the **Initial status** parameter.

Switching of: Phase B — Switching of phase B

on (default) | off

If selected, the switching of phase B is activated. If not selected, the breaker of phase B stays in its initial status specified in the **Initial status** parameter.

Switching of: Phase C — Switching of phase C

on (default) | off

If selected, the switching of phase C is activated. If not selected, the breaker of phase C stays in its initial status specified in the **Initial status** parameter.

Switching time (s) — Vector of switching times

[1/60 5/60] (default) | vector of scalars

Specify the vector of switching times when using the Three-Phase Breaker block in internal control mode. At each transition time the selected breakers opens or closes depending to their initial state.

Dependencies

To enable this parameter, clear the **External** parameter.

External — Option to enable external control signal

off (default) | on

If selected, adds a fourth input port to the Three-Phase Breaker block for an external control of the switching times of the breakers. The switching times are defined by a Simulink signal (0-1 sequence).

Breakers resistance Ron — Internal breaker resistance

0.01 (default) | positive scalar

Internal breaker resistance, in ohms (Ω). The **Breaker resistance Ron** parameter cannot be set to 0.

Snubbers resistance Rs — Snubber resistance

1e6 (default) | nonnegative scalar

Snubber resistance, in ohms (Ω). Set this parameter to `inf` to eliminate the snubbers from the model.

Snubbers capacitance Cs — Snubber capacitance

inf (default) | nonnegative scalar

Snubber capacitance, in farads (F). Set this parameter to 0 to eliminate the snubbers, or to `inf` to get resistive snubbers.

Measurements — Option for measurements

None (default) | Breaker voltages | Breaker currents | Breaker voltages and currents

Select `Breaker voltages` to measure the voltage across the three internal breaker terminals.

Select `Breaker currents` to measure the current flowing through the three internal breakers. If the snubber devices are connected, the measured currents are the ones flowing through the breakers contacts only.

Select `Breaker voltages and currents` to measure the breaker voltages and the breaker currents.

Place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurements** list box of the Multimeter block, the measurements is identified by a label followed by the block name and the phase:

Measurement	Label
Breaker voltages	Ub <block_name> /Breaker A:
Breaker currents	Ib <block_name> /Breaker A:

Version History

Introduced before R2006a

Extended Capabilities

C/C++ Code Generation

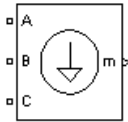
Generate C and C++ code using Simulink® Coder™.

See Also

Breaker | Three-Phase Fault

Three-Phase Dynamic Load

Implement three-phase dynamic load with active power and reactive power as function of voltage or controlled from external input



Libraries:

Simscape / Electrical / Specialized Power Systems / Passives

Description

The Three-Phase Dynamic Load block implements a three-phase, three-wire dynamic load whose active power P and reactive power Q vary as function of positive-sequence voltage. Negative- and zero-sequence currents are not simulated. The three load currents are therefore balanced, even under unbalanced load voltage conditions.

The load impedance is kept constant if the terminal voltage V of the load is lower than a specified value V_{min} . When the terminal voltage is greater than the V_{min} value, the active power P and reactive power Q of the load vary as follows:

$$P(s) = P_0 \left(\frac{V}{V_0} \right)^{n_p} \frac{1 + T_{p1}s}{1 + T_{p2}s}$$

$$Q(s) = Q_0 \left(\frac{V}{V_0} \right)^{n_q} \frac{1 + T_{q1}s}{1 + T_{q2}s},$$

where

- V_0 is the initial positive sequence voltage.
- P_0 and Q_0 are the initial active and reactive powers at the initial voltage V_0 .
- V is the positive-sequence voltage.
- n_p and n_q are exponents (usually between 1 and 3) controlling the nature of the load.
- T_{p1} and T_{p2} are time constants controlling the dynamics of the active power P .
- T_{q1} and T_{q2} are time constants controlling the dynamics of the reactive power Q .

For a constant current load, for example, you set n_p to 1 and n_q to 1, and for constant impedance load you set n_p to 2 and n_q to 2.

Ports

Input

PQ — Control for active and reactive power
vector of two elements

Control for the active and reactive powers of the load, specified as a vector of two signals $[P, Q]$.

Dependencies

To enable this port, select the **External control of PQ** parameter.

Output

m — Load output

vector of three elements

Vector that contains these three signals:

- Positive-sequence voltage (pu)
- Active power P (W)
- Reactive power Q (vars)

Conserving

A — Phase A terminal

specialized electrical

Specialized electrical conserving port associated with the terminal of phase A.

B — Phase B terminal

specialized electrical

Specialized electrical conserving port associated with the terminal of phase B.

C — Phase C terminal

specialized electrical

Specialized electrical conserving port associated with the terminal of phase C.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Nominal L-L voltage and frequency [Vn(Vrms) fn(Hz)] — Nominal phase-to-phase voltage and frequency of load

[500e3 60] (default) | vector of two elements

Nominal phase-to-phase voltage, in volts RMS, and nominal frequency, in hertz, of the load.

Active and reactive power at initial voltage [Po(W) Qo(var)] — Initial active and reactive power

[50e6 25e6] (default) | vector of two elements

Initial active power, Po, in watts, and initial reactive power, Qo, in vars, at the initial voltage, Vo.

When you use the Machine Initialization tool of the powergui block to initialize the dynamic load and start simulation in steady state, these parameters are automatically updated according to the P and Q points specified for the load.

When you use the Load Flow tool of the powergui block to initialize the dynamic load, these parameters represent the P and Q reference powers used by the load flow.

Initial positive-sequence voltage Vo [Mag(pu) Phase (deg.)] — Magnitude and phase of initial positive-sequence voltage of load
[0.994 -11.8] (default) | vector of two elements

Magnitude and phase of the initial positive-sequence voltage of the load.

When you use the Load Flow tool or the Machine Initialization tool of the powergui block to initialize the dynamic load and start simulation in steady state, this parameter is automatically updated using the values computed by the load flow.

External control of PQ — Option to define active and reactive power of load
off (default) | on

Option to define the active power and reactive power of the load by using an external Simulink vector of two signals.

[np nq] — Nature of the load
[1.3 2] (default) | vector of two elements

np and **nq** parameters that define the nature of the load.

Dependencies

To enable this parameter, clear the **External control of PQ** parameter.

Time constants [Tp1 Tp2 Tq1 Tq2] (s) — Time constants controlling dynamics of active and reactive power
[0 0 0 0] (default) | vector of four nonnegative elements

Time constants that control the dynamics of the active power and the reactive power.

Dependencies

To enable this parameter, clear the **External control of PQ** parameter.

Minimum voltage Vmin (pu) — Minimum voltage
0.7 (default) | nonnegative scalar

Minimum voltage at which the load dynamics commences. The load impedance is constant below this value.

Dependencies

To enable this parameter, clear the **External control of PQ** parameter.

Filtering time constant (s) — Time constant of current filter
1e-4 (default) | nonnegative scalar

Time constant, in seconds, of the current filter of the Three-Phase Dynamic Load block.

Version History

Introduced before R2006a

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

Three-Phase Parallel RLC Load | Three-Phase Series RLC Load

Three-Phase Fault

Implement programmable phase-to-phase and phase-to-ground fault breaker system



Libraries:

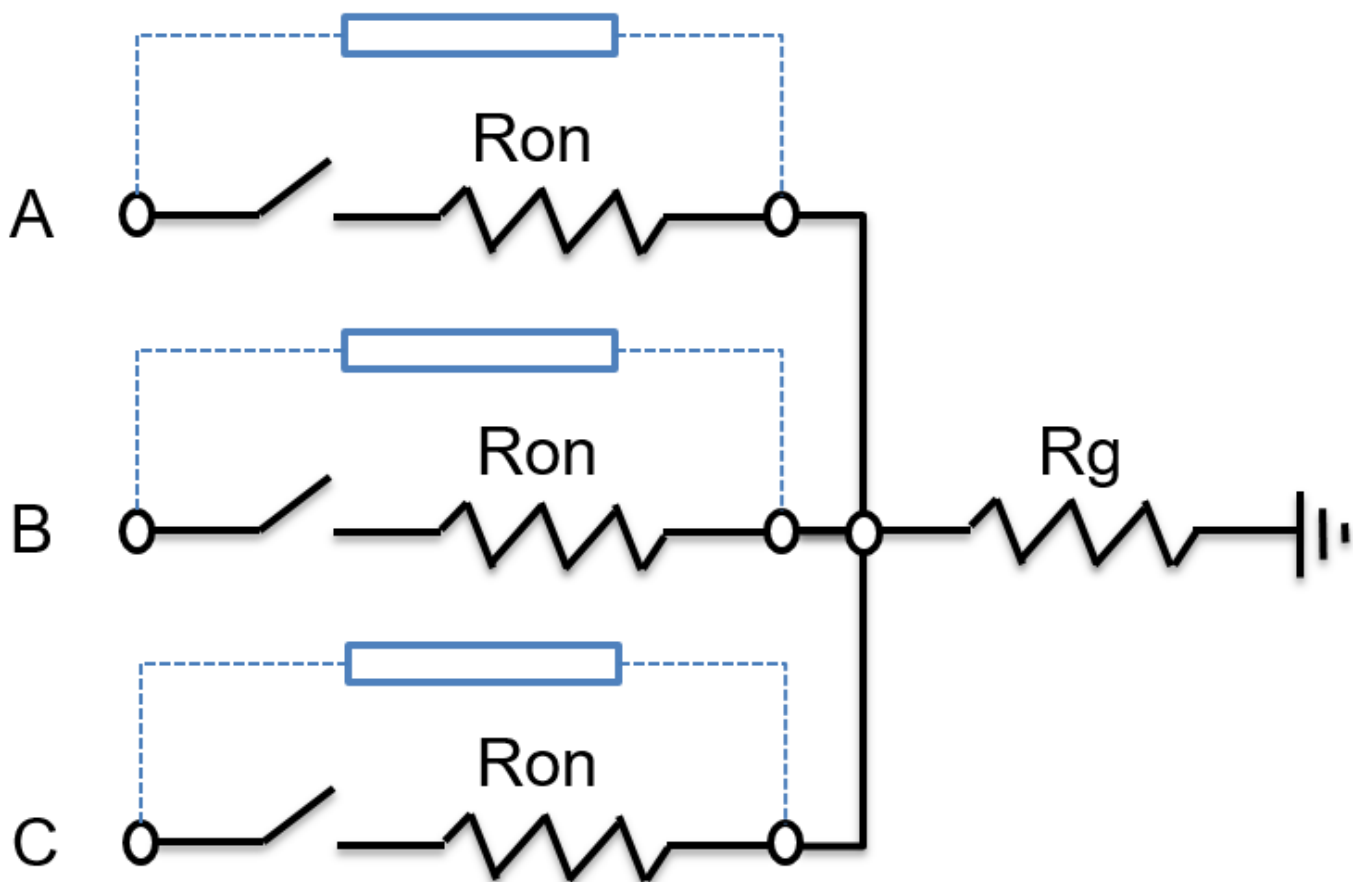
Simscape / Electrical / Specialized Power Systems / Power Grid Elements



Description

The Three-Phase Fault block implements a three-phase circuit breaker where the opening and closing times can be controlled either from an external Simulink signal (external control mode), or from an internal control timer (internal control mode).

The Three-Phase Fault block uses three Breaker blocks that can be individually switched on and off to program phase-to-phase faults, phase-to-ground faults, or a combination of phase-to-phase and ground faults. The arc extinction process of the Three-Phase Fault block is the same as for the Breaker block. See the Breaker block for details on the modeling of the single-phase breakers.



The ground resistance R_g is automatically set to 10^6 ohms when the ground fault option is not programmed. For example, to program a fault between the phases A and B you need to select the **Phase A** and **Phase B** block parameters only. To program a fault between the phase A and the ground, you need to select the **Phase A** and **Ground** parameters and specify a small value for the ground resistance.

If the Three-Phase Fault block is set in external control mode, a control input appears in the block icon. The control signal connected to the fourth Simulink input must be either 0, which opens the breakers, or any positive value, which closes the breakers. For clarity, a 1 signal is commonly used to close the breakers. If the Three-Phase Fault block is set in internal control mode, the switching times and status are specified in the dialog box of the block.

Series Rs-Cs snubber circuits are included in the model. They can be optionally connected to the fault breakers. If the Three-Phase Fault block is in series with an inductive circuit, an open circuit or a current source, you must use the snubbers.

Ports

Input

com — Control input
Simulink signal

Control input for the three-phase fault.

Dependencies

To enable this port, select the **External** parameter.

Conserving

A — Phase A terminal
specialized electrical

Specialized electrical conserving port associated with the phase A terminal.

B — Phase B terminal
specialized electrical

Specialized electrical conserving port associated with the phase B terminal.

C — Phase C terminal
specialized electrical

Specialized electrical conserving port associated with the phase C terminal.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Initial status — Initial status of fault breaker
0 (default) | scalar

Initial status of the fault breaker. You can also start the simulation in steady state with the fault initially applied on the system.

Phase A — Fault switching of phase A
on (default) | off

If selected, the fault switching of phase A is activated. If not selected, the breaker of phase A stays in the status specified in the **Initial status** parameter.

Phase B — Fault switching of phase B
on (default) | off

If selected, the fault switching of phase B is activated. If not selected, the breaker of phase B stays in the status specified in the **Initial status** parameter.

Phase C — Fault switching of phase C
on (default) | off

If selected, the fault switching of phase C is activated. If not selected, the breaker of phase C stays in the status specified in the **Initial status** parameter.

Ground — Fault switching of phase C
on (default) | off

If selected, the fault switching of ground is activated. A fault to the ground can be programmed for the activated phases. For example, if the **Phase C** and **Ground** parameters are selected, a fault to the

ground is applied to the phase C. The ground resistance is set internally to 1e6 ohms when the **Ground** parameter is not selected.

Switching time (s) — Vector of switching times

[1/60 5/60] (default) | vector of scalars

Specify the vector of switching times when using the block in internal control mode. At each transition time the selected fault breakers opens or closes depending to their initial state.

Dependencies

To enable this parameter, clear the **External** parameter.

External — Option to enable external control signal

off (default) | on

If selected, adds a fourth input port to the block for an external control of the switching times of the fault breakers. The switching times are defined by a Simulink signal (0-1 sequence).

Fault resistances Ron — Internal resistance of phase fault breakers

0.001 (default) | positive scalar

Internal resistance, in ohms (Ω), of the phase fault breakers. This parameter cannot be set to 0.

Dependencies

Ground resistance Rg — Ground resistance

0.01 (default) | positive scalar

Ground resistance, in ohms (Ω). This parameter cannot be set to 0.

Dependencies

To enable this parameter, select the **Ground** parameter.

Snubbers resistance Rs — Snubber resistance

1e6 (default) | nonnegative scalar

Snubber resistance, in ohms (Ω). Set this parameter to `inf` to eliminate the snubbers from the model.

Snubbers capacitance Cs — Snubber capacitance

`inf` (default) | nonnegative scalar

Snubber capacitance, in farads (F). Set this parameter to 0 to eliminate the snubbers, or to `inf` to get resistive snubbers.

Measurements — Option for measurements

None (default) | Fault voltages | Fault currents | Fault voltages and currents

Select **Fault voltages** to measure the voltage across the three internal fault breaker terminals.

Select **Fault currents** to measure the current flowing through the three internal breakers. If the snubber devices are connected, the measured currents are the ones flowing through the breakers contacts only.

Select **Fault voltages and currents** to measure the breaker voltages and the breaker currents.

Default is None.

Place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurements** list box of the Multimeter block, the measurements are identified by a label followed by the block name and the phase:

Measurement	Label
Fault voltages	Ub <block name> /Fault A:
Fault currents	Ib <block name> /Fault A:

Version History

Introduced before R2006a

Extended Capabilities

C/C++ Code Generation

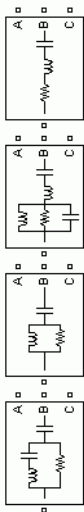
Generate C and C++ code using Simulink® Coder™.

See Also

Breaker | Multimeter | Three-Phase Breaker

Three-Phase Harmonic Filter

Implement four types of three-phase harmonic filters using RLC components



Libraries:

Simscape / Electrical / Specialized Power Systems / Passives

Description

The Three-Phase Harmonic Filter block models three-phase harmonic filters, which are shunt elements that are used in power systems for decreasing voltage distortion and for power factor correction. Nonlinear elements, such as power electronic converters, generate harmonic currents or harmonic voltages, which are injected into the power system. The resulting distorted currents flowing through the system impedance produce harmonic voltage distortion. Harmonic filters reduce distortion by diverting harmonic currents in low-impedance paths. Harmonic filters are capacitive at the fundamental frequency, so they are also used to produce the reactive power required by converters and for power factor correction.

To achieve an acceptable distortion, several banks of filters of different types are connected in parallel. The most commonly used filter types are:

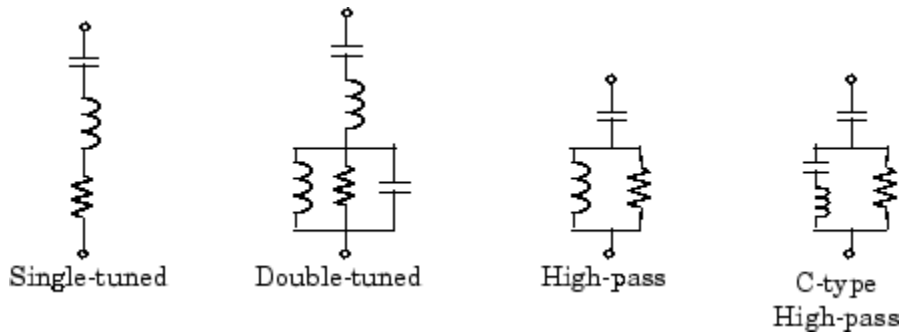
- Band-pass filters, which are used to filter the lowest order harmonics, such as 5th, 7th, 11th, and 13th order harmonics. Band-pass filters can be tuned at a single frequency (single-tuned filter) or at two frequencies (double-tuned filter).
- High-pass filters, which are used to filter high-order harmonics and cover a wide range of frequencies. A special type of high-pass filter, the C-type high-pass filter, is used to provide reactive power and avoid parallel resonances. It also allows filtering of the low-order harmonics (such as 3rd), while keeping zero losses at the fundamental frequency.

The Three-Phase Harmonic Filter block is built of RLC elements. The resistance, inductance, and capacitance values are determined from the filter type and the:

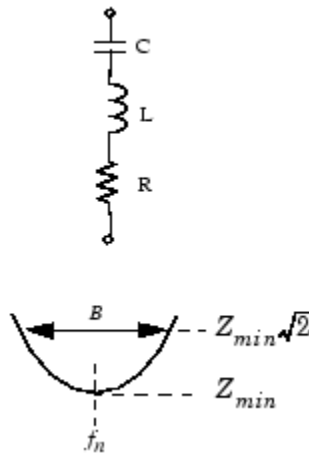
- Reactive power at nominal voltage
- Tuning frequencies

- Quality factor. The quality factor is a measure of the sharpness of the tuning frequency. It is determined by the resistance value.

The four types of filters that can be modeled with the Three-Phase Harmonic Filter block are shown below:



The simplest filter type is the single-tuned filter. The following figure gives the definition of the quality factor Q and formulae for computing the reactive power Q_C and losses (active power P). The quality factor Q of the filter is the quality factor of the reactance at the tuning frequency $Q = (nX_L)/R$. The quality factor determines the bandwidth B , which is a measure of the sharpness of the tuning frequency.



Tuned harmonic order	$n = f_n/f_1 = \sqrt{X_C/X_L}$
Quality factor	$Q = nX_L/R = X_C/(nR)$
Bandwidth	$B = f_n/Q$
Reactive power at f_1	$Q_C = (V^2/X_C) \cdot n^2/(n^2 - 1)$
Active power at f_1 (losses)	$P \approx (Q_C/Q) \cdot n/(n^2 - 1)$

where:

- f_1 = fundamental frequency
- $\omega = 2\pi f_1$ = angular frequency
- f_n = tuning frequency

- n = harmonic order = (f_n/f_1)
- V = nominal line-line voltage
- X_L = inductor reactance at fundamental frequency = $L\omega$
- X_C = capacitor reactance at fundamental frequency = $1/(C\omega)$

The double-tuned filter performs the same function as two single-tuned filters, although it has certain advantages: its losses are much lower and the impedance magnitude at the frequency of the parallel resonance that arises between the two tuning frequencies is lower.

The double-tuned filter consists of a series LC circuit and a parallel RLC circuit. If f_1 and f_2 are the two tuning frequencies, both the series circuit and the parallel circuit are tuned to approximately the mean geometric frequency, $f_m = \sqrt{f_1 f_2}$.

The quality factor Q of the double-tuned filter is defined as the quality factor of the parallel L and R elements at the mean frequency f_m : $Q = R / (L \cdot 2\pi f_m)$.

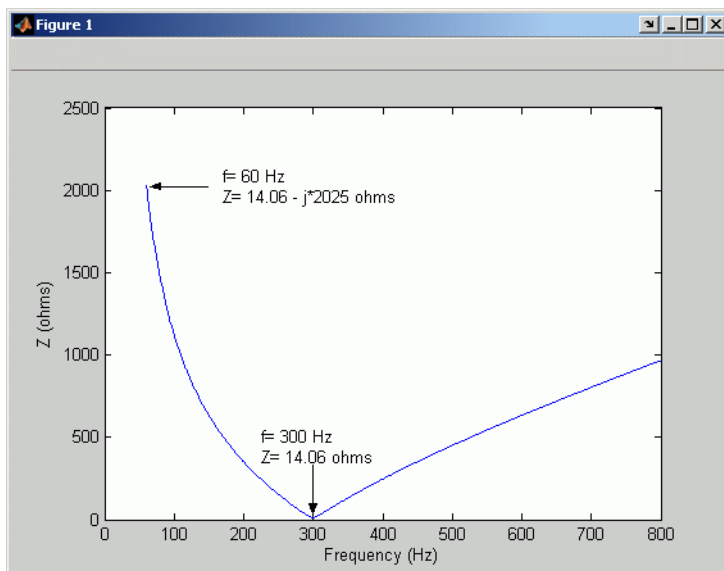
The high-pass filter is a single-tuned filter where the L and R elements are connected in parallel instead of series. This connection results in a wide-band filter that has an impedance at high frequencies limited by the resistance R.

The quality factor of the high-pass filter is the quality factor of the parallel RL circuit at the tuning frequency: $Q = R / (L \cdot 2\pi f_n)$.

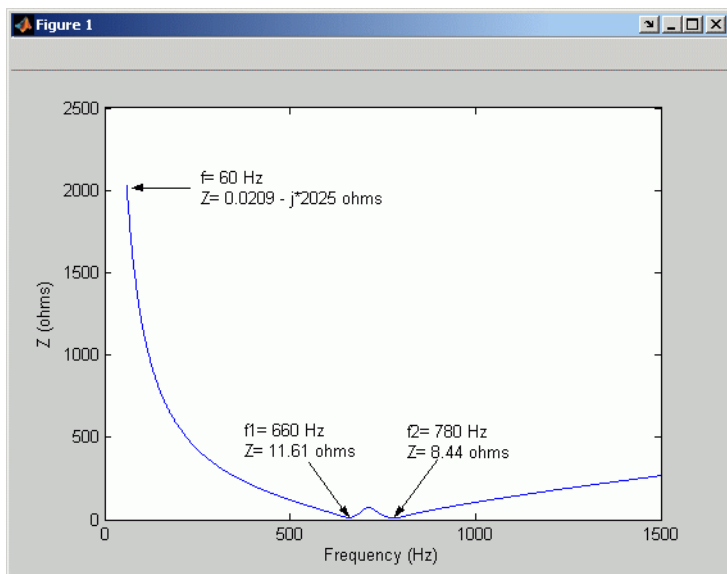
The C-type high-pass filter is a variation of the high-pass filter, where the inductance L is replaced with a series LC circuit tuned at the fundamental frequency. At fundamental frequency, the resistance is bypassed by the resonant LC circuit and losses are null.

The quality factor of the C-type filter is still given by the ratio: $Q = R / (L \cdot 2\pi f_n)$.

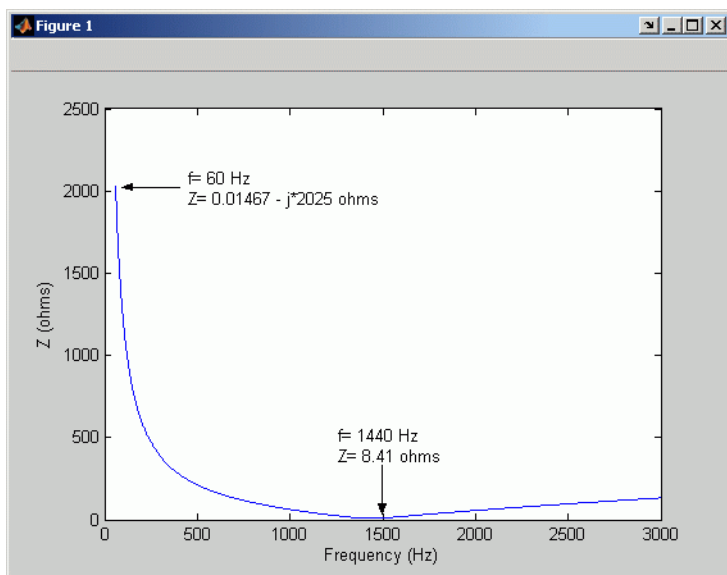
The following figures give values of R, L, and C, and the typical impedance-versus-frequency curves obtained for the four types of filters applied on a 60 Hz network. Each filter is rated 315 kV and 49 MVar.



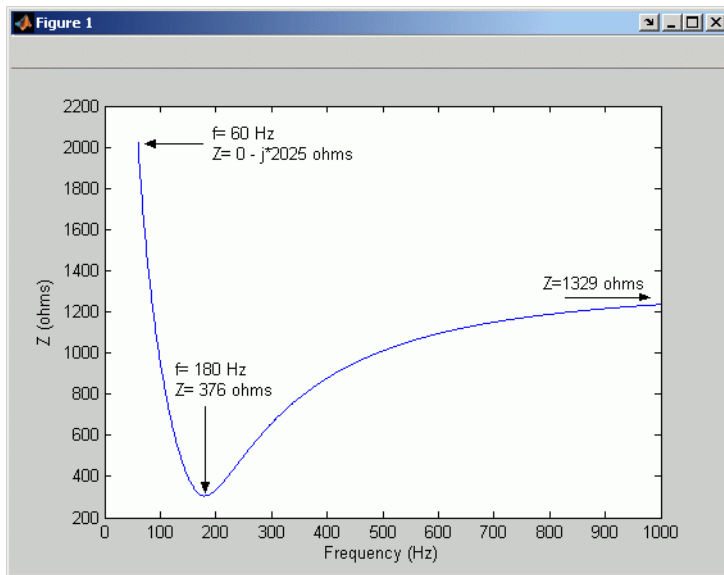
This figure illustrates a single-tuned filter at 315 kV and 49 MVar with a 5th order harmonic filter when Q is 30.



This figure illustrates a double-tuned filter at 315 kV and 49 MVar with 11th and 13th order harmonic filters when Q is 16.



This figure illustrates a high-pass filter at 315 kV and 49 MVar with a 24th order harmonic filter when Q is 10.



This figure illustrates a C-Type high-pass filter at 315 kV and 49 MVar with a 3rd order harmonic filter when Q is 1.75.

Ports

Conserving

A — Phase A electric terminal
specialized electrical

Specialized electrical conserving port associated with the phase A electrical terminal.

B — Phase B electric terminal
specialized electrical

Specialized electrical conserving port associated with the phase B electrical terminal.

C — Phase C electric terminal
specialized electrical

Specialized electrical conserving port associated with the phase C electrical terminal.

Port_4 — Neutral electric terminal
specialized electrical

Specialized electrical conserving port associated with the neutral electrical terminal.

Parameters

Type of filter — Filter type
Double-tuned (default) | Single-tuned | High-pass | C-type High-pass

Select one of the four filter types.

Filter connection — Connection of three filter branches

Y (neutral) (default) | Y (grounded) | Y (floating) | Delta

Select the connection of the three filter branches.

Y (grounded)	Neutral is grounded.
Y (floating)	Neutral is not accessible.
Y (neutral)	Neutral is made accessible through a fourth connector.
Delta	Three phases connected in delta.

Nominal L-L voltage and frequency [Vn(Vrms) fn(Hz)]: — Nominal phase-to-phase voltage and nominal frequency

[315e3 60] (default) | two-element vector of positive scalars

Nominal phase-to-phase voltage of the filter, in volts RMS (Vrms), and the nominal frequency, in hertz (Hz).

Nominal reactive power (var): — Three-phase capacitive reactive power

49e6 (default) | positive scalar

Three-phase capacitive reactive power Q_C , in vars.

Tuning frequencies [Fr1 (Hz) Fr2 (Hz)] — Turning frequency of filter

[11*60 13*60] (default) | two-element vector of positive scalars

Two tuning frequencies of the double-tuned filter, in hertz (Hz).

Dependencies

To enable this parameter, set **Type of filter** to Double - tuned.

Tuning frequency (Hz) — Turning frequency of filter

[5*60] (default) | positive scalar

Tuning frequency of the single frequency filter (single-tuned, high-pass or C-type high-pass), in hertz (Hz).

Dependencies

To enable this parameter, set **Type of filter** to Single-tuned, High-pass, or C-type High-pass.

Quality factor (Q) — Quality factor of filter

16 (default) | positive scalar

Quality factor Q of the filter, as defined in the “Description” on page 1-597 section.

Measurements — Internal variables to measure

None (default) | Branch voltages | Branch currents | Branch voltages and currents

Specify which internal variables to measure in the model using the Multimeter block.

- None
- Branch voltages -- Select to measure the three voltages across each phase of the Three-Phase Harmonic Filter block terminals. For a Y connection, these voltages are the phase-to-ground or phase-to-neutral voltages. For a delta connection, these voltages are the phase-to-phase voltages.

- **Branch currents** -- Select to measure the three currents flowing through each phase of the filter. For a delta connection, these currents are the currents flowing in each branch of the delta.
- **Branch voltages and currents** -- Select to measure the three voltages and the three currents of the Three-Phase Harmonic Filter block.

Place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurements** list of the Multimeter block, the measurements are identified by a label followed by the block name.

Measurement		Label
Branch voltages	Y(grounded)	Uag :
	Y(floating)	Uan :
	Y(neutral)	Uan :
	Delta	Uab :
Branch currents	Y(grounded)	Iag :
	Y(floating)	Ian :
	Y(neutral)	Ian :
	Delta	Iab :

Version History

Introduced before R2006a

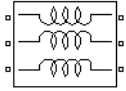
Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

Three-Phase Mutual Inductance Z1-Z0

Implement three-phase impedance with mutual coupling among phases



Library

Simscape / Electrical / Specialized Power Systems / Power Grid Elements

Description

The Three-Phase Mutual Inductance Z1-Z0 block implements a three-phase balanced inductive and resistive impedance with mutual coupling between phases. This block performs the same function as the three-winding Mutual Inductance block. For three-phase balanced power systems, it provides a more convenient way of entering system parameters in terms of positive- and zero-sequence resistances and inductances than the self- and mutual resistances and inductances.

Parameters

Positive-sequence parameters

The positive-sequence resistance R1, in ohms (Ω), and the positive-sequence inductance L1, in henries (H). Default is [2 50e-3].

Zero-sequence parameters

The zero-sequence resistance R0, in ohms (Ω), and the zero-sequence inductance L0, in henries (H). Default is [4 100e-3].

Examples

The power_3phmutseq10 example illustrates the use of the Three-Phase Mutual Inductance Z1-Z0 block to build a three-phase inductive source with different values for the positive-sequence impedance Z1 and the zero-sequence impedance Z0. The programmed impedance values are $Z_1 = 1 + j1 \Omega$ and $Z_0 = 2 + j2 \Omega$. The Three-Phase Programmable Voltage Source block is used to generate a 1-volt, 0-degree, positive-sequence internal voltage. At $t = 0.1$ s, a 1-volt, 0-degree, zero-sequence voltage is added to the positive-sequence voltage. The three source terminals are short-circuited to ground and the resulting positive-, negative-, and zero-sequence currents are measured using the Discrete 3-Phase Sequence Analyzer block.

The polar impedance values are $Z_1 = \sqrt{2} \angle 45^\circ \Omega$ and $Z_0 = 2\sqrt{2} \angle 45^\circ \Omega$.

Therefore, the positive- and zero-sequence currents displayed on the scope are

$$I_1 = \frac{V_1}{Z_1} = \frac{1}{\sqrt{2} \angle 45^\circ} = 0.7071 A \angle -45^\circ$$

$$I_0 = \frac{V_0}{Z_0} = \frac{1}{2\sqrt{2} \angle 45^\circ} = 0.3536 A \angle -45^\circ$$

The transients observed on the magnitude and the phase angle of the zero-sequence current when the zero-sequence voltage is added (at $t = 0.1$ s) are due to the Fourier measurement technique used by the Discrete 3-Phase Sequence Analyzer block. As the Fourier analysis uses a running average window of one cycle, it takes one cycle for the magnitude and phase to stabilize.

See Also

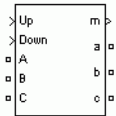
Mutual Inductance

Version History

Introduced before R2006a

Three-Phase OLTC Phase Shifting Transformer Delta-Hexagonal (Phasor Type)

Implement phasor model of three-phase OLTC phase-shifting transformer using delta hexagonal connection



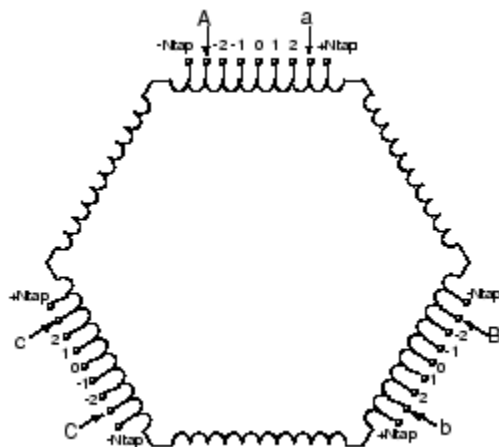
Library

Simscape / Electrical / Specialized Power Systems / Power Grid Elements

Description

This block is used to model a phase-shifting transformer using on-load tap changers (OLTC) for introducing a phase shift between three-phase voltages at two buses in a transmission system. Controlling phase-shift on a transmission system affects primarily flow of active power. Although the phase-shifting transformer does not provide as much flexibility and speed as power-electronics based FACTS, it can be considered as a basic power flow controller. This is why it has been included in the facts library. The dynamic performance of the phase-shifting transformer can be enhanced by using a thyristor-based tap changer instead of a mechanical tap changer. As this model is a phasor model which does not implement the details of current transfer from one tap to the next tap, you can use it to model a thyristor-based phase-shifter. You can also use this block together with the Three-phase OLTC Regulating Transformer (Phasor Type) block for building phasor models of more complex transformer-based FACTS topologies.

The delta hexagonal connection consists of three pairs of windings interconnected in a hexagonal configuration as shown on the figure below.



Delta-Hexagonal Phase-Shifting Transformer Connections

$$\psi = 2 \times \arctan(-k/\sqrt{3})$$

Symbol	Range
ψ = phase shift of abc output voltages with respect to ABC input voltages	$-60^\circ \leq \psi \leq +60^\circ$
N = tap position	$-N_{\text{tap}} \leq N \leq +N_{\text{tap}}$
k = relative tap position = N/N_{tap}	$-1 \leq k \leq +1$

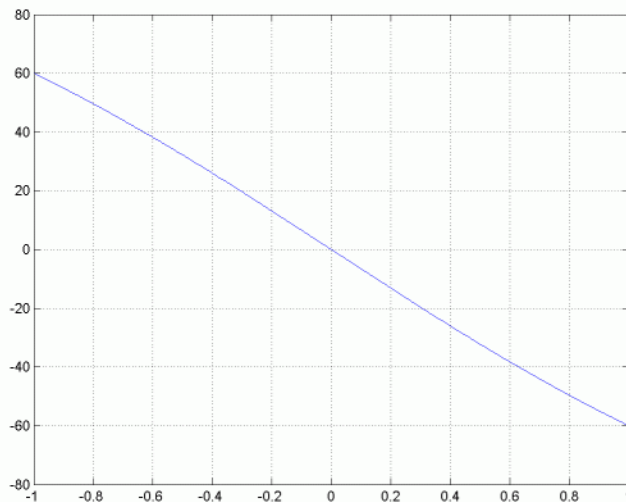
Each phase consists of two coupled windings drawn in parallel on the figure: one tapped winding with two OLTCs and one winding without taps. All windings have the same number of turns. The two OLTCs vary phase shift by moving transformer input terminals (A, B, C) and outputs terminals (a, b, c) symmetrically with respect to center tap 0. This delta hexagonal connection has the advantage of keeping a 1:1 voltage ratio while the phase shift is varied.

When the two OLTCs move taps from the center position (0) to the winding end (position N_{tap}), the phase shift between inputs (ABC) and outputs (abc) varies from 0 degrees to 60 degrees. When ABC are at position $-N_{\text{tap}}$ and abc are at position $+N_{\text{tap}}$, the output voltages abc are lagging input voltages ABC by 60 degrees. On the other end, when ABC are at position $+N_{\text{tap}}$ and abc are at position $-N_{\text{tap}}$, the output voltages abc are leading input voltages ABC by 60 degrees. For intermediate positions, the phase shift ψ is given by the equation on the figure. This equation assumes that all tap are evenly spaced.

For example, if each half tapped winding consists of 10 taps (total of 21 taps/ winding including center tap 0) and if ABC and abc terminals are respectively at tap -7 and tap $+7$, then, $k=7/10=0.7$.

Therefore, abc voltages are lagging ABC voltages by 44 degrees.

The phase angle varies almost linearly as function of tap position as shown on the figure below.



Variation of Phase Shift as Function of Tap Position

You control tap positions, and therefore phase shift, by sending pulses to one of the two block inputs labeled Up and Down. Applying a pulse to the Up (or Down) input moves the tap position upward (or downward) when the signal is changing from 0 to 1.

Mechanical tap changers are relatively slow devices. The time required to move from one tap position to the next one is comprised between 3 and 10 seconds. You specify this mechanical time delay in the block menu.

Note OLTCs use additional switches and resistors (or inductors) to transfer current from the outgoing tap to the ongoing tap without interrupting load current. During the transfer, taps are temporarily short circuited through resistors or inductors. The transfer time (typically from 40 ms to 60 ms) is fast as compared to the tap selection process (3 s to 10 s). As this block implements a phasor model for study of transient stability of power systems in the range of seconds to minutes, the tap transfer process is not modelled and an instantaneous tap transfer is assumed. A detailed delta-hexagonal phase-shifting transformer model is provided in the “Delta-Hexagonal Phase Shifting Transformer” example.

Parameters

Nominal parameters [Vnom Pnom Fnom]

The nominal line to line voltage, in volts rms, the three-phase nominal power, in VA and the nominal frequency, in hertz. Default is [120e3 300e6 60].

Resistance and leakage reactance at max. tap [R X]

The transformer short circuit impedance due to winding resistances and leakage reactances, at maximum phase shift (input and output taps at $-N_{tap}$ and $+N_{tap}$ positions). Specify resistance and reactance, in pu. Default is [0.15/30 0.15].

The transformer positive-sequence impedance Z_1 , in pu, and zero-sequence Z_0 , in pu, vary with relative tap position k as follows:

$$Z_1 = (R + jX) \frac{|k|(-3k^2 + 2|k| + 9)}{2(k^2 + 3)}$$

$$Z_0 = k^2 \frac{R_m jX_m}{R_m + jX_m},$$

where: R , X are the transformer resistance and reactance defined in the parameter **Resistance and leakage reactance at max. tap** ($k=1$ or $k=-1$) and R_m , X_m are the transformer **magnetizing** resistance and reactance defined in the **Magnetizing branch** parameter described below.

Note that these impedances keep the same value for positive and negative values of k . For $k=0$, impedances are zero because the transformer input terminals are short-circuited with output terminals.

Magnetizing branch [Rm Xm]

The resistance, in pu, and the reactance, in pu of the parallel R_m , X_m branch modelling respectively iron losses and reactive magnetizing currents. Saturation is not modelled. Default is [500 500].

Number of taps per half winding (Ntap)

The number of taps per half winding. The total number of taps per winding, including center tap 0, is therefore $2 \cdot N_{tap} + 1$. Default is 10.

Initial tap

Specify an integer (Tapinit) corresponding the initial tap position of output terminals abc. The initial tap position of input terminals ABC is therefore -Tapinit. Default is 0.

Tap selection time

The mechanical time delay, in seconds, required for the OLTCs to move the taps by one position. Typical values are in the 3s-10s range. Default is 2.9.

Initial pos. seq. output currents [Mag Phase]

The initial value of the positive-sequence current phasor (Magnitude in pu and Phase in degrees) flowing out of terminals abc. If you know the initial value of currents you may specify it in order to start simulation in steady state. If you don't know these values, you can leave [0 0] (default). The system will reach steady-state after a short transient.

Inputs and Outputs

A B C

The three transformer input terminals

a b c

The three transformer output terminals

Up

Simulink input for controlling tap position. Applying a pulse to this input will initiate upward tap changing when the pulse is changing from 0 to 1.

Down

Simulink input for controlling tap position. Applying a pulse to this input will initiate downward tap changing when the pulse is changing from 0 to 1.

m

Simulink output vector containing 17 internal signals. These signals are either complex signals (voltage phasors, current phasors or impedances) or control signals. They can be individually accessed by using the Bus Selector block. They are, in order:

Signal	Signal Group	Signal Names	Definition
1-3	VABC (cmplx)	VA (pu) VB (pu) VC (pu)	Phasor voltages (phase to ground) at the transformer input terminals A, B, C (pu)
4-6	Vabc (cmplx)	Va (pu) Vb (pu) Vc (pu)	Phasor voltages (phase to ground) at the transformer output terminals a, b, c (pu)
7-9	IABC (cmplx)	IA (pu) IB (pu) IC (pu)	Phasor currents flowing into the input terminals A, B, C
10-12	Iabc (cmplx)	Ia (pu) Ib (pu) Ic (pu)	Phasor currents flowing out of the output terminals a, b, c
13-14	Z (cmplx)	Z1 (pu) Z0 (pu)	Positive- and zero-sequence complex impedances (R+jX)

Signal	Signal Group	Signal Names	Definition
15	Psi (degrees)	Psi (degrees)	Phase shift of abc output voltages with respect to ABC input voltages
16	Tap	Tap	Tap position
17	Ready	Ready	Logical signal generated by the tap changer controller. The Ready signal becomes (1) after the tap selection has been completed, thus enabling a new tap change. The Up and Down pulses to the OLTC are blocked as long as the Ready signal is (0).

Examples

See the “Delta-Hexagonal Phase Shifting Transformer” example, which illustrates the use of the Three-Phase OLTC Phase Shifting Transformer Delta-Hexagonal (Phasor Type) block to control power transfer between two equivalent sources in a 120-kV transmission. The phasor model is compared with a detailed model of phase shifting transformer. Look under the masked blocks to see how both models are implemented. The detailed model uses switches and three Multi-Winding Transformer blocks, whereas the phasor model uses current sources.

See Also

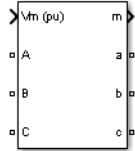
Three-Phase OLTC Regulating Transformer (Phasor Type)

Version History

Introduced in R2006a

Three-Phase OLTC Regulating Transformer (Phasor Type)

Implement phasor model of three-phase OLTC regulating transformer



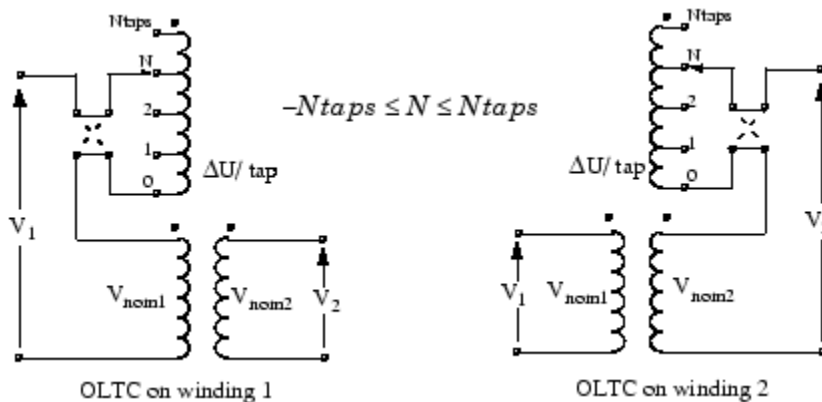
Library

Simscape / Electrical / Specialized Power Systems / Power Grid Elements

Description

This block is used to model a three-phase two-winding transformer or autotransformer using an on-load tap changer (OLTC) for regulating voltage on a transmission or distribution system. Controlling voltage on a transmission system affects primarily flow of reactive power, which, in turn, affects the power transfer limits. Although the regulating transformer does not provide as much flexibility and speed as power-electronics based FACTS, it can be considered as a basic power flow controller. This is why it has been included in the facts library. The dynamic performance of the regulating transformer can be enhanced by using a thyristor-based tap changer instead of a mechanical tap changer. As this model is a phasor model which does not implement the details of current commutation from one tap to the next tap, you can use it to model a thyristor-based tap changer and implement your own control system by selecting **External control of OLTC** in the block menu. You can also use this block together with the Three-phase OLTC Phase-Shifting Transformer Delta-Hexagonal (Phasor Type) block for building phasor models of more complex transformer-based FACTS topologies.

The figure below shows one phase of the three-phase regulating transformer. Each phase consists of main windings 1 and 2 and of a tapped regulation winding.



One Phase of the Regulating Transformer

$-N_{\text{taps}} \leq N \leq +N_{\text{taps}}$. The voltage ratios on the two windings are:

Winding 1	Winding 2
$\frac{V_2}{V_1} = \frac{1}{(1 + N \cdot \Delta U)} \times \frac{V_{\text{nom2}}}{V_{\text{nom1}}}$	$\frac{V_2}{V_1} = (1 + N \cdot \Delta U) \times \frac{V_{\text{nom2}}}{V_{\text{nom1}}}$

Voltage regulation is performed by varying the transformer ratio V_2/V_1 by means of the OLTC. The OLTC can be connected either on winding 1 (left diagram) or on winding 2 (right diagram). The OLTC can select any tap from position 0 (no voltage correction) to maximum tap (position N_{tap}) yielding maximum voltage correction. The OLTC is also equipped with a reversing switch which allows connecting the regulation winding either in additive or subtractive polarity. The factor multiplying $V_{\text{nom2}}/V_{\text{nom1}}$ is the voltage correction factor. It is given by:

$1/(1 + N \Delta U)$ for OLTC on winding 1
 $1 + N \Delta U$ for OLTC on winding 2

where

N = tap position

ΔU = voltage per tap in pu of nominal voltage of winding 1 or winding 2.

Negative values of N correspond to the reversing switch position shown in dashed lines (regulation winding in subtractive polarity). The transformer resistance and leakage reactance also vary with the tap position.

Mechanical tap changers are relatively slow devices. The time required to move from one tap position to the next one is comprised between 3 and 10 seconds. You specify this mechanical time delay in the block menu.

Note OLTCs use additional switches and resistors (or inductors) to transfer current from the outgoing tap to the ongoing tap without interrupting load current. During the transfer, taps are temporarily short circuited through resistors or inductors. The transfer time (typically from 40 ms to 60 ms) is fast as compared to the tap selection process (3 s to 10 s). As this block implements a phasor model for study of transient stability of power systems in the range of seconds to minutes, the tap transfer process is not modelled and an instantaneous tap transfer is assumed. A detailed regulating transformer model is provided in the “OLTC Regulating Transformer Phasor Model” example.

Automatic Voltage Control

The regulating transformer is associated with a control system which regulates voltage at the transformer terminals (side 1 or side 2) or at a remote bus. Such a control system is provided in the Three-Phase OLTC Regulating Transformer (Phasor Type) block. You then connect at the V_m input of the block a Simulink signal, which is usually the magnitude of the positive-sequence voltage (in pu) to be controlled, but it can be any signal. The control system adjusts automatically the tap position until the measured voltage V_m is equal to the reference voltage V_{ref} specified in the block menu.

The voltage regulator is a hysteresis type regulator. Each time a tap change is required, the regulator sends a pulse either to the Up or Down input of the Tap Changer Controller. Look under the block mask to see how these control blocks are built. The regulator asks for a tap change if

$$|V_m - V_{\text{ref}}| > \text{DeadBand}/2$$

during a time $t > \text{Delay}$, where V_{ref} , DeadBand, and Delay are parameters of the voltage regulator.

External Control of OLTC

If you want to implement your own control system or manually control the tap position, you select **External control of OLTC**. The V_m input is then replaced by two inputs labeled Up and Down. Applying a pulse to the Up or Down input moves the tap position upward or downward when the signal is changing from 0 to 1.

Parameters

The transformer and OLTC parameters are grouped in two categories: Transformer parameters and OLTC and Voltage Regulator parameters. Use the **Show** listbox to select which group of parameters you want to visualize.

Transformer Tab

Winding 1 connection (ABC terminals)

The connection of the three windings 1. Select Y (ungrounded wye), Yg (wye grounded), Delta(D1) (Delta connection introducing a -30 degrees phase-shift) or Delta(D11) (Delta connection introducing a $+30$ degrees phase-shift). Default is Y. For an explanation of the phase shifts introduced by D1 and D11 connections, see the Three-Phase Transformer (Two-Windings) block documentation. The zero-sequence impedance will vary according to the connection. Zero-sequence impedance is computed assuming that the three-phase transformer bank is built with three single-phase, two-winding units.

Winding 2 connection (abc terminals)

The connection of the three windings 2. Select Y (ungrounded wye), Yg (wye grounded), Delta(D1) (Delta connection introducing a -30 degrees phase-shift) or Delta(D11) (Delta connection introducing a $+30$ degrees phase-shift). Default is Delta(D1). The zero-sequence impedance will vary according to the connection.

Nominal Power and frequency

The three-phase nominal power, in VA and the nominal frequency, in hertz. Default is [47e6, 60].

Winding 1 parameters

The nominal line to line voltage of winding 1, in volts rms, winding resistance, in pu, and leakage reactance, in pu. Default is [120e3, 0.003, 0.09].

Winding 2 parameters

The nominal line to line voltage of winding 2, in volts rms, winding resistance in pu, and leakage reactance, in pu. Default is [25e3, 0.003, 0.09].

Initial pos. seq. output currents of winding 2

The initial value of the positive-sequence current phasor (Magnitude in pu and Phase in degrees) flowing out of winding 2. The currents flowing into winding 1 will be computed accordingly. If you know the initial value of currents you may specify it in order to start simulation in steady state. If you don't know these values, you can leave [0 0]. The system will reach steady-state after a short transient. Default is [0.7568, -45].

Magnetizing branch

The resistance, in pu and the reactance, in pu, of the parallel R_m , X_m branch modelling respectively iron losses and reactive magnetizing currents. Saturation is not modelled. Default is [300, 300].

OLTC and Voltage Regulator Tab

OLTC on

Select either Winding 1 (default) or Winding 2.

Tapped regulation winding

The resistance and the leakage reactance of the tapped winding depend on the tap position. Enter the resistance, in pu, and the leakage reactance, in pu, corresponding to the maximum tap position. Default is [0.003, 0.09].

Voltage step per tap

The voltage step per tap, in pu of the nominal voltage of winding 1 or winding 2. Default is 0.01875.

Minimum and maximum tap positions

You normally specify a negative integer corresponding to maximum tap position with the regulation winding connected in subtractive polarity, followed by a positive integer corresponding to maximum tap position with the regulation winding connected in additive polarity. These two integers specify the number of taps and their absolute values are usually the same. However, if you want to limit tap excursion, you may specify two different values. Default is [-8, +8].

Initial tap position

Specify an integer N corresponding to initial tap position with $\text{MinTap} \leq N \leq \text{MaxTap}$. Default is -4.

Tap selection time

The mechanical time delay, in seconds, required for the OLTC to move the taps by one position. Typical values are in the 3s-10s range. Default is 4.

External control of OLTC

The OLTC is usually controlled by the voltage regulator. Select this parameter if you want to implement your own control system or manually control the tap position. The Vm input will then be replaced by two inputs labeled Up and Down. Applying a pulse to the Up or Down input will move the tap position upward or downward when the signal is changing from 0 to 1. Default is cleared.

Voltage regulator

This parameter is not visible when the **External control of OLTC** parameter is selected.

Select on (default) for enabling the voltage regulator. If you select off the OLTC is frozen at the position specified by the **Initial tap position** parameter.

Regulator parameters

This parameter is not visible when the **External control of OLTC** parameter is checked.

Specify the reference voltage Vref, in pu, the dead band of the hysteresis regulator, in pu, and the time delay, in seconds, during which the voltage error must exceed half the dead band for the OLTC to move tap position upward or downward. The dead band is normally adjusted at two times the voltage step specified in the **Voltage step per tap** parameter. Default is [1.04, 0.0375, 1].

Inputs and Outputs

A B C

The three input terminals connected windings 1.

a b c

The three output terminals connected to windings 2.

V_m

This input is visible only when the **External control of OLTC** parameter is not checked.

Connect to this input the measured voltage to be controlled by the voltage regulator. This signal is usually the magnitude, in pu, of the positive-sequence voltage measured at the transformer terminals (side 1 or side 2) or at a remote bus.

Up

This input is visible only when the **External control of OLTC** parameter is checked.

Simulink input for controlling tap position. Applying a pulse to this input will initiate upward tap changing when the pulse is changing from 0 to 1.

Down

This input is visible only when the **External control of OLTC** parameter is checked.

Simulink input for controlling tap position. Applying a pulse to this input will initiate downward tap changing when the pulse is changing from 0 to 1.

m

Simulink output vector containing 16 internal signals. These signals are either complex signals (voltage phasors, current phasors or impedances) or control signals. They can be individually accessed by using the Bus Selector block. They are, in order:

Signal	Signal Group	Signal Names	Definition
1-3	VABC (cmplx)	VA (pu) VB (pu) VC (pu)	Phasor voltages (phase to ground) at the transformer input terminals A, B, C (pu)
4-6	Vabc (cmplx)	Va (pu) Vb (pu) Vc (pu)	Phasor voltages (phase to ground) at the transformer output terminals a, b, c (pu)
7-9	IABC (cmplx)	IA (pu) IB (pu) IC (pu)	Phasor currents flowing into the input terminals A, B, C
10-12	Iabc (cmplx)	Ia (pu) Ib (pu) Ic (pu)	Phasor currents flowing out of the output terminals a, b, c
13-14	Z (cmplx)	Z1 (pu) Z0 (pu)	Positive- and zero-sequence complex impedances (R+jX)
15	Tap	Tap	Tap position
16	Ready	Ready	Logical signal generated by the tap changer controller. This signal is used to enable voltage controller action. The Ready signal becomes (1) after the tap selection has been completed, thus enabling a new tap change. The Up and Down pulses to the OLTC are blocked as long as the Ready signal is (0).

Examples

See the “OLTC Regulating Transformer Phasor Model” example, which illustrates the use of the Three-Phase OLTC Regulator Transformer (Phasor Type) block to regulate voltage on a 25-kV distribution feeder. The phasor model is compared with a detailed model of regulating transformer. Look under the masked blocks to see how both models are implemented. The detailed model uses switches and three Multi-Winding Transformer blocks, whereas the phasor model uses current sources.

See Also

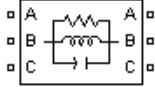
Three-Phase OLTC Phase Shifting Transformer Delta-Hexagonal (Phasor Type)

Version History

Introduced in R2006a

Three-Phase Parallel RLC Branch

Implement three-phase parallel RLC branch



Library

Simscape / Electrical / Specialized Power Systems / Passives

Description

The Three-Phase Parallel RLC Branch block implements three balanced branches consisting each of a resistor, an inductor, a capacitor, or a parallel combination of these. Use the **Branch type** parameter to select elements you want to include in each branch.

Negative values are allowed for resistance, inductance, and capacitance.

Parameters

Branch type

Select the elements you want to include in the branch. The **R** letter defines the resistor, the **L** letter defines the inductor, and the **C** letter defines the capacitor. Select **Open circuit** to define an open circuit ($R=\infty$, $L=\infty$, $C=0$). Only existing elements are displayed in the block icon. Default is RLC.

Resistance R

The branch resistances, in ohms (Ω). Default is 1.

Inductance L

The branch inductances, in henries (H). Default is $1e-3$.

Capacitance C

The branch capacitances, in farads (F). Default is $1e-6$.

Measurements

Select **Branch voltages** to measure the three voltages across the Three-Phase Parallel RLC Branch block terminals.

Select **Branch currents** to measure the three total currents (sum of R, L, C currents) flowing through the Three-Phase Parallel RLC Branch block.

Select **Branch voltages and currents** to measure the three voltages and the three currents of the Three-Phase Parallel RLC Branch block.

Default is None.

Place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurements** list box of the Multimeter block, the measurements are identified by a label followed by the block name.

Measurement	Label
Branch voltages	Ua :
Branch currents	Ia :

See Also

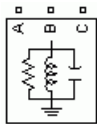
Multimeter, Three-Phase Parallel RLC Load, Three-Phase Series RLC Branch, Three-Phase Series RLC Load

Version History

Introduced before R2006a

Three-Phase Parallel RLC Load

Implement three-phase parallel RLC load with selectable connection



Library

Simscape / Electrical / Specialized Power Systems / Passives

Description

The Three-Phase Parallel RLC Load block implements a three-phase balanced load as a parallel combination of RLC elements. At the specified frequency, the load exhibits a constant impedance. The active and reactive powers absorbed by the load are proportional to the square of the applied voltage.

Only elements associated with nonzero powers are displayed in the block icon.

Parameters

Parameters Tab

Configuration

The connection of the three phases. Select one of the following four connections:

Y(grounded)	Neutral is grounded. Default
Y(floating)	Neutral is not accessible.
Y(neutral)	Neutral is made accessible through a fourth connector.
Delta	Three phases connected in delta

The block icon is updated according to the load connection.

Nominal phase-to-phase voltage Vn

The nominal phase-to-phase voltage of the load, in volts RMS (Vrms). Default is 1000.

Nominal frequency fn

The nominal frequency, in hertz (Hz). Default is 60.

Specify PQ powers for each phase

Select to specify active powers, inductive reactive powers, and capacitive reactive powers for each phase of the load. Default is cleared.

Active power P (W)

The three-phase active power of the load, in watts. This parameter is visible only when **Specify PQ powers for each phase** is cleared. Default is 10e3.

Inductive reactive power QL (positive var)

The three-phase inductive reactive power QL, in vars. Specify a positive value or 0. This parameter is visible only when **Specify PQ powers for each phase** is cleared. Default is 100.

Capacitive reactive power QC (negative var)

The three-phase capacitive reactive power QC, in vars. Specify a positive value or 0. This parameter is visible only when **Specify PQ powers for each phase** is cleared. Default is 100.

Active powers [Pa Pb Pc] (W):

The three-phase active power of the load, in watts. Specify three positive values. You can specify different powers for each phase, including 0. This parameter is visible only when you select **Specify PQ powers for each phase**. Default is [10e3 9e3 11e3].

Inductive reactive powers [QLa QLb QLc] (positive var):

The three-phase inductive reactive power QL, in vars. Specify three positive values. You can specify different powers for each phase, including 0. This parameter is visible only when you select **Specify PQ powers for each phase**. Default is [100 90 110].

Capacitive reactive powers [QCa QCb QCc] (negative var):

The three-phase capacitive reactive power QC, in vars. Specify three positive values. You can specify different powers for each phase, including 0. This parameter is visible only when you select **Specify PQ powers for each phase**. Default is [100 90 110].

Measurements

Select **Branch voltages** to measure the three voltages across each phase of the Three-Phase Parallel RLC Load block terminals. For a Y connection, these voltages are the phase-to-ground or phase-to-neutral voltages. For a delta connection, these voltages are the phase-to-phase voltages.

Select **Branch currents** to measure the three total currents (sum of R, L, C currents) flowing through each phase of the Three-Phase Parallel RLC Load block. For a delta connection, these currents are the currents flowing in each branch of the delta.

Select **Branch voltages** and **currents** to measure the three voltages and the three currents of the Three-Phase Parallel RLC Load block.

Default is None.

Place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurements** list box of the Multimeter block, the measurements are identified by a label followed by the block name.

Measurement		Label
Branch voltages	Y(grounded)	Uag:
	Y(floating)	Uan:
	Y(neutral)	Uan:
	Delta	Uab:
Branch currents	Y(grounded)	Iag:
	Y(floating)	Ian:
	Y(neutral)	Ian:
	Delta	Iab:

Load Flow Tab

The parameters on this tab are used by the Load Flow tool of the Powergui block. These load flow parameters are used for model initialization only, they have no impact on the block model and on the simulation performance.

Load type

Specify the load type.

If you select **constant Z** (default), the load impedance is determined from the nominal phase-to-phase voltage V_n , active power P , and reactive power (QL-QC) specified on the **Parameters** tab of the block dialog box. During the load flow solution, the impedance is kept constant. The effective P and Q are therefore varying proportionally to the square of the bus voltage computed by the Load Flow tool.

If you select **constant PQ**, the active power P and reactive power Q are kept constant and equal to the values specified on the **Parameters** tab of the block dialog box. When you apply the load flow solution to the model (by clicking the **Apply** button of the Load Flow tool), the **Nominal phase-to-phase voltage V_n** parameter on the **Parameters** tab is automatically adjusted to the phase-to-phase bus voltage computed by the Load Flow tool.

If you select **constant I**, the load current is kept constant at its nominal value determined from base voltage specified in the dialog box of the associated Load Flow Bus block and from active power P and reactive power Q_L and Q_C specified on the **Parameters** tab of the block dialog box. The effective P and Q are therefore varying proportionally to the bus voltage computed by the Load Flow tool. When you apply the load flow solution to the model (by clicking the **Apply** button of the Load Flow tool), the p.u. value of the nominal voltage V_n on the **Parameters** tab is automatically adjusted to the square root of the load voltage (in pu) computed by the Load Flow tool.

See Also

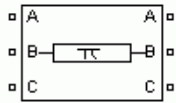
Multimeter, Three-Phase Dynamic Load, Three-Phase Parallel RLC Branch, Three-Phase Series RLC Branch, Three-Phase Series RLC Load

Version History

Introduced before R2006a

Three-Phase PI Section Line

Implement three-phase transmission line section with lumped parameters



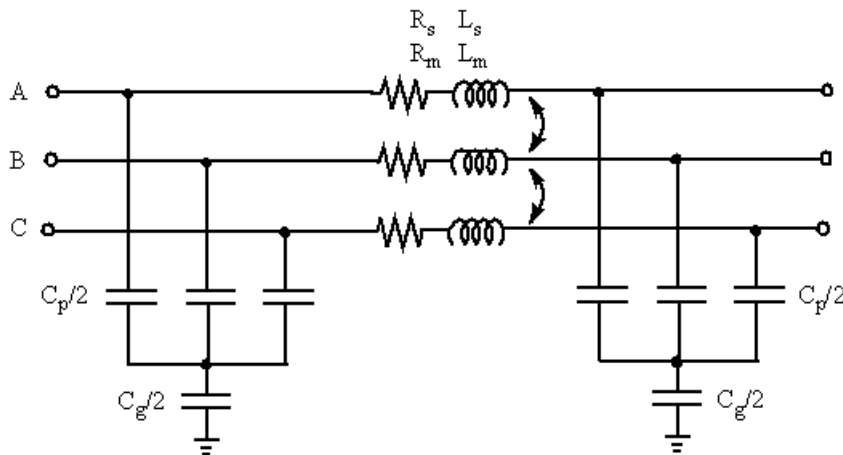
Libraries:

Simscape / Electrical / Specialized Power Systems / Power Grid Elements

Description

The Three-Phase PI Section Line block implements a balanced three-phase transmission line model with parameters lumped in a PI section.

Contrary to the Distributed Parameters Line model where the resistance, inductance, and capacitance are uniformly distributed along the line, the Three-Phase PI Section Line block lumps the line parameters in a single PI section as shown in the figure below.



The line parameters R , L , and C are specified as positive- and zero-sequence parameters that take into account the inductive and capacitive couplings between the three phase conductors, as well as the ground parameters. This method of specifying line parameters assumes that the three phases are balanced.

The self and mutual resistances (R_s , R_m), self and mutual inductances (L_s , L_m) of the three coupled inductors, as well as phase capacitances C_p and ground capacitances C_g , are deduced from the positive- and zero-sequence RLC parameters as follows.

Let us assume the following line parameters:

r_1, r_0	Positive- and zero-sequence resistances per unit length (Ω/km)
l_1, l_0	Positive- and zero-sequence inductances per unit length (H/km)
c_1, c_0	Positive- and zero-sequence capacitances per unit length (F/km)

f	Frequency (Hz)
$lsec$	Line section length (km)

The total positive- and zero-sequence RLC parameters including hyperbolic corrections are first evaluated:

$$R_1 = r_1 \cdot lsec \cdot k_{r1}$$

$$L_1 = l_1 \cdot lsec \cdot k_{l1}$$

$$C_1 = c_1 \cdot lsec \cdot k_{c1}$$

$$R_0 = r_0 \cdot lsec \cdot k_{r0}$$

$$L_0 = l_0 \cdot lsec \cdot k_{l0}$$

$$C_0 = c_0 \cdot lsec \cdot k_{c0}$$

where

k_{r1} , k_{l1} , k_{c1} , k_{r0} , k_{l0} , k_{c0} — positive-sequence and zero-sequence hyperbolic correction factors

Note See the PI Section Line block reference page for an explanation on how to compute RLC parameters taking into account hyperbolic corrections.

The Powergui block provides a graphical tool for the calculation of resistance, inductance, and capacitance per unit length based on the line geometry and the conductor characteristics. See the `power_lineparam` reference page to learn how to use this tool.

For a short line section (approximately $lsec < 50$ km), these correction factors are negligible (close to unity). However, for long lines, these hyperbolic corrections must be taken into account in order to get an exact line model at the specified frequency.

The RLC line section parameters are then computed as follows:

$$R_s = (2R_1 + R_0)/3$$

$$L_s = (2L_1 + L_0)/3$$

$$R_m = (R_0 - R_1)/3$$

$$L_m = (L_0 - L_1)/3$$

$$C_p = C_1$$

$$C_g = 3C_1C_0/(C_1 - C_0)$$

Ports

Conserving

A — Phase A terminal
specialized electrical

Specialized electrical conserving port associated with the terminal of phase A.

B — Phase B terminal
specialized electrical

Specialized electrical conserving port associated with the terminal of phase B.

C — Phase C terminal
specialized electrical

Specialized electrical conserving port associated with the terminal of phase C.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Line length (km) — Line section length
100 (default) | positive scalar

Line section length in kilometers (km).

Frequency used for rlc specification (Hz) — Frequency used for specification of per unit length rlc line parameters
60 (default) | positive scalar

Frequency used for specification of per unit length rlc line parameters, in hertz (Hz). This is usually the nominal system frequency (50 Hz or 60 Hz).

Positive- and zero-sequence resistances (Ohms/km) [r1 r0] — Positive- and zero-sequence resistances
[0.01273 0.3864] (default) | vector of two positive elements

Positive- and zero-sequence resistances in ohms/kilometer (Ω/km).

Positive- and zero-sequence inductances (H/km) [l1 l0] — Positive- and zero-sequence inductances
[0.9337e-3 4.1264e-3] (default) | vector of two positive elements

Positive- and zero-sequence inductances in henries/kilometer (H/km). The zero-sequence inductance cannot be zero, because it would result in an invalid propagation speed computation.

Positive- and zero-sequence capacitances (F/km) [c1 c0] — Positive- and zero-sequence capacitances
[12.74e-9 7.751e-9] (default) | vector of two positive elements

Positive- and zero-sequence capacitances in farads/kilometer (F/km). The zero-sequence capacitance cannot be zero, because it would result in an invalid propagation speed computation.

Version History

Introduced before R2006a

Extended Capabilities

C/C++ Code Generation

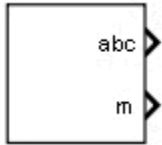
Generate C and C++ code using Simulink® Coder™.

See Also

Distributed Parameters Line | PI Section Line

Three-Phase Programmable Generator

Generate three-phase signal with programmable time variation of amplitude, phase, frequency, and harmonics



Library

Simscape / Electrical / Specialized Power Systems / Sources / Signal Generator Sources

Description

Use the Three-Phase Programmable Generator block to generate a three-phase sinusoidal signal with time-varying parameters. This vectorized signal usually represents a set of three-phase voltages or currents. You can program the time variation for the amplitude, phase, or frequency of the fundamental component of the signal. In addition, you can program two harmonics and superimpose them on the fundamental signal.

Parameters

Positive-sequence [Amplitude, Phase (degrees), Freq (Hz)]

Specify the amplitude (in volts, amperes, or pu), the phase (in degrees), and the frequency (in hertz) of the positive-sequence component of the three-phase signal. Default is [100, 0, 60].

Time variation of

Specify whether you want to program the time variation of the amplitude, phase, or frequency. If you do not want to program a generator time variation, select **None** (default).

The time variation applies to the three phases of the source except when the **Type of variation** parameter is set to **Table of amplitude-pairs**. In that case, you can apply a variation to phase A only.

Type of variation

Specify the type of variation to apply to the amplitude, phase, or frequency specified by the **Time variation of** parameter. Options include **Step**, **Ramp** (default), **Modulation** (for sinusoidal modulation), and **Table of amplitude-pairs** (a series of step changes of amplitude or phase at specific times).

Step magnitude

This parameter is visible only when the **Type of Variation** parameter is set to **Step**.

Specify the amplitude of the step change of the magnitude of a positive-sequence amplitude (in volts, amperes, or pu), phase (in degrees), or frequency (in hertz). Default is -0.5 .

Rate of change (value/s)

This parameter is visible only when the **Type of Variation** parameter is set to **Ramp**.

Specify the rate of change of amplitude (in volts/second, amperes/second, or pu/second), phase (in degrees/second), or frequency (in hertz/second). Default is 10.

Amplitude of the modulation

This parameter is visible only when the **Type of Variation** parameter is set to Modulation.

Specify the amplitude of the modulation for the amplitude, frequency, or phase, as specified in the **Time variation of** parameter. Default is 0.3.

Frequency of the modulation (Hz)

This parameter is visible only when the **Type of Variation** parameter is set to Modulation.

Specify the frequency of the modulation for the amplitude, phase, or frequency, as specified in the **Time variation of** parameter. Default is 2.

Variation timing (s) [Start, End]

This parameter is visible only when the **Type of Variation** parameter is set to Step, Ramp, or Modulation.

Specify the time, in seconds, when the programmed time variation takes effect and stops. Default is [0.2 1.2].

Variation on phase A only

This parameter is visible only when the **Type of Variation** parameter is set to Table of time-amplitude pairs.

When this parameter is selected, the time variation specified by the **Amplitude or phase values** and **Time values** parameters is applied on phase A only. Default is selected.

Amplitude or phase values (degrees)

This parameter is visible only when the **Type of Variation** parameter is set to Table of time-amplitude pairs.

Use this parameter to specify the amplitude or phase values of phase A when **Variation on phase A only** is selected. Otherwise, use it to specify the positive sequence amplitudes or phase values (in degrees) corresponding to times specified in the **Time values** parameter. Default is [1 0.2 1.2 1.0].

Time values

This parameter is visible only when the **Type of Variation** parameter is set to Table of time-amplitude pairs.

Specify the times, in seconds, when the amplitude or phase will change, according to the values specified in the **Amplitude or phase values** parameter. Default is [0 0.1 0.15 0.2].

Harmonic generation

Select this check box to enable programming of two harmonics that can be superimposed on the fundamental voltage of the source. Default is cleared.

A: [Order (n), Amplitude, Phase (degrees), Seq(0, 1 or 2)]

This parameter is visible only when the **Harmonic generation** check box is selected.

Specify the order, amplitude, phase, and the type of sequence (1 = positive-sequence; 2 = negative-sequence; 0 = zero-sequence) of the first harmonic to superimpose on the fundamental signal. Default is [3 5 -25 0].

Specify 1 for the harmonic order and 1 for the sequence to produce a positive-sequence amplitude change without imbalance and without harmonics.

Specify 1 for the harmonic order and 0 or 2 for the sequence to produce a signal imbalance without harmonics.

B: [Order (n), Amplitude, Phase (degrees), Seq(0, 1 or 2)]

This parameter is visible only when the Harmonic generation check box is selected. Specify the order, amplitude, phase, and the type of sequence (1 = positive-sequence; 2 = negative-sequence; 0 = zero-sequence) of the second harmonic to superimpose on the fundamental signal. Default is [5 10 35 2].

Harmonic timing (s) [Start, End]

This parameter is visible only when the **Harmonic generation** check box is selected.

Specify the time, in seconds, when the harmonic generation is superimposed on the fundamental signal and the time when it stops. Default is [0.05 99].

Sample time

Specify the sample time of the block, in seconds. Set to 0 to implement a continuous block. Default is 0.

Inputs and Outputs

abc

Returns the vectorized signal of phases a, b, and c.

m

Simulink output vector containing eight internal signals.

Signal	Signal Names	Definition
1-3	Mag_abc	Magnitudes of the fundamental component of a, b, c signals
4-6	Phase_abc (deg)	Phases of the fundamental component of a, b, c signals (degrees)
7	Freq (Hz)	Frequency of the fundamental component of a, b, c signals (hertz)
8	wt (rad)	Instantaneous angle of the rotating vector of the positive-sequence component (radians)

Characteristics

Sample Time	- Specified in the Sample Time parameter - Continuous if Sample Time = 0
Scalar Expansion	No
Dimensionalized	No

Examples

The `power_ThreePhaseProgrammableGenerator` model shows two applications of the Three-Phase Programmable Generator block.

In the top model, a sinusoidal modulation of amplitude is programmed from 0.1 s to 0.6 s. Then a positive-sequence signal sag is produced from 0.7 s to 0.9 s by adding a positive-sequence harmonic of order 1 with a 180-degree phase-shift.

In the bottom model, a frequency ramp is programmed from 0.1 s to 0.6 s, and then two harmonics (2nd and 5th) are added from 0.7 s to 0.9 s. The magnitude, phase, and frequency of positive-sequence component are measured using a PLL and a Positive-Sequence (PLL-Driven) block. Variations of measured quantities are compared with specified values available at the measurement output (m).

The model sample time is parameterized with variable T_s (default value $T_s=50e-6$). To simulate the continuous Three-Phase Programmable Generator block, specify $T_s = 0$ in the MATLAB Command Window before starting the simulation.

See Also

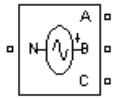
Three-Phase Programmable Voltage Source, Three-Phase Sine Generator

Version History

Introduced in R2013a

Three-Phase Programmable Voltage Source

Implement three-phase voltage source with programmable time variation of amplitude, phase, frequency, and harmonics



Libraries:

Simscape / Electrical / Specialized Power Systems / Sources

Description

Use this block to generate a three-phase sinusoidal voltage with time-varying parameters. You can program the time variation for the amplitude, phase, or frequency of the fundamental component of the source. In addition, two harmonics can be programmed and superimposed on the fundamental signal.

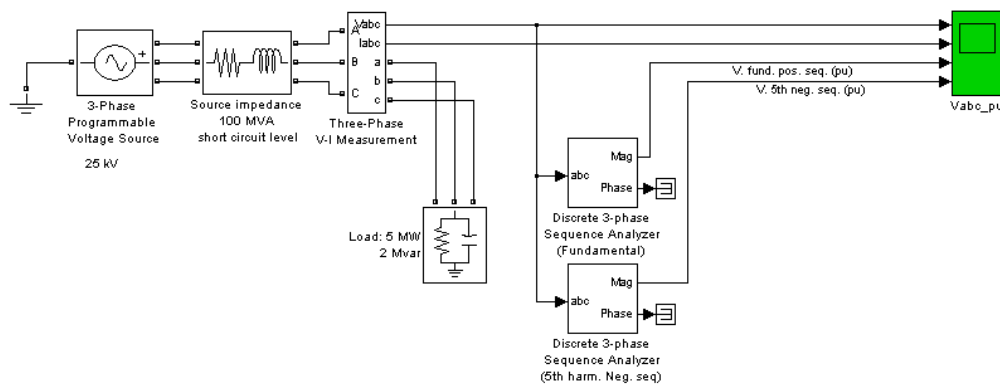
Examples

The “Three-Phase Programmable Source, V-I Measurement and Sequence Analyzer” circuit illustrates the use of the Three-Phase Programmable Voltage Source block to produce a step variation of the positive-sequence voltage and to inject harmonics into the circuit.

A 25 kV, 100 MVA short-circuit level, equivalent network feeds a 5 MW, 2 Mvar capacitive load. The internal voltage of the source is controlled by the Discrete 3-phase Programmable Voltage Source block.

A positive sequence of 1.0 pu, 0 degrees is specified for the fundamental signal. At $t = 0.05$ s a step of 0.5 pu is applied to the positive-sequence voltage magnitude, then at $t = 0.1$ s, 0.08 pu of fifth harmonic in negative sequence is added to the 1.5 pu voltage.

The three-phase voltage and current are measured at the output of the source impedance. Two Discrete Sequence Analyzer blocks are used to measure the positive-sequence fundamental component and the negative-sequence fifth harmonic of the three-phase voltage.



Ports

Conserving

N — Neutral terminal
specialized electrical

Electrical conserving port associated with the neutral terminal.

A — Phase A terminal
specialized electrical

Electrical conserving port associated with the phase A terminal.

B — Phase B terminal
specialized electrical

Electrical conserving port associated with the phase B terminal.

C — Phase C terminal
specialized electrical

Electrical conserving port associated with the phase C terminal.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Parameters

Positive-sequence — Parameters of positive-sequence component
[100 0 60] (default) | vector of scalars

The amplitude in volts RMS phase-to-phase, the phase in degrees, and the frequency in hertz of the positive-sequence component of the three voltages. Default is [100 0 60].

Phase B and phase C internal voltages lag phase A by 120 degrees and 240 degrees, respectively ([0 -120 +120]).

Time variation of — Parameter for which you want to program time variation
None (default) | Amplitude | Phase | Frequency

Specify the parameter for which you want to program the time variation. Select **None** (default) if you do not want to program the time variation of the source parameters. Select **Amplitude** if you want to program the time variation of the amplitude. Select **Phase** if you want to program the time variation of the phase. Select **Frequency** if you want to program the time variation of the frequency.

The time variation applies to the three phases of the source except when the **Type of variation** parameter is set to **Table of amplitude-pairs**. In that case you can apply a variation to phase A only.

Type of variation — Type of variation
Table of amplitude-pairs (default) | Step | Ramp | Modulation

Specify the type of variation that is applied to the parameter specified by the **Time variation of** parameter. Select **Step** to program a step variation. Select **Ramp** to program a ramp variation. Select **Modulation** to program a modulated variation. Select **Table of amplitude-pairs** to program a series of step changes of amplitudes at specific times.

The **Table of amplitude-pairs** option is available only when the **Time variation of** parameter is set to **Amplitude**.

Table of amplitude-pairs is the default when **Time variation of** is set to **Amplitude**. **Step** is the default when **Time variation of** is set to **Phase** or **Frequency**.

Dependencies

To enable this parameter, set **Time variation of** to **Amplitude**, **Phase**, or **Frequency**.

Variation on phase A only — Variation on phase A only
off (default) | on

If selected, the programmed time variation of the amplitude is applied only to phase A of the Three-Phase Programmable Voltage Source block.

Dependencies

To enable this parameter, set **Time variation of** to **Amplitude** and **Type of variation** to **Table of amplitude-pairs**.

Amplitude values (pu) — Amplitude variation points
[1 0.8 1.2 1.0] (default) | vector of positive values

Specifies the amplitude variation points taken by the voltage source during the simulation. The specified values are in pu, based on the positive-sequence amplitude of the voltage source. The corresponding times where the variation will take place are defined by the **Time values** parameter.

Dependencies

To enable this parameter, set **Time variation of** to **Amplitude** and **Type of variation** to **Table of amplitude-pairs**.

Time values — Transition times
[0 0.1 0.15 0.2] (default) | vector of positive values

Specifies the transition times, in seconds, of the amplitude variation defined in the **Amplitude values (pu)** parameter.

Dependencies

To enable this parameter, set **Time variation of** to **Amplitude** and **Type of variation** to **Table of amplitude-pairs**.

Step magnitude — Amplitude of step change
0.5 (default) | nonnegative scalar

Specify the amplitude of the step change. The variation of amplitude is specified in pu of the positive-sequence amplitude.

Dependencies

To enable this parameter, set **Type of variation** to **Step**.

Rate of change — Rate of change

10 (default) | positive scalar

Specify the rate of change, in volt/seconds. The rate of change of voltage is specified in (pu of the positive-sequence voltage)/second.

Dependencies

To enable this parameter, set **Type of variation** to Ramp.

Amplitude of the modulation — Amplitude of modulation

0.3 (default) | positive scalar

Specify the amplitude of the modulation for the source parameter that is specified in the **Time variation of** parameter. When the varying quantity is the voltage amplitude, the amplitude of the modulation is specified in pu of the positive-sequence amplitude.

Dependencies

To enable this parameter, set **Type of variation** to Modulation.

Frequency of the modulation — Frequency of modulation

2 (default) | positive scalar

Specify the frequency of the modulation for the source parameter that is specified in the **Time variation of** parameter.

Dependencies

To enable this parameter, set **Type of variation** to Modulation.

Variation timing (s) — Variation timing

[1 2] (default) | vector of positive values

Specify the time, in seconds, when the programmed time variation takes effect and the time when it stops.

Fundamental and/or Harmonic generation — Fundamental and/or harmonic generation

off (default) | on

If selected, two harmonics can be programmed to be superimposed on the fundamental voltage of the source.

A: [Order Amplitude Phase Seq] — Order, amplitude, phase, and type of sequence of first harmonic

[3 0.2 -25 0] (default) | vector of nonnegative values

Specify the order, amplitude, phase, and the type of sequence (1 = positive-sequence; 2 = negative-sequence; 0 = zero-sequence) of the first harmonic to be superimposed on the fundamental signal. The voltage of the harmonic is specified in pu of the positive-sequence voltage.

Specify 1 for the harmonic order and 0 or 2 for the sequence to produce a voltage imbalance without harmonics.

Dependencies

To enable this parameter, select the **Fundamental and/or Harmonic generation** parameter.

B: [Order Amplitude Phase Seq] — Order, amplitude, phase, and type of sequence of second harmonic

[2 0.15 35 2] (default) | vector of nonnegative values

Specify the order, amplitude, phase, and the type of sequence (1 = positive-sequence; 2 = negative-sequence; 0 = zero-sequence) of the second harmonic to be superimposed on the fundamental signal. The voltage of the harmonic is specified in pu of the positive-sequence voltage.

Specify 1 for the harmonic order and 0 or 2 for the sequence to produce a voltage imbalance without harmonics.

Dependencies

To enable this parameter, select the **Fundamental and/or Harmonic generation** parameter.

Timing (s) — Time when harmonic generation is superimposed

[0.05 3] (default) | vector of positive values

Specify the time, in seconds, when the harmonic generation is superimposed on the fundamental signal and the time when it stops.

Dependencies

To enable this parameter, select the **Fundamental and/or Harmonic generation** parameter.

Load Flow

The load flow parameters are used to define block parameters for use with the Load Flow tool of the Powergui block. These load flow parameters are used for model initialization only. They have no impact on the block model and on the simulation's performance.

The configuration of the **Load Flow** tab depends on the option selected for the **Generator type** parameter.

Generator type — Generator type of voltage source

swing (default) | PV | PQ

Specify the generator type of the voltage source.

Select swing (default) to implement a generator controlling magnitude and phase angle of its terminal voltage. The reference voltage magnitude and angle are specified by the **Swing bus or PV bus voltage** and **Swing bus voltage angle** parameters of the Load Flow Bus block connected to the voltage source terminals.

Select PV to implement a generator controlling its output active power P and voltage magnitude V. P is specified by the **Active power generation P** parameter of the block. V is specified by the **Swing bus or PV bus voltage** parameter of the Load Flow Bus block connected to the voltage source terminals. You can control the minimum and maximum reactive power generated by the block by using the **Minimum reactive power Qmin** and **Maximum reactive power Qmax** parameters.

Select PQ to implement a generator controlling its output active power P and reactive power Q. P and Q are specified by the **Active power generation P** and **Reactive power generation Q** parameters of the block, respectively.

Active power generation P — Active power generated by source

0 (default) | nonnegative scalar

Specify the active power that you want generated by the source, in watts.

Dependencies

To enable this parameter, set **Generator type** to PV or PQ.

Reactive power generation Q — Reactive power generated by source
0 (default) | nonnegative scalar

Specify the reactive power that you want generated by the source, in vars.

Dependencies

To enable this parameter, set **Generator type** to PQ.

Minimum reactive power Qmin — Minimum reactive power generated by the source
-inf (default) | scalar

Minimum reactive power that can be generated by the source while keeping the terminal voltage at its reference value. This reference voltage is specified in the **Swing bus or PV bus voltage** parameter of the Load Flow Bus block connected to the source terminals. The default value is -inf, which means that there is no lower limit on the reactive power output.

Dependencies

To enable this parameter, set **Generator type** to PV.

Maximum reactive power Qmax — Maximum reactive power generated by the source
inf (default) | scalar

Maximum reactive power that can be generated by the source while keeping the terminal voltage at its reference value. This reference voltage is specified in the **Swing bus or PV bus voltage** parameter of the Load Flow Bus block connected to the source terminals. The default value is inf, which means that there is no upper limit on the reactive power output.

Dependencies

To enable this parameter, set **Generator type** to PV.

Version History

Introduced before R2006a

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

Three-Phase Source

Three-Phase Sequence Analyzer

Measure positive-, negative-, and zero-sequence components of three-phase signal



Library

powerlib_extras/Measurements, powerlib_extras/Discrete Measurements

Note The Measurements section of the Control and Measurements library contains the Sequence Analyzer block. This is an improved version of the Three-Phase Sequence Analyzer block. The new block features a mechanism that eliminates duplicate continuous and discrete versions of the same block by basing the block configuration on the simulation mode. If your legacy models contain the Three-Phase Sequence Analyzer block, they continue to work. However, for best performance, use the Sequence Analyzer block in your new models.

Description

The Three-Phase Sequence Analyzer block outputs the magnitude and phase of the positive- (denoted by the index 1), negative- (index 2), and zero-sequence (index 0) components of a set of three balanced or unbalanced signals. The signals can contain harmonics or not. The three sequence components of a three-phase signal (voltages V_1 V_2 V_0 or currents I_1 I_2 I_0) are computed as follows:

$$V_1 = (V_a + aV_b + a^2V_c)/3 \quad V_2 = (V_a + a^2V_b + aV_c)/3 \quad V_0 = (V_a + V_b + V_c)/3$$

where

V_a, V_b, V_c = three voltage phasors at specified frequency
 $a = e^{j2\pi/3} = 1 \angle 120^\circ$ complex operator.

A Fourier analysis over a sliding window of one cycle of the specified frequency is first applied to the three input signals. It evaluates the phasor values V_a, V_b , and V_c at the specified fundamental or harmonic frequency. Then the transformation is applied to obtain the positive sequence, negative sequence, and zero sequence.

The Three-Phase Sequence Analyzer block is not sensitive to harmonics or imbalances. However, as this block uses a running average window to perform the Fourier analysis, one cycle of simulation has to be completed before the outputs give the correct magnitude and angle. For example, its response to a step change of V_1 is a one-cycle ramp.

The discrete version of this block allows you to specify the initial magnitude and phase of the output signal. For the first cycle of simulation, the outputs are held to the values specified by the initial input parameter.

You can modify any parameter during the simulation in order to obtain the different sequence and harmonic components of the input signals.

Parameters

Fundamental frequency f1

The fundamental frequency, in hertz, of the three-phase input signal.

Harmonic n

Specify the harmonic component from which you want to evaluate the sequences. For DC, enter 0. For fundamental, enter 1.

Sequence

Specify which sequence component the block outputs. Select **Positive** to calculate the positive sequence, select **Negative** to calculate the negative sequence, select **0** to compute the zero sequence of the fundamental or specified harmonic of the three-phase input signal. Select **Positive Negative Zero** to get all the sequences.

Inputs and Outputs

abc

Connect to the input the vectorized signal of the three [a b c] sinusoidal signals.

Mag

The first output gives the magnitude (peak value) of the specified sequence component, in the same units as the abc input signals.

Phase

The second output gives the phase in degrees of the specified component(s).

Examples

The “Three-Phase Programmable Source, V-I Measurement and Sequence Analyzer” example illustrates the use of the Sequence Analyzer block (the improved version of the Three-Phase Sequence Analyzer block).

Version History

Introduced before R2006a

See Also

Sequence Analyzer

Three-Phase Series RLC Branch

Implement three-phase series RLC branch



Libraries:

Simscape / Electrical / Specialized Power Systems / Passives

Description

The Three-Phase Series RLC Branch block implements three balanced branches consisting each of a resistor, an inductor, or a capacitor or a series combination of these. Use the **Branch type** parameter to select elements you want to include in each branch.

Negative values are allowed for resistance, inductance, and capacitance.

Ports

Conserving

A — Phase A terminal 1

specialized electrical

Specialized electrical conserving port associated with the terminal 1 of phase A.

B — Phase B terminal 1

specialized electrical

Specialized electrical conserving port associated with the terminal 1 of phase B.

C — Phase C terminal 1

specialized electrical

Specialized electrical conserving port associated with the terminal 1 of phase C.

a — Phase A terminal 2

specialized electrical

Specialized electrical conserving port associated with the terminal 2 of phase A.

b — Phase B terminal 2

specialized electrical

Specialized electrical conserving port associated with the terminal 2 of phase B.

c — Phase C terminal 2

specialized electrical

Specialized electrical conserving port associated with the terminal 2 of phase C.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Branch type — Elements to include in branch

RLC (default) | R | L | C | RL | RC | LC | Open circuit

Elements you want to include in the branch. The **R** letter defines the resistor, the **L** letter defines the inductor, and the **C** letter defines the capacitor. Select **Open circuit** to define an open circuit ($R=0$, $L=0$, $C=\infty$). Only existing elements are displayed in the block icon.

Resistance R (Ohms) — Branch resistances

1 (default) | scalar

Branch resistances, in ohms (Ω).

Inductance L (H) — Branch inductances

1e-3 (default) | scalar

Branch inductances, in henries (H).

Capacitance C (F) — Branch capacitances

1e-6 (default) | scalar

Branch capacitances, in farads (F).

Measurements — Block measurements

None (default) | Branch voltages | Branch currents | Branch voltages and currents

Select **Branch voltages** to measure the three voltages across the Three-Phase Series RLC Branch block terminals.

Select **Branch currents** to measure the three currents flowing through the Three-Phase Series RLC Branch block.

Select **Branch voltages and currents** to measure the three voltages and the three currents of the Three-Phase Series RLC Branch block.

Place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurements** list box of the Multimeter block, the measurements are identified by a label followed by the block name.

Measurement	Label
Branch voltages	Ua :
Branch currents	Ia :

Version History

Introduced before R2006a

Extended Capabilities

C/C++ Code Generation

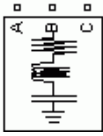
Generate C and C++ code using Simulink® Coder™.

See Also

Multimeter | Three-Phase Parallel RLC Branch | Three-Phase Parallel RLC Load | Three-Phase Series RLC Load

Three-Phase Series RLC Load

Implement three-phase series RLC load with selectable connection



Libraries:

Simscape / Electrical / Specialized Power Systems / Passives

Description

The Three-Phase Series RLC Load block implements a three-phase balanced load as a series combination of RLC elements. At the specified frequency, the load exhibits a constant impedance. The active and reactive powers absorbed by the load are proportional to the square of the applied voltage.

Only elements associated with nonzero powers are displayed in the block icon.

Ports

Conserving

A — Phase A

specialized electrical

Specialized electrical conserving port associated with phase A.

B — Phase B

specialized electrical

Specialized electrical conserving port associated with phase B.

C — Phase C

specialized electrical

Specialized electrical conserving port associated with phase C.

N — Neutral

specialized electrical

Specialized electrical conserving port associated with the neutral.

Dependencies

To enable this port, set **Configuration** to Y (neutral).

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Parameters

Configuration — Connection of three phases

Y(grounded) (default) | Y(floating) | Y(neutral) | Delta

The connection of the three phases. Select one of the following four connections:

Y(grounded)	Neutral is grounded.
Y(floating)	Neutral is not accessible.
Y(neutral)	Neutral is made accessible through a fourth connector.
Delta	Three phases connected in delta

The block icon is updated according to the load connection.

Nominal phase-to-phase voltage Vn — Nominal phase-to-phase voltage of load

1000 (default)

Nominal phase-to-phase voltage of the load, in volts RMS (Vrms).

Dependencies

To enable this parameter, clear the **Specify PQ powers for each phase** parameter.

Nominal phase-to-neutral voltages [Va Vb Vc] (Vrms) — Nominal phase-to-neutral voltages of load

[1e3 1e3 1e3] (default)

Nominal phase-to-neutral voltages of the load, in volts RMS (Vrms).

Dependencies

To enable this parameter, select the **Specify PQ powers for each phase** parameter.

Nominal frequency fn (Hz) — Nominal frequency

60 (default)

The nominal frequency, in hertz (Hz). Default is 60.

Specify PQ powers for each phase — Option to specify PQ powers for each phase

off (default) | on

Option to specify active powers, inductive reactive powers, and capacitive reactive powers for each phase of the load.

Active power P (W) — Three-phase active power of load

10e3 (default)

Three-phase active power of the load, in watts.

Dependencies

To enable this parameter, clear the **Specify PQ powers for each phase** parameter.

Inductive reactive power QL (positive var) — Three-phase inductive reactive power QL

100 (default) | nonnegative scalar

Three-phase inductive reactive power QL, in vars.

Dependencies

To enable this parameter, clear the **Specify PQ powers for each phase** parameter.

Capacitive reactive power QC (negative var) — Three-phase capacitive reactive power QC
100 (default) | nonnegative scalar

Three-phase capacitive reactive power QC, in vars.

Dependencies

To enable this parameter, clear the **Specify PQ powers for each phase** parameter.

Active powers [Pa Pb Pc] (W) — Three-phase active power of load
[10e3 9e3 11e3] (default) | vector of three nonnegative elements

Three-phase active power of the load, in watts. You can specify different powers for each phase, including 0.

Dependencies

To enable this parameter, select the **Specify PQ powers for each phase** parameter.

Inductive reactive powers [QLa QLb QLc] (positive var) — Three-phase inductive reactive power QL
[100 90 110] (default) | vector of three nonnegative elements

Three-phase inductive reactive power QL, in vars. You can specify different powers for each phase, including 0.

Dependencies

To enable this parameter, select the **Specify PQ powers for each phase** parameter.

Capacitive reactive powers [QCa QCb QCc] (negative var) — Three-phase capacitive reactive power QC
[100 90 110] (default) | vector of three nonnegative elements

Three-phase capacitive reactive power QC, in vars. You can specify different powers for each phase, including 0.

Dependencies

To enable this parameter, select the **Specify PQ powers for each phase** parameter.

Measurements — Type of measurements
None (default) | Branch voltages | Branch currents | Branch voltages and currents

Select **Branch voltages** to measure the three voltages across each phase of the Three-Phase Series RLC Load block terminals. For a Y connection, these voltages are the phase-to-ground or phase-to-neutral voltages. For a delta connection, these voltages are the phase-to-phase voltages.

Select **Branch currents** to measure the three total currents (sum of R, L, C currents) flowing through each phase of the Three-Phase Series RLC Load block. For a delta connection, these currents are the currents flowing in each branch of the delta.

Select **Branch voltages** and **currents** to measure the three voltages and the three currents of the Three-Phase Series RLC Load block.

Place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurements** list box of the Multimeter block, the measurements are identified by a label followed by the block name.

Measurement		Label
Branch voltages	Y(grounded)	Uag:
	Y(floating)	Uan:
	Y(neutral)	Uan:
	Delta	Uab:
Branch currents	Y(grounded)	Iag:
	Y(floating)	Ian:
	Y(neutral)	Ian:
	Delta	Iab:

Load Flow

Load type — Option to specify load type

constant Z (default) | constant PQ | constant I

Specify the load type.

If you select **constant Z**, the load impedance is determined from the nominal phase-to-phase voltage V_n , active power P , and reactive power (QL-QC) specified on the **Parameters** tab of the block dialog box. During the load flow solution, the impedance is kept constant. The effective P and Q are therefore varying proportionally to the square of the bus voltage computed by the Load Flow tool.

If you select **constant PQ**, the active power P and reactive power Q are kept constant and equal to the values specified on the **Parameters** tab of the block dialog box. When you apply the load flow solution to the model (by clicking the **Apply** button of the Load Flow tool), the **Nominal phase-to-phase voltage V_n** parameter on the **Parameters** tab is automatically adjusted to the phase-to-phase bus voltage computed by the Load Flow tool.

If you select **constant I**, the load current is kept constant at its nominal value determined from base voltage specified in the dialog box of the associated Load Flow Bus block and from active power P and reactive power Q_L and Q_C specified on the **Parameters** tab of the block dialog box. The effective P and Q are therefore varying proportionally to the bus voltage computed by the Load Flow tool. When you apply the load flow solution to the model (by clicking the **Apply** button of the Load Flow tool), the p.u. value of the nominal voltage V_n on the **Parameters** tab is automatically adjusted to the square root of the load voltage (in pu) computed by the Load Flow tool.

Version History

Introduced before R2006a

Extended Capabilities

C/C++ Code Generation

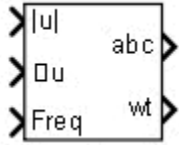
Generate C and C++ code using Simulink® Coder™.

See Also

Multimeter | Three-Phase Dynamic Load | Three-Phase Parallel RLC Branch | Three-Phase Parallel RLC Load | Three-Phase Series RLC Branch

Three-Phase Sine Generator

Generate three-phase balanced signal, amplitude, phase, and frequency controlled by block inputs



Library

Simscape / Electrical / Specialized Power Systems / Sources / Signal Generator Sources

Description

Use the Three-Phase Sine Generator block to generate a three-phase balanced sinusoidal signal (positive-sequence). This vectorized signal usually represents a set of three-phase voltages or currents. Three inputs let you control the amplitude, phase, and frequency of the three-phase output signal.

Parameters

Sample time

Specify the sample time of the block, in seconds. Set to 0 to implement a continuous block. Default is 0.

Inputs and Outputs

mag u

Specify the amplitude of the three-phase output signal.

angle u

Specify the angle, in degrees, of the three-phase output signal.

Freq

Specify the frequency, in hertz, of the three-phase output signal.

abc

Returns the three-phase signal.

wt

Angle (rad) varying between 0 and 2π , synchronized on zero crossings of the fundamental (positive-sequence) of phase A.

Characteristics

Sample Time	- Specified in the Sample Time parameter - Continuous if Sample Time = 0
Scalar Expansion	No
Dimensionalized	No

Examples

The `power_ThreePhaseSineGenerator` model shows the use of the Three-Phase Sine Generator block.

The initial three-phase signal is set to 1 pu, 60 Hz, 90 deg. The following variations are applied on amplitude and frequency while phase stays constant:

- From 0.1 s to 0.4 s, frequency is ramped from 60 Hz to 63 Hz.
- From 0.2 s to 0.4 s, amplitude is sinusoidally modulated with an amplitude of 0.4 pu and a frequency of 5 Hz.

Measured amplitude and frequency are compared with specified values.

The model sample time is parameterized with variable T_s (default value $T_s = 50e-6$). To simulate the continuous Three-Phase Programmable Generator block, specify $T_s = 0$ in the command window.

See Also

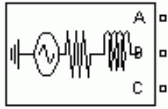
Three-Phase Programmable Voltage Source, Three-Phase Programmable Generator

Version History

Introduced in R2013a

Three-Phase Source

Implement three-phase source with internal R-L impedance



Libraries:

Simscape / Electrical / Specialized Power Systems / Sources

Description

The Three-Phase Source block implements a balanced three-phase voltage source with an internal R-L impedance. The block connects the three voltage sources in Y with a neutral connection that you can internally ground or make it accessible. You can specify the source internal resistance and inductance either directly by entering R and L values or indirectly by specifying the source inductive short-circuit level and X/R ratio.

Ports

Conserving

A — A-phase

specialized electrical

Specialized electrical conserving port associated with the A-phase of the block.

B — B-phase

specialized electrical

Specialized electrical conserving port associated with the B-phase of the block.

C — C-phase

specialized electrical

Specialized electrical conserving port associated with the C-phase of the block.

N — Neutral connection

specialized electrical

Specialized electrical conserving port associated with the neutral connection of the block.

Dependencies

To enable this port, set **Configuration** to Yn.

Parameters

Parameters

Configuration — Internal connection of three voltage sources

Yg (default) | Yn | Y

The internal connection of the three internal voltage sources. The block icon changes according to the source connection:

Y	The block connects the three voltage sources in Y to an internal floating neutral.
Yn	The block connects the three voltage sources in Y to a neutral connection which is accessible through a fourth terminal.
Yg	Default. The block connects the three voltage sources in Y to an internally grounded neutral.

Source

Specify internal voltages for each phase — Option for internal voltages for each phase
off (default) | on

Option to specify the internal voltage for each phase.

Phase-to-phase voltage (Vrms) — Internal phase-to-phase voltage
25e3 (default)

Internal phase-to-phase voltage, in volts RMS.

Dependencies

To enable this parameter, clear the **Specify internal voltages for each phase** parameter.

Phase angle of phase A (degrees) — Phase angle of phase A
0 (default)

Phase angle of the internal voltage that phase A generates, in degrees. The three voltages are generated in a positive sequence. For this reason, phase B and phase C internal voltages lag phase A by 120 degrees and 240 degrees, respectively.

Dependencies

To enable this parameter, clear the **Specify internal voltages for each phase** parameter.

Line-to-neutral voltages [Va Vb Vc] (Vrms) — Internal line-to-neutral voltages
[25e3 25e3 25e3]/sqrt(3) (default)

Internal line-to-neutral voltage of phase A, B, and C, in volts RMS.

Dependencies

To enable this parameter, select the **Specify internal voltages for each phase** parameter.

Phase angle of line-to-neutral voltages [phia phib phic] (deg) — Phase angle of line-to-neutral voltages
[0 -120 +120] (default)

Phase angles of the internal voltages generated by phase A, B, and C, in degrees.

Dependencies

To enable this parameter, select the **Specify internal voltages for each phase** parameter.

Frequency (Hz) — Source frequency
60 (default)

Source frequency, in hertz (Hz).

Impedance

Internal — Option to model internal RL inductance
on (default) | off

Whether to model an internal RL inductance.

Specify short-circuit level parameters — Option to specify internal impedance
off (default) | on

Whether to specify internal impedance using the inductive short-circuit level and X/R ratio.

Dependencies

To enable this parameter, select the **Internal** parameter.

Source resistance (Ohms) — Source internal resistance
0.8929 (default)

Source internal resistance, in ohms (Ω).

Dependencies

To enable this parameter, clear the **Internal** and **Specify short-circuit level parameters** parameters.

Source inductance (H) — Source internal inductance
16.58e-3 (default)

Source internal inductance, in henries (H).

Dependencies

To enable this parameter, clear the **Internal** and **Specify short-circuit level parameters** parameters.

3-phase short-circuit level at base voltage(VA) — Three-phase inductive short-circuit power
100e6 (default)

Three-phase inductive short-circuit power, in volts-amperes (VA), at specified base voltage. The block uses this value to compute the internal inductance L.

This equation computes the internal inductance L (in H) from the inductive three-phase short-circuit power Psc (in VA), base voltage Vbase (in Vrms phase-to-phase), and source frequency f (in Hz):

$$L = \frac{V_{base}^2}{P_{sc}} \cdot \frac{1}{2\pi f}.$$

Dependencies

To enable this parameter, select the **Internal** and **Specify short-circuit level parameters** parameters.

Base voltage (Vrms ph-ph) — Phase-to-phase base voltage
25e3 (default)

Phase-to-phase base voltage, in volts RMS. Use this parameter to specify the three-phase short-circuit level. Default is 25e3. The base voltage is usually the nominal source voltage.

Dependencies

To enable this parameter, select the **Internal** and **Specify short-circuit level parameters** parameters.

X/R ratio — X/R ratio at nominal source frequency
7 (default)

X/R ratio at nominal source frequency or quality factor of the internal source impedance.

This equation computes the internal resistance R (in ohms) from the source reactance X (in ohms) at specified frequency, and X/R ratio:

$$R = \frac{X}{(X/R)} = \frac{2\pi fL}{X/R}.$$

Dependencies

To enable this parameter, select the **Internal** and **Specify short-circuit level parameters** parameters.

Load Flow

The Load Flow tool of the powergui block uses the parameters in this section. These load flow parameters are used for model initialization only, they have no impact on the block model and on the simulation performance.

The configuration of the **Load Flow** tab depends on the option you select for the **Generator type** parameter.

Generator type — Generator type of the voltage source
swing (default) | PV | PQ

Specify the generator type of the voltage source:

- **swing** — Implement a generator controlling magnitude and phase angle of its terminal voltage. The **Swing bus or PV bus voltage** and **Swing bus voltage angle** parameters of the Load Flow Bus block connected to the voltage source terminals specify the reference voltage magnitude and angle.
- **PV** — Implement a generator controlling its output active power P and voltage magnitude V. The **Active power generation P** parameter of the block specifies the value of the output active power P. The **Swing bus or PV bus voltage** parameter of the Load Flow Bus block connected to the voltage source terminals specifies the value of the voltage magnitude V. You can control the minimum and maximum reactive power of this block by using the **Minimum reactive power Qmin** and **Maximum reactive power Qmax** parameters.
- **PQ** — Implement a generator controlling its output active power P and reactive power Q. The **Active power generation P** and **Reactive power generation Q** parameters of the block specify the values of the output active power P and reactive power Q, respectively.

Active power generation P (W) — Desired active power
10e3 (default)

Desired active power that the source generates, in watts.

Dependencies

To enable this parameter, set **Generator type** to PV or PQ and clear the **Specify internal voltages for each phase** parameter.

Reactive power generation Q (var) — Desired reactive power
0 (default)

Desired reactive power that the source generates, in vars.

Dependencies

To enable this parameter, set **Generator type** to PQ and clear the **Specify internal voltages for each phase** parameter.

Active power generation [Pa Pb Pc] (W) — Desired active power that each phase of source generates
[10e3, 10e3, 10e3] (default)

Desired active power that each phase of the source generates, in watts.

Dependencies

To enable this parameter, set **Generator type** to PV or PQ and select the **Specify internal voltages for each phase** parameter.

Reactive power generation [Qa Qb Qc] (var) — Desired reactive power that each phase of source generates
[0, 0, 0] (default)

Desired reactive power that each phase of the source generates, in vars.

Dependencies

To enable this parameter, set **Generator type** to PQ and select the **Specify internal voltages for each phase** parameter.

Minimum reactive power Qmin or [Qamin,Qbmin Qcmin] (var) — Minimum reactive power that source can generate
-inf (default)

Minimum reactive power that the source can generate while keeping the terminal voltage at its reference value. The **Swing bus or PV bus voltage** parameter of the Load Flow Bus block connected to the source terminals specifies this reference voltage. The default value is -inf, which means that there is no lower limit on the reactive power output.

Dependencies

To enable this parameter, set **Generator type** to PV.

Maximum reactive power Qmax or [Qamax,Qbmax Qcmax] (var) — Maximum reactive power that source can generate
inf (default)

Maximum reactive power that the source can generate while keeping the terminal voltage at its reference value. The **Swing bus or PV bus voltage** parameter of the Load Flow Bus block connected to the source terminals specifies this reference voltage. The default value is `inf`, which means that there is no upper limit on the reactive power output.

Dependencies

To enable this parameter, set **Generator type** to PV.

Version History

Introduced before R2006a

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

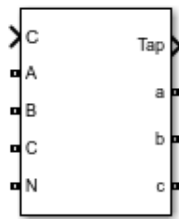
Three-Phase Programmable Voltage Source

Topics

“Series-Compensated Transmission System”

Three-Phase Tap-Changing Transformer (Two-Windings)

Three-phase tap-changing transformer



Libraries:

Simscape / Electrical / Specialized Power Systems / Power Grid Elements

Description

The Three-Phase Tap-Changing Transformer (Two Windings) block models a three-phase, tap-changing transformer with two windings. The tap changer is on winding 1. Winding 1 is connected in wye. Winding 2 can be connected in wye or delta.

The transformer voltage ratio is determined by the position of the tap and by the method used to control it. When **Mode** is set to **Tap control**, the position of the tap is controlled by the input signal. When **Mode** is set to **Voltage regulation**, the position of the tap is controlled by an internal voltage regulator that compares the input signal to the reference voltage.

When you use a Three-Phase Tap-Changing Transformer block, set the powergui block **Simulation type** to **Discrete** and select **Automatically handle Discrete solver**. The robust discrete solver is used to discretize the electrical model. Simulink returns an error if the robust discrete solver is not used.

Ports

Conserving

A - Winding 1 phase A terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the phase A terminal of winding 1.

B - Winding 1 phase B terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the phase B terminal of winding 1.

C - Winding 1 phase C terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the phase C terminal of winding 1.

N - Winding 1 neutral terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the neutral terminal of winding 1.

a - Winding 2 phase A terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the phase A terminal of winding 2.

b - Winding 2 phase B terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the phase B terminal of winding 2.

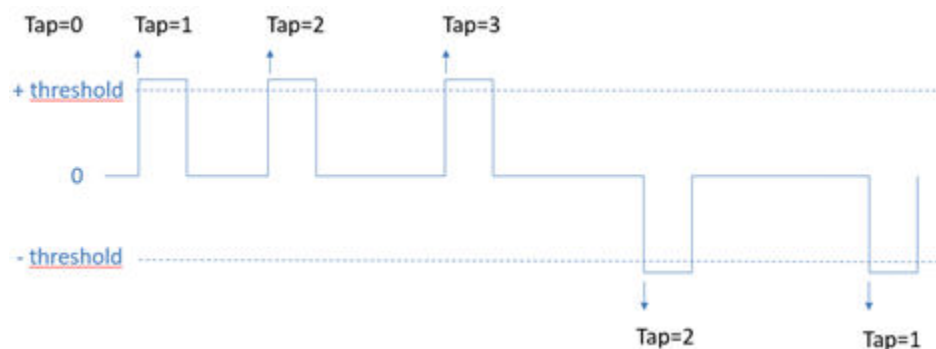
c - Winding 2 phase C terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the phase C terminal of winding 2.

Simulink

ctrl - Control input — Signal used to control tap position
scalar

Signal used to control the tap position of the transformer. The tap position of the transformer increases when the **ctrl** input signal becomes greater than the specified **Threshold** parameter value. The tap position decreases when the **ctrl** input signal becomes less than the negative of the **Threshold** parameter value.



The new tap position becomes effective after the time specified by the **Tap selection time** parameter.

Dependencies

To enable this port, set **Mode** to Tap Control.

Vm (pu) - Measured voltage signal — Signal used to control tap position
scalar

Signal used to control the tap position of the transformer with an internal voltage regulator. Use the magnitude of voltage, in pu, measured at the secondary winding or at a remote bus.

The tap position of the transformer increases when the difference between the **Vm (pu)** input signal and the **Reference voltage Vref** parameter value becomes greater than the value of the **Dead band** parameter divided by 2, for a time duration greater than the **Delay** parameter value.

The tap position decreases when the difference between the **Vm (pu)** input signal and the **Reference voltage Vref (pu)** parameter value becomes lower than the value of the negative of the **Dead band** parameter divided by 2, for a time duration greater than the **Delay** parameter value.

Dependencies

To enable this port, set **Mode** to Voltage regulation.

Tap - Tap position value — Output signal
integer

Output signal that gives the current tap position.

Parameters

Transformer

[Pnom(VA), Fnom(Hz)] - Transformer nominal parameters — Nominal power rating
[47e6 , 60] (default) | two-element vector

Nominal power rating, in volt-amperes (VA), and nominal frequency, in hertz (Hz), of the transformer.

Winding 1 connection (ABC) — Winding connection
Yn (default)

Winding connection for winding 1.

Winding 1 [V1 Ph-Ph(Vrms), R1(pu), L1(pu)] - Winding 1 parameters — Phase-to-phase nominal voltage
[120e3, 0.003, 0.09] (default) | three-element vector

Phase-to-phase nominal voltage, in volts RMS, and resistance and leakage inductance, in pu, for winding 1.

Winding 2 connection (abc) - Winding 2 connection — Winding connection
Delta (D1) (default) | Y | Yg | Delta (D11)

Winding connection for winding 2.

Winding 2 parameters [V2 Ph-Ph(Vrms), R2(pu), L2(pu)] - Winding 2 parameters — Phase-to-phase nominal voltage
[25e3 , 0.003 , 0.09] (default) | three-element vector

Phase-to-phase nominal voltage, in volts RMS, and resistance and leakage inductance, in pu, for winding 2.

Core type - Transformer core type — Core type
Three single-phase transformers (default) | Three-limb core (core-type)

Select **Three single-phase transformers** to implement a three-phase transformer using three single-phase transformer models. This core type is used in very large power transformers found in utility grids.

Select **Three-limb core (core type)** to implement a three-limb core three-phase transformer. In most applications, three-phase transformers use a three-limb core (core-type transformer). Selecting this core type will produce accurate results during an asymmetrical fault for both linear and nonlinear models (including saturation). During asymmetrical voltage conditions, the zero-sequence flux of a core-type transformer returns outside the core, through an air gap, structural steel, and tank. Consequently, the natural zero-sequence inductance L_0 (without delta winding) of a core-type transformer is usually very low (typically $0.3 \text{ pu} < L_0 < 2 \text{ pu}$) compared with a three-phase transformer that uses three single-phase units ($L_0 > 100 \text{ pu}$). This low L_0 value affects voltages, currents, and flux unbalances during linear and saturated operation.

Inductance L_0 of zero-sequence flux path return (pu) - Three-limb core inductance — Inductance L_0

0.5 (default) | scalar number

Inductance L_0 , in pu, of the three-limb core transformer.

Dependencies

To enable this parameter, set **Core Type** to **Three-limb core (core type)**.

R_m (pu) - Magnetisation resistance — Magnetization resistance

500 (default) | scalar number

Magnetization resistance R_m , in pu.

L_m (pu) - Magnetization inductance — Magnetization inductance

500 (default) | scalar number

Magnetization inductance L_m , in pu, for a nonsaturable core.

Dependencies

To enable this parameter, clear **Simulate saturation**.

Simulate saturation - Activate transformer saturation — Implement saturable three-phase transformer

off (default) | on

Select this parameter to implement a saturable three-phase transformer.

[i_1 , ϕ_{i1} ; i_2 , ϕ_{i2} ; ...] (pu) - Saturation characteristic — Saturation characteristic

[0,0 ; 0.0024,1.2 ; 1.0,1.52] (default) | matrix

Saturation characteristic for the saturable core. Specify a series of current and flux pairs, in pu, starting with the pair (0, 0).

Dependencies

To enable this parameter, select **Simulate saturation**.

Specify initial fluxes - Activate initial flux specification — Specify initial fluxes

off (default) | on

If selected, the initial fluxes are defined by the **[phi0A, phi0B, phi0C] (pu)** parameter.

When the **Specify initial fluxes** parameter is cleared, the Simscape Electrical Specialized Power Systems software automatically computes the initial fluxes to start the simulation in steady state. The computed values are saved in the **Initial Fluxes** parameter and overwrite any previous values.

Dependencies

To enable this parameter, select **Simulate saturation**.

[phi0A , phi0B , phi0C] (pu) - Initial fluxes — Initial fluxes
[-1.0 0.5 0.5] (default) | vector of 3 values

Initial fluxes for each phase of the transformer.

When the **Specify initial fluxes** parameter is not selected upon simulation, the Simscape Electrical Specialized Power Systems software automatically computes the initial fluxes to start the simulation in steady state. The computed values are saved in the **Initial Fluxes** parameter and overwrite any previous values.

Dependencies

To enable this parameter, select **Simulate saturation** and **Specify initial fluxes**.

Measurements - Measurements — Measurements

None (default) | Winding voltages | Winding currents | Fluxes and excitation currents (Imag + IRm) | Fluxes and magnetization currents (Imag) | All measurements (V I Fluxes)

Select **Winding voltages** to measure the voltage across the winding terminals.

Select **Winding currents** to measure the current flowing through the windings.

Select **Fluxes and excitation currents (Imag + IRm)** to measure the flux linkage, in volt seconds (Vs), and the total excitation current including iron losses modeled by Rm.

Select **Fluxes and magnetization currents (Imag)** to measure the flux linkage, in volt seconds (Vs), and the magnetization current, in amperes (A), not including iron losses modeled by Rm.

Select **All measurements (V I Fluxes)** to measure the winding voltages, currents, magnetization currents, and the flux linkages.

Place a Multimeter block in your model to display the selected measurements during simulation. In the **Available Measurements** parameter of the Multimeter block, the measurements are identified by a label followed by the block name.

Tap-Changer

Minimum tap position - Minimum tap position of transformer — Minimum tap position
-20 (default) | integer

Minimum tap position of the transformer. This value is typically between -20 and +20. At the minimum tap position, the transformer voltage ratio is:

$$U_2/U_1 = 1 / (1 + \text{Minimum tap position} * \text{Voltage step per tap (pu)}).$$

For example, for a minimum tap position of -20 and a voltage step of 0.015, the voltage ratio is:

$$U_2/U_1 = 1/(1 + -20 \cdot 0.015) = 1.428 \text{ pu.}$$

Maximum tap position - Maximum tap position of transformer — Maximum tap position
20 (default) | integer

Maximum tap position of the transformer. This value is typically between -20 and +20. At the maximum tap position, the transformer voltage ratio is:

$$U_2/U_1 = 1 / (1 + \text{Maximum tap position} * \text{Voltage step per tap (pu)}).$$

For example, for a maximum tap position of +20 and a voltage step of 0.015, the voltage ratio is:

$$U_2/U_1 = 1/(1 + 20 \cdot 0.015) = 0.769 \text{ pu.}$$

Initial tap position - Initial tap position of transformer — Initial tap position
0 (default) | integer

Initial position of the tap when the simulation starts. The initial position must be a value between the minimum tap position and the maximum tap position. When the tap position is 0, the transformer voltage ratio is equal to 1 pu (nominal value).

Voltage step per tap (pu) - Voltage step between two taps — Voltage step
0.015 (default) | positive scalar

Voltage step between two taps of the transformer, in pu. This value is typically in the order of 0.01 to 0.05 pu.

The transformer voltage ratio is given by:

$$U_2/U_1 = 1 / (1 + \text{Tap position} * \text{Voltage step per tap (pu)}).$$

For example, a voltage step per tap value of 0.015, and a tap position of -20, would produce a voltage ratio of:

$$U_2/U_1 = 1/(1 + -20 \cdot 0.015) = 1.428 \text{ pu.}$$

The value of the **Voltage step per tap (pu)** parameter must be chosen in accordance with the value specified for the **Minimum tap position** parameter. Follow this constraint according to the above equation:

$$\text{Minimum tap position} * \text{Voltage step per tap} > -1$$

For example, for a **Voltage step per tap (pu)** parameter of 0.05 pu, the minimum value you can specify for the **Minimum tap position** parameter is: -19. A value of -20 gives an infinite voltage ratio and a value lower than -20 yields to a negative voltage ratio, which is not relevant.

Tap selection time (s) - Time delay — Time for tap to move
0.5 (default)

Time for the tap to move from one position to the next. This value is typically between 3 and 10 seconds.

Mode - Tap changer control mode — Control tap position
Tap control (default) | Voltage regulation

Select **Tap control** to control the tap position of the transformer with the **ctrl** input signal. Select **Voltage regulation** to control the tap position of the transformer using an internal voltage regulator monitoring the **Vm (pu)** input signal.

Threshold - Control value threshold — Control value at which tap changes position
0.5 (default)

Control value at which the tap changes position. To increase the tap position, apply a control signal **ctrl** greater than this value. To lower the tap position, apply a control signal **ctrl** smaller than the negative of this value.

Dependencies

To enable this parameter, set **Mode** to **Tap control**.

Reference voltage Vref (pu) - Voltage regulator reference voltage — Reference voltage
1.0 (default) | positive scalar

Reference voltage of the voltage regulator, in pu.

Dependencies

To enable this parameter, set **Mode** to **Voltage regulation**.

Dead band (pu) - Voltage regulator dead band — Dead band of voltage regulator
0.03 (default)

Dead band of the voltage regulator, in pu.

Reference voltage of the voltage regulator, in pu.

Dependencies

To enable this parameter, set **Mode** to **Voltage regulation**.

Delay (sec) - Voltage regulator delay — Delay of voltage regulator
0.4 (default)

Delay of the voltage regulator, in sec.

Dependencies

To enable this parameter, set **Mode** to **Voltage regulation**.

Version History

Introduced in R2021b

Extended Capabilities

C/C++ Code Generation

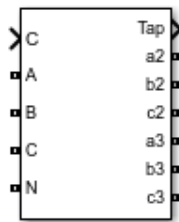
Generate C and C++ code using Simulink® Coder™.

See Also

Three-Phase OLTC Regulating Transformer (Phasor Type) | Three-Phase OLTC Phase Shifting Transformer Delta-Hexagonal (Phasor Type)

Three-Phase Tap-Changing Transformer (Three-Windings)

Three-phase tap-changing transformer



Libraries:

Simscape / Electrical / Specialized Power Systems / Power Grid Elements

Description

The Three-Phase Tap-Changing Transformer (Three Windings) block models a three-phase, tap-changing transformer with three windings. The tap changer is on winding 1. Winding 1 is connected in wye. Windings 2 and 3 can be connected in wye or delta.

The transformer voltage ratio is determined by the position of the tap and by the method used to control it. When **Mode** is set to **Tap control**, the position of the tap is controlled by the input signal. When **Mode** is set to **Voltage regulation**, the position of the tap is controlled by an internal voltage regulator that compares the input signal to the reference voltage.

When you use a Three-Phase Tap-Changing Transformer block, set the powergui block **Simulation type** to **Discrete** and select **Automatically handle Discrete solver**. The robust discrete solver is used to discretize the electrical model. Simulink returns an error if the robust discrete solver is not used.

Ports

Conserving

A - Winding 1 phase A terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the phase A terminal of winding 1.

B - Winding 1 phase B terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the phase B terminal of winding 1.

C - Winding 1 phase C terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the phase C terminal of winding 1.

N - Winding 1 neutral terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the neutral terminal of winding 1.

a2 - Winding 2 phase A terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the phase A terminal of winding 2.

b2 - Winding 2 phase B terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the phase B terminal of winding 2.

c2 - Winding 2 phase C terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the phase C terminal of winding 2.

a3 - Winding 3 phase A terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the phase A terminal of winding 3.

b3 - Winding 3 phase B terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the phase B terminal of winding 3.

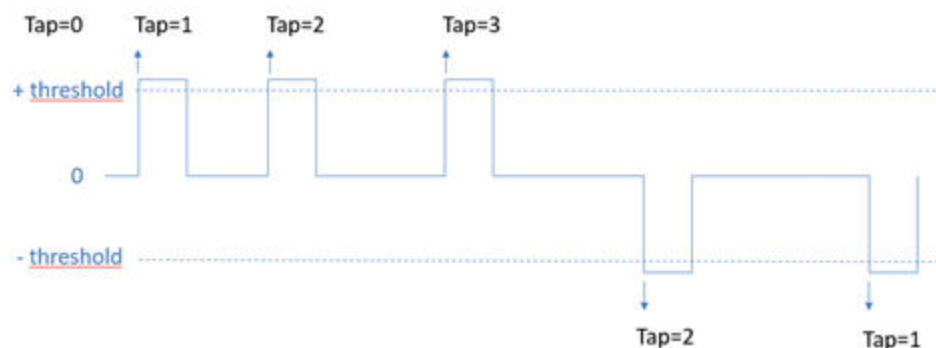
c3 - Winding 3 phase C terminal — Electrical conserving port
specialized electrical

Electrical conserving port associated with the phase C terminal of winding 3.

Simulink

ctrl - Control input — Signal used to control tap position
scalar

Signal used to control the tap position of the transformer. The tap position of the transformer increases when the **ctrl** input signal becomes greater than the specified **Threshold** parameter value. The tap position decreases when the **ctrl** input signal becomes less than the negative of the **Threshold** parameter value.



The new tap position becomes effective after the time specified by the **Tap selection time** parameter.

Dependencies

To enable this port, set **Mode** to Tap Control.

Vm (pu) - Measured voltage signal — Signal used to control tap position
scalar

Signal used to control the tap position of the transformer with an internal voltage regulator. Use the magnitude of voltage, in pu, measured at one of the two secondary windings (winding 2 or winding 3) or at a remote bus.

The tap position of the transformer increases when the difference between the **Vm (pu)** input signal and the **Reference voltage Vref** parameter value becomes greater than the value of the **Dead band** parameter divided by 2, for a time duration greater than the **Delay** parameter value.

The tap position decreases when the difference between the **Vm (pu)** input signal and the **Reference voltage Vref (pu)** parameter value becomes lower than the value of the negative of the **Dead band** parameter divided by 2, for a time duration greater than the **Delay** parameter value.

Dependencies

To enable this port, set **Mode** to Voltage regulation.

Tap - Tap position value — Output signal
integer

Output signal that gives the current tap position.

Parameters

Transformer

[Pnom(VA), Fnom(Hz)] - Transformer nominal parameters — Nominal power rating
[47e6 , 60] (default) | two-element vector

Nominal power rating, in volt-amperes (VA), and nominal frequency, in hertz (Hz), of the transformer.

Winding 1 connection (ABC) — Winding connection
Yn (default)

Winding connection for winding 1.

Winding 1 [V1 Ph-Ph(Vrms), R1(pu), L1(pu)] - Winding 1 parameters — Phase-to-phase nominal voltage
[120e3, 0.003, 0.09] (default) | three-element vector

Phase-to-phase nominal voltage, in volts RMS, and resistance and leakage inductance, in pu, for winding 1.

Winding 2 connection (abc2) - Winding 2 connection — Winding connection
Delta (D1) (default) | Y | Yg | Delta (D11)

Winding connection for winding 2.

Winding 2 parameters [V2 Ph-Ph(Vrms), R2(pu), L2(pu)] - Winding 2 parameters — Phase-to-phase nominal voltage

[25e3 , 0.003 , 0.09] (default) | three-element vector

Phase-to-phase nominal voltage, in volts RMS, and resistance and leakage inductance, in pu, for winding 2.

Winding 3 connection (abc3) - Winding 3 connection — Winding connection

Delta (D1) (default) | Y | Yg | Delta (D11)

Winding connection for winding 3.

Winding 3 [V2 Ph-Ph(Vrms) , R3(pu) , L3(pu)] - Winding 3 parameters — Phase-to-phase nominal voltage

[25e3 , 0.003 , 0.09] (default) | three-element vector

Phase-to-phase nominal voltage, in volts RMS, and resistance and leakage inductance, in pu, for winding 3.

Core type - Transformer core type — Core type

Three single-phase transformers (default) | Three-limb core (core-type)

Select Three single-phase transformers to implement a three-phase transformer using three single-phase transformer models. This core type is used in very large power transformers found in utility grids.

Select Three-limb core (core type) to implement a three-limb core three-phase transformer. In most applications, three-phase transformers use a three-limb core (core-type transformer). Selecting this core type will produce accurate results during an asymmetrical fault for both linear and nonlinear models (including saturation). During asymmetrical voltage conditions, the zero-sequence flux of a core-type transformer returns outside the core, through an air gap, structural steel, and tank. Consequently, the natural zero-sequence inductance L_0 (without delta winding) of a core-type transformer is usually very low (typically $0.3 \text{ pu} < L_0 < 2 \text{ pu}$) compared with a three-phase transformer that uses three single-phase units ($L_0 > 100 \text{ pu}$). This low L_0 value affects voltages, currents, and flux unbalances during linear and saturated operation.

Inductance L0 of zero-sequence flux path return (pu) - Three-limb core inductance —Inductance L_0

0.5 (default) | scalar number

Inductance L_0 , in pu, of the three-limb core transformer.

Dependencies

To enable this parameter, set **Core Type** to Three-limb core (core type).

Rm (pu) - Magnetisation resistance — Magnetization resistance

500 (default) | scalar number

Magnetization resistance R_m , in pu.

Lm (pu) - Magnetization inductance — Magnetization inductance

500 (default) | scalar number

Magnetization inductance L_m , in pu, for a nonsaturable core.

Dependencies

To enable this parameter, clear **Simulate saturation**.

Simulate saturation - Activate transformer saturation — Implement saturable three-phase transformer

off (default) | on

Select this parameter to implement a saturable three-phase transformer.

[i1 , phi1 ; i2 , phi2 ; ...] (pu) - Saturation characteristic — Saturation characteristic

[0,0 ; 0.0024,1.2 ; 1.0,1.52] (default) | matrix

Saturation characteristic for the saturable core. Specify a series of current and flux pairs, in pu, starting with the pair (0, 0).

Dependencies

To enable this parameter, select **Simulate saturation**.

Specify initial fluxes - Activate initial flux specification — Specify initial fluxes

off (default) | on

If selected, the initial fluxes are defined by the **[phi0A, phi0B, phi0C] (pu)** parameter.

When the **Specify initial fluxes** parameter is cleared, the Simscape Electrical Specialized Power Systems software automatically computes the initial fluxes to start the simulation in steady state. The computed values are saved in the **Initial Fluxes** parameter and overwrite any previous values.

Dependencies

To enable this parameter, select **Simulate saturation**.

[phi0A , phi0B , phi0C] (pu) - Initial fluxes — Initial fluxes

[-1.0 0.5 0.5] (default) | vector of 3 values

Initial fluxes for each phase of the transformer.

When the **Specify initial fluxes** parameter is not selected upon simulation, the Simscape Electrical Specialized Power Systems software automatically computes the initial fluxes to start the simulation in steady state. The computed values are saved in the **Initial Fluxes** parameter and overwrite any previous values.

Dependencies

To enable this parameter, select **Simulate saturation** and **Specify initial fluxes**.

Measurements - Measurements — Measurements

None (default) | Winding voltages | Winding currents | Fluxes and excitation currents (Imag + IRm) | Fluxes and magnetization currents (Imag) | All measurements (V I Fluxes)

Select **Winding voltages** to measure the voltage across the winding terminals.

Select **Winding currents** to measure the current flowing through the windings.

Select **Fluxes** and excitation currents (**Imag + IRm**) to measure the flux linkage, in volt seconds (Vs), and the total excitation current including iron losses modeled by Rm.

Select **Fluxes** and magnetization currents (**Imag**) to measure the flux linkage, in volt seconds (Vs), and the magnetization current, in amperes (A), not including iron losses modeled by Rm.

Select **All measurements (V I Fluxes)** to measure the winding voltages, currents, magnetization currents, and the flux linkages.

Place a Multimeter block in your model to display the selected measurements during simulation. In the **Available Measurements** parameter of the Multimeter block, the measurements are identified by a label followed by the block name.

Tap-Changer

Minimum tap position - Minimum tap position of transformer — Minimum tap position
-20 (default) | integer

Minimum tap position of the transformer. This value is typically between -20 and +20. At the minimum tap position, the transformer voltage ratio is:

$$U2/U1 = 1 / (1 + \text{Minimum tap position} * \text{Voltage step per tap (pu)}).$$

For example, for a minimum tap position of -20 and a voltage step of 0.015, the voltage ratio is:

$$U2/U1 = 1 / (1 + -20 * 0.015) = 1.428 \text{ pu.}$$

Maximum tap position - Maximum tap position of transformer — Maximum tap position
20 (default) | integer

Maximum tap position of the transformer. This value is typically between -20 and +20. At the maximum tap position, the transformer voltage ratio is:

$$U2/U1 = 1 / (1 + \text{Maximum tap position} * \text{Voltage step per tap (pu)}).$$

For example, for a maximum tap position of +20 and a voltage step of 0.015, the voltage ratio is:

$$U2/U1 = 1 / (1 + 20 * 0.015) = 0.769 \text{ pu.}$$

Initial tap position - Initial tap position of transformer — Initial tap position
0 (default) | integer

Initial position of the tap when the simulation starts. The initial position must be a value between the minimum tap position and the maximum tap position. When the tap position is 0, the transformer voltage ratio is equal to 1 pu (nominal value).

Voltage step per tap (pu) - Voltage step between two taps — Voltage step
0.015 (default) | positive scalar

Voltage step between two taps of the transformer, in pu. This value is typically in the order of 0.01 to 0.05 pu.

The transformer voltage ratio is given by:

$$U2/U1 = 1 / (1 + \text{Tap position} * \text{Voltage step per tap (pu)}).$$

For example, a voltage step per tap value of 0.015, and a tap position of -20, would produce a voltage ratio of:

$$U2/U1 = 1/(1 + -20 \cdot 0.015) = 1.428 \text{ pu.}$$

The value of the **Voltage step per tap (pu)** parameter must be chosen in accordance with the value specified for the **Minimum tap position** parameter. Follow this constraint according to the above equation:

$$\text{Minimum tap position} \cdot \text{Voltage step per tap} > -1$$

For example, for a **Voltage step per tap (pu)** parameter of 0.05 pu, the minimum value you can specify for the **Minimum tap position** parameter is: -19. A value of -20 gives an infinite voltage ratio and a value lower than -20 yields to a negative voltage ratio, which is not relevant.

Tap selection time (s) - Time delay — Time for tap to move

0.5 (default)

Time for the tap to move from one position to the next. This value is typically between 3 and 10 seconds.

Mode - Tap changer control mode — Control tap position

Tap control (default) | Voltage regulation

Select `Tap control` to control the tap position of the transformer with the **ctrl** input signal. Select `Voltage regulation` to control the tap position of the transformer using an internal voltage regulator monitoring the **Vm (pu)** input signal.

Threshold - Control value threshold — Control value at which tap changes position

0.5 (default)

Control value at which the tap changes position. To increase the tap position, apply a control signal **ctrl** greater than this value. To lower the tap position, apply a control signal **ctrl** smaller than the negative of this value.

Dependencies

To enable this parameter, set **Mode** to `Tap control`.

Reference voltage Vref (pu) - Voltage regulator reference voltage — Reference voltage

1.0 (default) | positive scalar

Reference voltage of the voltage regulator, in pu.

Dependencies

To enable this parameter, set **Mode** to `Voltage regulation`.

Dead band (pu) - Voltage regulator dead band — Dead band of voltage regulator

0.03 (default)

Dead band of the voltage regulator, in pu.

Reference voltage of the voltage regulator, in pu.

Dependencies

To enable this parameter, set **Mode** to Voltage regulation.

Delay (sec) - Voltage regulator delay — Delay of voltage regulator
0.4 (default)

Delay of the voltage regulator, in sec.

Dependencies

To enable this parameter, set **Mode** to Voltage regulation.

Version History

Introduced in R2021b

Extended Capabilities

C/C++ Code Generation

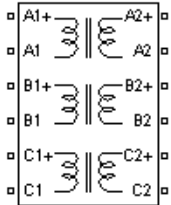
Generate C and C++ code using Simulink® Coder™.

See Also

Three-Phase OLTC Regulating Transformer (Phasor Type) | Three-Phase OLTC Phase Shifting Transformer Delta-Hexagonal (Phasor Type)

Three-Phase Transformer 12 Terminals

Implement three single-phase, two-winding transformers where all terminals are accessible



Library

Simscape / Electrical / Specialized Power Systems / Power Grid Elements

Description

The Three-Phase Transformer 12 Terminals block implements three single-phase, two-winding linear transformers where all the 12 winding connectors are accessible. The block can be used in place of the Three-Phase Transformer (Two Windings) block to implement a three-phase transformer when primary and secondary are not necessarily connected in Y or Delta.

Parameters

[Three-phase rated power Frequency]

The total nominal power of the three phases, in volt-amperes (VA), and the nominal frequency, in hertz (Hz). Default is [10e6 60].

Winding 1: [phase voltage R X]

The nominal voltage of the three primary windings (labeled 1) in volts RMS (V_{rms}), the winding resistances, in pu, and the winding leakage reactances, in pu. Default is [10e3 0.002 0.05].

Winding 2: [phase voltage R X]

The nominal voltage of the three secondary windings (labeled 2) in volts RMS (V_{rms}), the winding resistances, in pu, and the winding leakage reactances, in pu. Default is [25e3 0.002 0.05].

Magnetizing branch: [Rm Xm]

The resistance and reactance simulating the core active and reactive losses, both in pu. For example, to specify 0.2% of active and reactive core losses, at nominal voltage, use $R_m = 500$ pu and $L_m = 500$ pu. L_m can be set to `inf` (no reactive core losses), but R_m must have a finite value. Default is [200 200].

Note Refer to the Linear Transformer for explanations on the per unit system.

See Also

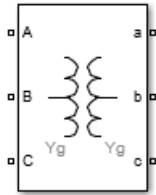
Linear Transformer, Three-Phase Transformer (Two Windings)

Version History

Introduced before R2006a

Three-Phase Transformer (Two Windings)

Implement three-phase transformer with configurable winding connections



Libraries:

Simscape / Electrical / Specialized Power Systems / Power Grid Elements

Description

This block implements a three-phase transformer using three single-phase transformers. For a detailed description of the electrical model of a single-phase transformer, see the Linear Transformer block.

When activated, the saturation characteristic is the same as the one described for the Saturable Transformer block. If the fluxes are not specified, the initial values are automatically adjusted so that the simulation starts in steady state.

The leakage inductance and resistance of each winding are given in pu based on the transformer nominal power P_n and on the nominal voltage of the winding (V_1 or V_2). For a description of per units, refer to the Linear Transformer and to the Saturable Transformer.

The two windings of the transformer can be connected as follows:

- Y
- Y with accessible neutral
- Grounded Y
- Delta (D1), delta lagging Y by 30 degrees
- Delta (D11), delta leading Y by 30 degrees

If you select the Y connection with accessible neutral for winding 1, an input port labeled N is added to the block. If you ask for an accessible neutral on winding 2, an extra output port labeled n is generated.

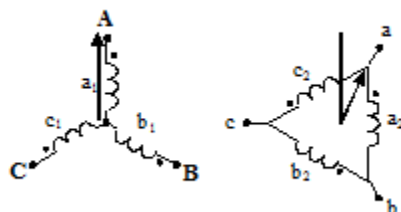
The D1 and D11 notations refer to the clock convention that assumes that the reference Y voltage phasor is at noon (12) on a clock display. D1 and D11 refer respectively to 1:00 p.m. (delta voltages lagging Y voltages by 30 degrees) and 11:00 a.m. (delta voltages leading Y voltages by 30 degrees).

Standard Notation for Winding Connections

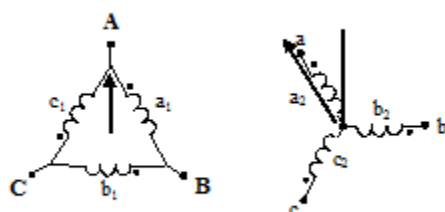
The conventional notation for a two-winding three-phase transformer uses two letters followed by a number. The first letter (Y or D) indicates a high-voltage wye or delta winding connection. The second letter (y or d) indicates a low-voltage wye or delta winding connection. The number, an integer between 0 and 12, indicates the position of the low-voltage positive-sequence voltage phasor on a clock display when the high-voltage positive-sequence voltage phasor is at 12:00.

The following three figures are examples of standard winding connections. The dots indicate polarity marks, and arrows indicate the position of phase A-to-neutral voltage phasors on high-voltage and low-voltage windings. The phasors are assumed to rotate in a counterclockwise direction so that rising numbers indicate increasing phase lag.

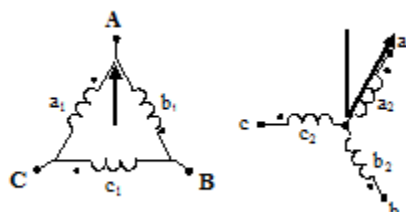
- Yd1: The low-voltage winding (d) is lagging high-voltage winding (Y) by 30 degrees. The **Winding 2 connection** parameter is set to **D1**.



- Dy11: The low-voltage winding (y) is leading high-voltage winding (D) by 30 degrees. The **Winding 1 connection** parameter is set to **D1**.



- Dy1: The low-voltage winding (y) is lagging high-voltage winding (D) by 30 degrees. The **Winding 1 connection** parameter is set to **D11**.

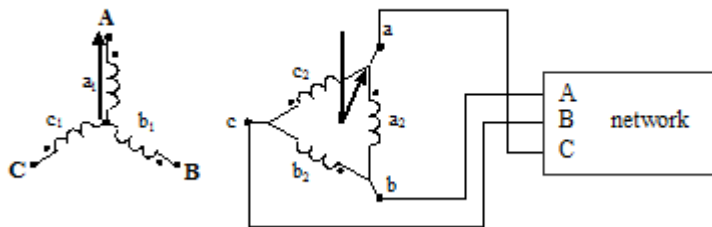


You can represent many other connections with phase shifts between 0 and 360 degrees (by steps of 30 degrees) by combining the +30- or -30-degree phase shift provided by the D1 and D11 block parameter settings and, in some cases, an additional +/-120-degree phase shift obtained by connecting the output terminals of delta winding to the appropriate phases of the network.

The table explains how to set up the Three-Phase Transformer block to obtain common connections.

Clock Position	Phase Shift (degrees)	Connection	Winding 1 Connection	Winding 2 Connection	Terminals of Delta Winding to Connect to Network ABC Phases
0	0	Yy0	Y	Y	—
		Dd0	D1	D1	abc
1	-30	Yd1	Y	D1	abc
		Dy1	D11	Y	abc
2	-60	Dd2	D11	D1	abc
5	-150	Yd5	Y	D1	bca
		Dy5	D11	Y	cab
7	+150	Yd7	Y	D11	cab
		Dy7	D1	Y	bca
10	+60	Dd10	D1	D11	abc
11	+30	Yd11	Y	D11	abc
		Dy11	D1	Y	abc

For example, to obtain the Yd5 connection, set the **Winding 1 connection** parameter to **Y** and the **Winding 2 connection** parameter to **D1**, and connect the network phases to the winding 2 as follows:



For more details on conventional transformer winding notations, see International Standard IEC 60076-1 [1].

Examples

The `power_transfo3ph` circuit uses the Three-Phase Transformer block where the saturable core is simulated. Both windings are connected in a Y grounded configuration. The neutral points of the two windings are internally connected to the ground.

The 500 kV/ 230-kV saturable transformer is energized on the 500-kV system. Remanent fluxes of 0.8 pu, -0.4 pu, and 0.4 pu have been specified respectively for phases A, B, and C. Run the simulation and observe inrush currents due to core saturation. See also the `power_xfonotation` model that shows four types of three-phase Yd and Dy transformer connections.

Ports

Conserving

A — Winding 1 phase A terminal
specialized electrical

Electrical conserving port associated with the phase A terminal of winding 1.

B — Winding 1 phase B terminal
specialized electrical

Electrical conserving port associated with the phase B terminal of winding 1.

C — Winding 1 phase C terminal
specialized electrical

Electrical conserving port associated with the phase C terminal of winding 1.

N — Winding 1 neutral terminal
specialized electrical

Electrical conserving port associated with the neutral terminal of winding 1.

Dependencies

To enable this port, set the **Winding 1 connection (ABC terminals)** parameter to Yn.

a — Winding 2 phase A terminal
specialized electrical

Electrical conserving port associated with the phase A terminal of winding 2.

b — Winding 2 phase B terminal
specialized electrical

Electrical conserving port associated with the phase B terminal of winding 2.

c — Winding 2 phase C terminal
specialized electrical

Electrical conserving port associated with the phase C terminal of winding 2.

n2 — Winding 2 neutral terminal
specialized electrical

Electrical conserving port associated with the neutral terminal of winding 2.

Dependencies

To enable this port, set the **Winding 2 connection (abc terminals)** parameter to Yn.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Configuration

Winding 1 connection (ABC terminals) — Winding connections for winding 1

Yg (default) | Yn | Y | Delta (D1) | Delta (D11)

Winding connections for winding 1. Choices are Y, Yn, Yg (default), Delta (D1), and Delta (D3).

Winding 2 connection (abc terminals) — Winding connections for winding 2

Yg (default) | YnY | Delta (D1) | Delta (D11)

Winding connections for winding 2. Choices are Y, Yn, Yg (default), Delta (D1), and Delta (D11).

Type — Type of transformer

Three single-phase transformers (default) | Three-limb core (core-type) | Five-limb core (shell-type)

Select **Three single-phase transformers (default)** to implement a three-phase transformer using three single-phase transformer models. You can use this core type to represent very large power transformers found in utility grids (hundreds of MW).

Select **Three-limb core (core-type)** to implement a three-limb core three-phase transformer. In most applications, three-phase transformers use a three-limb core (core-type transformer). This type of core produces accurate results during an asymmetrical fault for both linear and nonlinear models (including saturation). During asymmetrical voltage conditions, the zero-sequence flux of a core-type transformer returns outside the core, through an air gap, structural steel, and a tank. Thus, the natural zero-sequence inductance L_0 (without delta winding) of such a core-type transformer is usually very low (typically $0.5 \text{ pu} < L_0 < 2 \text{ pu}$) compared with a three-phase transformer using three single-phase units ($L_0 > 100 \text{ pu}$). This low L_0 value affects voltages, currents, and flux unbalances during linear and saturated operation.

Select **Five-limb core (shell-type)** to implement a five-limb core three-phase transformer. On rare occasions, very large transformers are built with a five-leg core (three phase legs and two external legs). This core configuration, also known as shell type, is chosen mainly to reduce the height of the transformer and make transportation easier. During unbalanced voltage conditions, as opposed to the three-limb transformer, the zero-sequence flux of the five-limb transformer stays inside the steel core and returns through the two external limbs. The natural zero-sequence inductance (without delta) is therefore very high ($L_0 > 100 \text{ pu}$). Except for small current unbalances due to core asymmetry, the behavior of the five-limb shell-type transformer is similar to that of a three-phase transformer built with three single-phase units.

Simulate saturation — Option to simulate saturation

off (default) | on

If selected, implements a saturable three-phase transformer. Default is cleared.

If you want to simulate the transformer in the phasor mode of the Powergui block, you must clear this parameter.

Simulate hysteresis — Option to simulate hysteresis

off (default) | on

Select to model a saturation characteristic including hysteresis instead of a single-valued saturation curve.

If you want to simulate the transformer in the phasor mode of the Powergui block, you must clear this parameter.

Dependencies

To enable this parameter, select the **Simulate saturation** parameter.

Hysteresis Mat file (ex: 'hysteresis.mat') — File containing data for use in hysteresis model
'hysteresis' (default)

Specify a .mat file containing the data for use in the hysteresis model. When you open the **Hysteresis Design Tool** of the Powergui block, the default hysteresis loop and parameters saved in the hysteresis.mat file are displayed. Use the **Load** button of the Hysteresis Design tool to load another .mat file. Use the **Save** button of the Hysteresis Design tool to save your model in a new .mat file.

Dependencies

To enable this parameter, select the **Simulate hysteresis** parameter.

Specify initial fluxes — Option to specify initial fluxes
off (default) | on

If selected, the initial fluxes are defined by the **Initial fluxes** parameter on the **Parameters** tab.

When the **Specify initial fluxes** parameter is not selected upon simulation, Simscape Electrical Specialized Power Systems software automatically computes the initial fluxes to start the simulation in steady state. The computed values are saved in the **Initial Fluxes** parameter and will overwrite any previous values.

Dependencies

To enable this parameter, select the **Simulate saturation** parameter.

Measurements — Type of measurements
None (default) | Winding voltages | Winding currents | Fluxes and excitation currents (Im + IRm) | Fluxes and magnetization currents (Im) | All measurements (V, I, Flux)

Select **Winding voltages** to measure the voltage across the winding terminals.

Select **Winding currents** to measure the current flowing through the windings.

Select **Fluxes and excitation currents (Im + IRm)** to measure the flux linkage, in volt seconds (Vs), and the total excitation current including iron losses modeled by Rm.

Select **Fluxes and magnetization currents (Im)** to measure the flux linkage, in volt seconds (Vs), and the magnetization current, in amperes (A), not including iron losses modeled by Rm.

Select **All measurements (V, I, Flux)** to measure the winding voltages, currents, magnetization currents, and the flux linkages.

Place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurements** list box of the Multimeter block, the measurements are identified by a label followed by the block name.

If the **Winding 1 connection (ABC terminals)** parameter is set to Y, Yn, or Yg, the labels are as follows.

Measurement	Label
Winding 1 voltages	Uan_w1: or Uag_w1:
Winding 1 currents	Ian_w1: or Iag_w1:
Fluxes	Flux_A:
Magnetization currents	Imag_A:
Excitation currents	Iexc_A:

The same labels apply for winding 2, except that 1 is replaced by 2 in the labels.

If the **Winding 1 connection (ABC terminals)** parameter is set to Delta (D1) or Delta (D3), the labels are as follows.

Measurement	Label
Winding 1 voltages	Uab_w1:
Winding 1 currents	Iab_w1:
Flux linkages	Flux_A:
Magnetization currents	Imag_A:
Excitation currents	Iexc_A:

Parameters

Units — Units of block parameters

pu (default) | SI

Specify the units used to enter the parameters of this block. Select pu to use per unit. Select SI to use SI units. Changing the **Units** parameter from pu to SI, or from SI to pu, automatically converts the parameters displayed in the mask of the block. The per unit conversion is based on the transformer rated power Pn in VA, nominal frequency fn in Hz, and nominal voltage Vn, in Vrms, of the windings.

Nominal power and frequency — Nominal power rating and frequency

[250e6 , 60] (default)

Nominal power rating, in volt-amperes (VA), and nominal frequency, in hertz (Hz), of the transformer. The nominal parameters have no impact on the transformer model when the **Units** parameter is set to SI.

Winding 1 parameters — Parameters of winding 1

[735e3 , 0.002 , 0.08] (default) | [7.35e+05 4.3218 0.45856]

Phase-to-phase nominal voltage in volts RMS, resistance, and leakage inductance in pu for winding 1. Default is [735e3 , 0.002 , 0.08] when the **Units** parameter is pu and [7.35e+05 4.3218 0.45856] when the **Units** parameter is SI.

Winding 2 parameters — Parameters of winding 2

[315e3 , 0.002 , 0.08] (default) | [3.15e+05 0.7938 0.084225]

Phase-to-phase nominal voltage in volts RMS, resistance, and leakage inductance in pu for winding 2. Default is [315e3 , 0.002 , 0.08] when the **Units** parameter is pu and [3.15e+05 0.7938 0.084225] when the **Units** parameter is SI.

Magnetization resistance Rm — Magnetization resistance

500 (default) | 1.0805e+06

Magnetization resistance Rm, in pu. Default is 500 when the **Units** parameter is pu and 1.0805e+06 when the **Units** parameter is SI.

Magnetization inductance Lm — Magnetization inductance for nonsaturable core

500 (default) | 2866

Magnetization inductance Lm, in pu, for a nonsaturable core. Default is 500 when the **Units** parameter is pu and 2866 when the **Units** parameter is SI.

Dependencies

To enable this parameter, clear the **Saturable core** parameter.

Inductance L0 of zero-sequence flux path return — Inductance L0 of zero-sequence flux path return for three-limb core transformer type

0.5 (default) | 2.866

The Inductance L0 of zero-sequence flux path return, in pu, for the three-limb core transformer type.

Default is 0.5 when the **Units** parameter is pu and 2.866 when the **Units** parameter is SI.

Dependencies

To enable this parameter, set **Type** to Three-limb core (core type).

Saturation characteristic — Saturation characteristic

[0,0 ; 0.0024,1.2 ; 1.0,1.52] (default) | [0 0;0.66653 1910.3;277.72 2419.7]

Saturation characteristic. Default is [0,0 ; 0.0024,1.2 ; 1.0,1.52] when the **Units** parameter is pu and [0 0;0.66653 1910.3;277.72 2419.7] when the **Units** parameter is SI.

The saturation characteristic for the saturable core. Specify a series of current/ flux pairs (in pu) starting with the pair (0,0).

Dependencies

To enable this parameter, select the **Simulate saturation** parameter.

Initial fluxes — Initial fluxes for each phase of transformer

[0.8 , -0.8 , 0.7] (default) | [1273.5 -1273.5 1114.3]

Specify initial fluxes for each phase of the transformer. Default is [0.8 , -0.8 , 0.7] when the **Units** parameter is pu and [1273.5 -1273.5 1114.3] when the **Units** parameter is SI.

When the **Specify initial fluxes** parameter is not selected upon simulation, Simscape Electrical Specialized Power Systems software automatically computes the initial fluxes to start the simulation in steady state. The computed values are saved in the **Initial Fluxes** parameter and overwrite any previous values.

Dependencies

To enable this parameter, select the **Specify initial fluxes** and **Simulate saturation** parameters.

Advanced

The **Advanced** tab of the block is not visible when you set the **Simulation type** parameter of the powergui block to **Continuous**, or when you select the **Automatically handle discrete solver** parameter of the powergui block. The tab is visible when you set the **Simulation type** parameter of the powergui block to **Discrete**, and when the **Automatically handle discrete solver** parameter of the powergui block is cleared.

Break Algebraic loop in discrete saturation model — Option to break Algebraic loop in discrete saturation model

`off` (default) | `on`

When selected, a delay is inserted at the output of the saturation model computing magnetization current as a function of flux linkage (the integral of input voltage computed by a Trapezoidal method). This delay eliminates the algebraic loop resulting from trapezoidal discretization methods and speeds up the simulation of the model. However, this delay introduces a one simulation step time delay in the model and can cause numerical oscillations if the sample time is too large. The algebraic loop is required in most cases to get an accurate solution.

When cleared (default), the **Discrete solver model** parameter specifies the discretization method of the saturation model.

Discrete solver model — Method to resolve algebraic loop

`Trapezoidal iterative` (default) | `Trapezoidal robust` | `Backward Euler robust`

Select one of these methods to resolve the algebraic loop.

- **Trapezoidal iterative**—Although this method produces correct results, it is not recommended because Simulink tends to slow down and may fail to converge (simulation stops), especially when the number of saturable transformers is increased. Also, because of the Simulink algebraic loop constraint, this method cannot be used in real time. In R2018b and previous releases, you used this method when the **Break Algebraic loop in discrete saturation model** parameter was cleared.
- **Trapezoidal robust**—This method is slightly more accurate than the **Backward Euler robust** method. However, it may produce slightly damped numerical oscillations on transformer voltages when the transformer is at no load.
- **Backward Euler robust**—This method provides good accuracy and prevents oscillations when the transformer is at no load.

The maximum number of iterations for the robust methods is specified in the **Preferences** tab of the powergui block, in the **Solver details for nonlinear elements** section. For real time applications, you may need to limit the number of iterations. Usually, limiting the number of iterations to 2 produces acceptable results. The two robust solvers are the recommended methods for discretizing the saturation model of the transformer.

For more information on what method to use in your application, see "Simulating Discretized Electrical Systems".

Version History

Introduced before R2006a

References

[1] IEC. *International Standard IEC 60076-1, Power Transformer - Part 1: General*, Edition 2.1, 2000-04. "Annex D: Three-phase transformer connections." 2000.

Extended Capabilities

C/C++ Code Generation

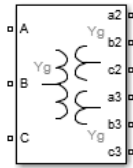
Generate C and C++ code using Simulink® Coder™.

See Also

Linear Transformer | Multimeter | Saturable Transformer | Three-Phase Transformer (Three Windings) | Three-Phase Transformer Inductance Matrix Type (Two Windings)

Three-Phase Transformer (Three Windings)

Implement three-phase transformer with configurable winding connections



Libraries:

Simscape / Electrical / Specialized Power Systems / Power Grid Elements

Description

This block implements a three-phase transformer by using three single-phase transformers with three windings. For a detailed description of the electrical model of a single-phase transformer, see the Linear Transformer block.

When activated, the saturation characteristic is the same as the one described for the Saturable Transformer block. If the fluxes are not specified, the initial values are automatically adjusted so that the simulation starts in steady state.

The leakage inductances and resistance of each winding are given in pu based on the transformer nominal power P_n and on the nominal voltage of the winding (V_1 , V_2 , or V_3). For an explanation of per units, refer to the Linear Transformer and to the Saturable Transformer.

The three windings of the transformer can be connected in the following manner:

- Y
- Y with accessible neutral (for windings 1 and 3 only)
- Grounded Y
- Delta (D1), delta lagging Y by 30 degrees
- Delta (D11), delta leading Y by 30 degrees

An input port labeled N is added to the block if you select the Y connection with accessible neutral for winding 1. If you ask for an accessible neutral on winding 3, an extra output port labeled n3 is generated.

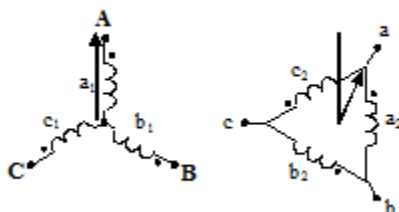
The D1 and D11 notations refer to the following clock convention. It assumes that the reference Y voltage phasor is at noon (12) on a clock display. D1 and D11 refer respectively to 1 PM (delta voltages lagging Y voltages by 30 degrees) and 11 AM (delta voltages leading Y voltages by 30 degrees).

Standard Notation for Winding Connections

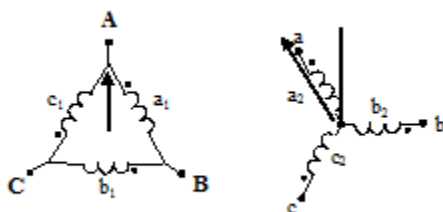
The conventional notation for a two-winding three-phase transformer uses two letters followed by a number. The first letter (Y or D) indicates a high-voltage wye or delta winding connection. The second letter (y or d) indicates a low-voltage wye or delta winding connection. The number, an integer between 0 and 12, indicates the position of the low-voltage positive-sequence voltage phasor on a clock display when the high-voltage positive-sequence voltage phasor is at 12:00.

The following three figures are examples of standard winding connections. The dots indicate polarity marks, and arrows indicate the position of phase A-to-neutral voltage phasors on high-voltage and low-voltage windings. The phasors are assumed to rotate in a counterclockwise direction so that rising numbers indicate increasing phase lag.

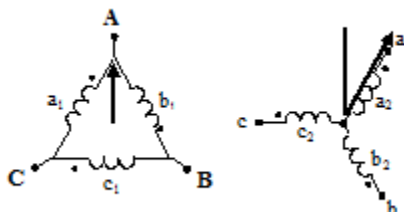
- Yd1: The low-voltage winding (d) is lagging high-voltage winding (Y) by 30 degrees. The **Winding 2 connection** parameter is set to **D1**.



- Dy11: The low-voltage winding (y) is leading high-voltage winding (D) by 30 degrees. The **Winding 1 connection** parameter is set to **D1**.



- Dy1: The low-voltage winding (y) is lagging high-voltage winding (D) by 30 degrees. The **Winding 1 connection** parameter is set to **D11**.

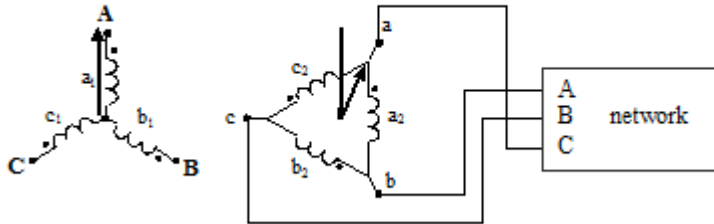


You can represent many other connections with phase shifts between 0 and 360 degrees (by steps of 30 degrees) by combining the +30- or -30-degree phase shift provided by the D1 and D11 block parameter settings and, in some cases, an additional +/-120-degree phase shift obtained by connecting the output terminals of delta winding to the appropriate phases of the network.

The table explains how to set up the Three-Phase Transformer block to obtain common connections.

Clock Position	Phase Shift (degrees)	Connection	Winding 1 Connection	Winding 2 Connection	Terminals of Delta Winding to Connect to Network ABC Phases
0	0	Yy0	Y	Y	—
		Dd0	D1	D1	abc
1	-30	Yd1	Y	D1	abc
		Dy1	D11	Y	abc
2	-60	Dd2	D11	D1	abc
5	-150	Yd5	Y	D1	bca
		Dy5	D11	Y	cab
7	+150	Yd7	Y	D11	cab
		Dy7	D1	Y	bca
10	+60	Dd10	D1	D11	abc
11	+30	Yd11	Y	D11	abc
		Dy11	D1	Y	abc

For example, to obtain the Yd5 connection, set the **Winding 1 connection** parameter to **Y** and the **Winding 2 connection** parameter to **D1**, and connect the network phases to the winding 2 as follows:



For more details on conventional transformer winding notations, see International Standard IEC 60076-1 [1].

Examples

The power_transfo3wdn circuit uses two Three-Phase Transformer blocks. Two Multimeter blocks are used to measure the phase A voltage (or AB for delta connections) of each winding

Ports

Conserving

A — Winding 1 phase A terminal
specialized electrical

Electrical conserving port associated with the phase A terminal of winding 1.

B — Winding 1 phase B terminal
specialized electrical

Electrical conserving port associated with the phase B terminal of winding 1.

C — Winding 1 phase C terminal
specialized electrical

Electrical conserving port associated with the phase C terminal of winding 1.

N — Winding 1 neutral terminal
specialized electrical

Electrical conserving port associated with the neutral terminal of winding 1.

Dependencies

To enable this port, set the **Winding 1 connection (ABC terminals)** parameter to Yn.

a2 — Winding 2 phase A terminal
specialized electrical

Electrical conserving port associated with the phase A terminal of winding 2.

b2 — Winding 2 phase B terminal
specialized electrical

Electrical conserving port associated with the phase B terminal of winding 2.

c2 — Winding 2 phase C terminal
specialized electrical

Electrical conserving port associated with the phase C terminal of winding 2.

n2 — Winding 2 neutral terminal
specialized electrical

Electrical conserving port associated with the neutral terminal of winding 2.

Dependencies

To enable this port, set the **Winding 2 connection (abc-2 terminals)** parameter to Yn.

a3 — Winding 3 phase A terminal
specialized electrical

Electrical conserving port associated with the phase A terminal of winding 3.

b3 — Winding 3 phase B terminal
specialized electrical

Electrical conserving port associated with the phase B terminal of winding 3.

c3 — Winding 3 phase C terminal
specialized electrical

Electrical conserving port associated with the phase C terminal of winding 3.

n3 — Winding 3 neutral terminal
specialized electrical

Electrical conserving port associated with the neutral terminal of winding 3.

Dependencies

To enable this port, set the **Winding 3 connection (abc-3 terminals)** parameter to Yn.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Configuration

Winding 1 connection (ABC terminals) — Winding connections for winding 1
Yg (default) | Yn | Y | Delta (D1) | Delta (D11)

Winding connections for winding 1. Choices are Y, Yn, Yg (default), Delta (D1), and Delta (D11).

Winding 2 connection (abc-2 terminals) — Winding connections for winding 2
Yg (default) | YnY | Delta (D1) | Delta (D11)

Winding connections for winding 2. Choices are Y, Yn, Yg (default), Delta (D1), and Delta (D11).

Winding 3 connection (abc-3 terminals) — Winding connections for winding 3
Yg (default) | YnY | Delta (D1) | Delta (D11)

Winding connections for winding 3. Choices are Y, Yn, Yg (default), Delta (D1), and Delta (D11).

Type — Type of transformer

Three single-phase transformers (default) | Three-limb core (core-type) | Five-limb core (shell-type)

Select **Three single-phase transformers (default)** to implement a three-phase transformer using three single-phase transformer models. You can use this core type to represent very large power transformers found in utility grids (hundreds of MW).

Select **Three-limb core (core type)** to implement a three-limb core three-phase transformer. In most applications, three-phase transformers use a three-limb core (core-type transformer). This type of core produces accurate results during an asymmetrical fault for both linear and nonlinear models (including saturation). During asymmetrical voltage conditions, the zero-sequence flux of a core-type transformer returns outside the core, through an air gap, structural steel, and a tank. Thus, the natural zero-sequence inductance L_0 (without delta winding) of such a core-type transformer is usually very low (typically $0.5 \text{ pu} < L_0 < 2 \text{ pu}$) compared with a three-phase transformer using three single-phase units ($L_0 > 100 \text{ pu}$). This low L_0 value affects voltages, currents, and flux unbalances during linear and saturated operation.

Select **Five-limb core (shell type)** to implement a five-limb core three-phase transformer. On rare occasions, very large transformers are built with a five-leg core (three phase legs and two external legs). This core configuration, also known as shell type, is chosen mainly to reduce the height of the transformer and make transportation easier. During unbalanced voltage conditions, as opposed to the three-limb transformer, the zero-sequence flux of the five-limb transformer stays

inside the steel core and returns through the two external limbs. The natural zero-sequence inductance (without delta) is therefore very high ($L_0 > 100$ pu). Except for small current unbalances due to core asymmetry, the behavior of the five-limb shell-type transformer is similar to that of a three-phase transformer built with three single-phase units.

Simulate saturation — Option to simulate saturation

off (default) | on

If selected, implements a saturable three-phase transformer.

Simulate hysteresis — Option to simulate hysteresis

off (default) | on

Select to model a saturation characteristic including hysteresis instead of a single-valued saturation curve.

Dependencies

To enable this parameter, select the **Simulate saturation** parameter.

Hysteresis Mat file (ex: 'hysteresis.mat') — File containing data for use in hysteresis model

'hysteresis' (default)

Specify a .mat file containing the data for use in the hysteresis model. When you open the **Hysteresis Design Tool** of the Powergui block, the default hysteresis loop and parameters saved in the hysteresis.mat file are displayed. Use the **Load** button of the Hysteresis Design tool to load another .mat file. Use the **Save** button of the Hysteresis Design tool to save your model in a new .mat file.

Dependencies

To enable this parameter, select the **Simulate hysteresis** parameter.

Specify initial fluxes — Option to specify initial fluxes

off (default) | on

If selected, the initial fluxes are defined by the **Initial fluxes** parameter on the **Parameters** tab.

When the **Specify initial fluxes** parameter is not selected upon simulation, Simscape Electrical Specialized Power Systems software automatically computes the initial fluxes to start the simulation in steady state. The computed values are saved in the **Initial Fluxes** parameter and will overwrite any previous values.

Dependencies

To enable this parameter, select the **Simulate saturation** parameter.

Measurements — Type of measurements

None (default) | Winding voltages | Winding currents | Fluxes and excitation currents (Imag + IRm) | Fluxes and magnetization currents (Imag) | All measurements (V I Fluxes)

Select **Winding voltages** to measure the voltage across the winding terminals of the Three-Phase Transformer block.

Select **Winding currents** to measure the current flowing through the windings of the Three-Phase Transformer block.

Select **Fluxes** and excitation currents ($I_{\text{mag}} + I_{\text{Rm}}$) to measure the flux linkage, in volt seconds (Vs), and the total excitation current including iron losses modeled by R_m .

Select **Fluxes** and magnetization currents (I_{mag}) to measure the flux linkage, in volt seconds (Vs), and the magnetization current, in amperes (A), not including iron losses modeled by R_m .

Select **All measurements** (V , I , Flux) to measure the winding voltages, currents, magnetization currents, and the flux linkages.

Place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurements** list box of the Multimeter block, the measurements are identified by a label followed by the block name.

If the **Winding 1 connection (ABC terminals)** parameter is set to Y , Y_n , or Y_g , the labels are as follows.

Measurement	Label
Winding 1 voltages	$U_{\text{an\_w1}}$ or $U_{\text{ag\_w1}}$
Winding 1 currents	$I_{\text{an\_w1}}$ or $I_{\text{ag\_w1}}$
Flux linkages	Flux_A
Magnetization currents	$I_{\text{mag\_A}}$
Excitation currents	$I_{\text{exc\_A}}$

The same labels apply for winding 2 and winding 3, except that the 1 is replaced by 2 or by 3 in the labels.

If the **Winding 1 connection (ABC terminals)** parameter is set to **Delta (D11)** or **Delta (D1)**, the labels are as follows.

Measurement	Label
Winding 1 voltages	$U_{\text{ab\_w1}}$
Winding 1 currents	$I_{\text{ab\_w1}}$
Flux linkages	Flux_A
Magnetization currents	$I_{\text{mag\_A}}$
Excitation currents	$I_{\text{exc\_A}}$

Parameters

Units — Units of block parameters
 pu (default) | SI

Specify the units used to enter the parameters of this block. Select pu to use per unit. Select SI to use SI units. Changing the **Units** parameter from pu to SI , or from SI to pu , automatically converts

the parameters displayed in the mask of the block. The per unit conversion is based on the transformer rated power P_n in VA, nominal frequency f_n in Hz, and nominal voltage V_n , in V_{rms} , of the windings.

Nominal power and frequency — Nominal power rating and frequency

[250e6 , 60] (default)

Nominal power rating, in volt-amperes (VA), and nominal frequency, in hertz (Hz), of the transformer. The nominal parameters have no impact on the transformer model when the **Units** parameter is set to SI.

Winding 1 parameters — Parameters of winding 1

[735e3 , 0.002 , 0.08] (default) | [7.35e+05 4.3218 0.45856]

Phase-to-phase nominal voltage in volts RMS, resistance, and leakage inductance in pu for winding 1. Default is [735e3 , 0.002 , 0.08] when the **Units** parameter is pu and [7.35e+05 4.3218 0.45856] when the **Units** parameter is SI.

Winding 2 parameters — Parameters of winding 2

[315e3 , 0.002 , 0.08] (default) | [3.15e+05 0.7938 0.084225]

Phase-to-phase nominal voltage in volts RMS, resistance, and leakage inductance in pu for winding 2. Default is [315e3 , 0.002 , 0.08] when the **Units** parameter is pu and [3.15e+05 0.7938 0.084225] when the **Units** parameter is SI.

Winding 3 parameters — Parameters of winding 3

[315e3 , 0.002 , 0.08] (default) | [3.15e+05 0.7938 0.084225]

Phase-to-phase nominal voltage in volts RMS, resistance, and leakage inductance in pu for winding 3. Default is [315e3 , 0.002 , 0.08] when the **Units** parameter is pu and [3.15e+05 0.7938 0.084225] when the **Units** parameter is SI.

Magnetization resistance Rm — Magnetization resistance

500 (default) | 1.0805e+06

Magnetization resistance R_m , in pu. Default is 500 when the **Units** parameter is pu and 1.0805e+06 when the **Units** parameter is SI.

Magnetization inductance Lm — Magnetization inductance for nonsaturable core

500 (default) | 2866

Magnetization inductance L_m , in pu, for a nonsaturable core. Default is 500 when the **Units** parameter is pu and 2866 when the **Units** parameter is SI.

Dependencies

To enable this parameter, clear the **Saturable core** parameter.

Inductance L0 of zero-sequence flux path return — Inductance L0 of zero-sequence flux path return for three-limb core transformer type

0.5 (default) | 2.866

The Inductance L0 of zero-sequence flux path return, in pu, for the three-limb core transformer type.

Default is 0.5 when the **Units** parameter is pu and 2.866 when the **Units** parameter is SI.

Dependencies

To enable this parameter, set **Type** to **Three-limb core** (core type).

Saturation characteristic — Saturation characteristic

[0,0 ; 0.0024,1.2 ; 1.0,1.52] (default) | [0 0;0.66653 1910.3;277.72 2419.7]

Saturation characteristic. Default is [0,0 ; 0.0024,1.2 ; 1.0,1.52] when the **Units** parameter is pu and [0 0;0.66653 1910.3;277.72 2419.7] when the **Units** parameter is SI.

The saturation characteristic for the saturable core. Specify a series of current/ flux pairs (in pu) starting with the pair (0,0).

Dependencies

To enable this parameter, select the **Simulate saturation** parameter.

Initial fluxes — Initial fluxes for each phase of transformer

[0.8 , -0.8 , 0.7] (default) | [1273.5 -1273.5 1114.3]

Specify initial fluxes for each phase of the transformer. Default is [0.8 , -0.8 , 0.7] when the **Units** parameter is pu and [1273.5 -1273.5 1114.3] when the **Units** parameter is SI.

When the **Specify initial fluxes** parameter is not selected upon simulation, Simscape Electrical Specialized Power Systems software automatically computes the initial fluxes to start the simulation in steady state. The computed values are saved in the **Initial Fluxes** parameter and overwrite any previous values.

Dependencies

To enable this parameter, select the **Specify initial fluxes** and **Simulate saturation** parameters.

Advanced

The **Advanced** tab of the block is not visible when you set the **Simulation type** parameter of the powergui block to **Continuous**, or when you select the **Automatically handle discrete solver** parameter of the powergui block. The tab is visible when you set the **Simulation type** parameter of the powergui block to **Discrete**, and when the **Automatically handle discrete solver** parameter of the powergui block is cleared.

Break Algebraic loop in discrete saturation model — Option to break Algebraic loop in discrete saturation model

off (default) | on

When selected, a delay is inserted at the output of the saturation model computing magnetization current as a function of flux linkage (the integral of input voltage computed by a Trapezoidal method). This delay eliminates the algebraic loop resulting from trapezoidal discretization methods and speeds up the simulation of the model. However, this delay introduces a one simulation step time delay in the model and can cause numerical oscillations if the sample time is too large. The algebraic loop is required in most cases to get an accurate solution.

When cleared (default), the **Discrete solver model** parameter specifies the discretization method of the saturation model.

Discrete solver model — Method to resolve algebraic loop

Trapezoidal iterative (default) | Trapezoidal robust | Backward Euler robust

Select one of these methods to resolve the algebraic loop.

- **Trapezoidal iterative**—Although this method produces correct results, it is not recommended because Simulink tends to slow down and may fail to converge (simulation stops), especially when the number of saturable transformers is increased. Also, because of the Simulink algebraic loop constraint, this method cannot be used in real time. In R2018b and previous releases, you used this method when the **Break Algebraic loop in discrete saturation model** parameter was cleared.
- **Trapezoidal robust**—This method is slightly more accurate than the **Backward Euler robust** method. However, it may produce slightly damped numerical oscillations on transformer voltages when the transformer is at no load.
- **Backward Euler robust**—This method provides good accuracy and prevents oscillations when the transformer is at no load.

The maximum number of iterations for the robust methods is specified in the **Preferences** tab of the powergui block, in the **Solver details for nonlinear elements** section. For real time applications, you may need to limit the number of iterations. Usually, limiting the number of iterations to 2 produces acceptable results. The two robust solvers are the recommended methods for discretizing the saturation model of the transformer.

For more information on what method to use in your application, see “Simulating Discretized Electrical Systems”.

Version History

Introduced before R2006a

Extended Capabilities

C/C++ Code Generation

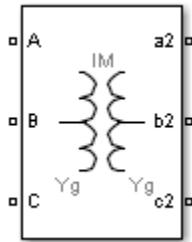
Generate C and C++ code using Simulink® Coder™.

See Also

Linear Transformer | Multimeter | Saturable Transformer | Three-Phase Transformer (Two Windings) | Three-Phase Transformer Inductance Matrix Type (Three Windings)

Three-Phase Transformer Inductance Matrix Type (Two Windings)

Implement three-phase two-winding transformer with configurable winding connections and core geometry

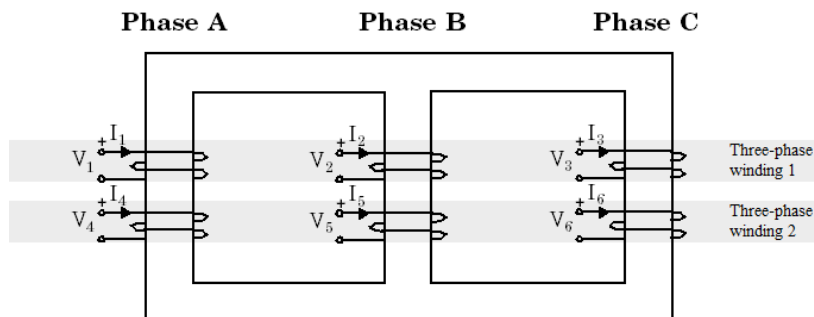


Library

Simscape / Electrical / Specialized Power Systems / Power Grid Elements

Description

The Three-Phase Transformer Inductance Matrix Type (Two Windings) block is a three-phase transformer with a three-limb core and two windings per phase. Unlike the Three-Phase Transformer (Two Windings) block, which is modeled by three separate single-phase transformers, this block takes into account the couplings between windings of different phases. The transformer core and windings are shown in the following illustration.



The phase windings of the transformer are numbered as follows:

- 1 and 4 on phase A
- 2 and 5 on phase B
- 3 and 6 on phase C

This core geometry implies that phase winding 1 is coupled to all other phase windings (2 to 6), whereas in Three-Phase Transformer (Two Windings) block (a three-phase transformer using three independent cores) winding 1 is coupled only with winding 4.

Note The phase winding numbers 1 and 2 should not be confused with the numbers used to identify the three-phase windings of the transformer. Three-phase winding 1 consists of phase windings 1,2,3, and three-phase winding 2 consists of phase windings 4,5,6.

Transformer Model

The Three-Phase Transformer Inductance Matrix Type (Two Windings) block implements the following matrix relationship:

$$\begin{bmatrix} V_1 \\ V_2 \\ \vdots \\ V_6 \end{bmatrix} = \begin{bmatrix} R_1 & 0 & \dots & 0 \\ 0 & R_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & R_6 \end{bmatrix} \cdot \begin{bmatrix} I_1 \\ I_2 \\ \vdots \\ I_6 \end{bmatrix} + \begin{bmatrix} L_{11} & L_{12} & \dots & L_{16} \\ L_{21} & L_{22} & \dots & L_{26} \\ \vdots & \vdots & \ddots & \vdots \\ L_{61} & L_{62} & \dots & L_{66} \end{bmatrix} \cdot \frac{d}{dt} \begin{bmatrix} I_1 \\ I_2 \\ \vdots \\ I_6 \end{bmatrix}.$$

R_1 to R_6 represent the winding resistances. The self-inductance terms L_{ii} and the mutual inductance terms L_{ij} are computed from the voltage ratios, the inductive component of the no load excitation currents and the short-circuit reactances at nominal frequency. Two sets of values in positive-sequence and in zero-sequence allow calculation of the 6 diagonal terms and 15 off-diagonal terms of the symmetrical inductance matrix.

When the parameter **Core type** is set to **Three single-phase cores**, the model uses two independent circuits with (3x3) R and L matrices. In this condition, the positive-sequence and zero-sequence parameters are identical and you only specify positive-sequence values.

The self and mutual terms of the (6x6) L matrix are obtained from excitation currents (one three-phase winding is excited and the other three-phase winding is left open) and from positive- and zero-sequence short-circuit reactances X_{112} and X_{012} measured with three-phase winding 1 excited and three-phase winding 2 short-circuited.

Assuming the following positive-sequence parameters:

$Q1_1$ = Three-phase reactive power absorbed by winding 1 at no load when winding 1 is excited by a positive-sequence voltage V_{nom1} with winding 2 open

$Q1_2$ = Three-phase reactive power absorbed by winding 2 at no load when winding 2 is excited by a positive-sequence voltage V_{nom2} with winding 1 open

X_{112} = Positive-sequence short-circuit reactance seen from winding 1 when winding 2 is short-circuited

V_{nom1} , V_{nom2} = Nominal line-line voltages of windings 1 and 2

The positive-sequence self and mutual reactances are given by:

$$X_1(1, 1) = \frac{V_{nom1}^2}{Q1_1}$$

$$X_1(2, 2) = \frac{V_{nom2}^2}{Q1_2}$$

$$X_1(1, 2) = X_1(2, 1) = \sqrt{X_1(2, 2) \cdot (X_1(1, 1) - X_{112})}.$$

The zero-sequence self-reactances $X_0(1,1)$, $X_0(2,2)$, and mutual reactance $X_0(1,2) = X_0(2,1)$ are also computed using similar equations.

Extension from the following two (2x2) reactance matrices in positive-sequence and in zero-sequence

$$\begin{bmatrix} X_1(1,1) & X_1(1,2) \\ X_1(2,1) & X_1(2,2) \end{bmatrix} \quad \begin{bmatrix} X_0(1,1) & X_0(1,2) \\ X_0(2,1) & X_0(2,2) \end{bmatrix}$$

to a (6x6) matrix, is performed by replacing each of the four $[X_1 \ X_0]$ pairs by a (3x3) submatrix of the form:

$$\begin{bmatrix} X_s & X_m & X_m \\ X_m & X_s & X_m \\ X_m & X_m & X_s \end{bmatrix}$$

where the self and mutual terms are given by:

$$X_s = (X_0 + 2X_1)/3$$

$$X_m = (X_0 - X_1)/3$$

In order to model the core losses (active power $P1$ and $P0$ in positive- and zero-sequences), additional shunt resistances are also connected to terminals of one of the three-phase windings. If winding 1 is selected, the resistances are computed as:

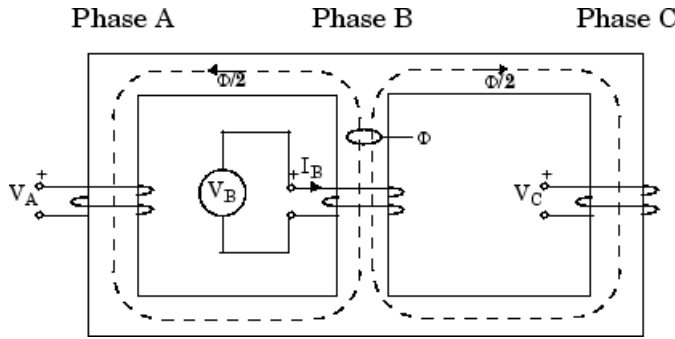
$$R1_1 = \frac{V_{nom1}^2}{P1_1} \quad R0_1 = \frac{V_{nom1}^2}{P0_1}.$$

The block takes into account the connection type you select, and the icon of the block is automatically updated. An input port labeled N is added to the block if you select the Y connection with accessible neutral for winding 1. If you ask for an accessible neutral on winding 2, an extra output port labeled n2 is generated.

Excitation Current in Zero Sequence

Often, the zero-sequence excitation current of a transformer with a 3-limb core is not provided by the manufacturer. In such a case, a reasonable value can be guessed as explained below.

The following figure shows a three-limb core with a single three-phase winding. Only phase B is excited and voltage is measured on phase A and phase C. The flux Φ produced by phase B shares equally between phase A and phase C so that $\Phi/2$ is flowing in limb A and in limb C. Therefore, in this particular case, if leakage inductance of winding B would be zero, voltage induced on phases A and C would be $-k.V_B = -V_B/2$. In fact, because of the leakage inductance of the three windings, the average value of induced voltage ratio k when windings A, B, and C are successively excited must be slightly lower than 0.5.



Assume:

Z_s = average value of the three self-impedances

Z_m = average value of mutual impedance between phases

Z_1 = positive-sequence impedance of three-phase winding

Z_0 = zero-sequence impedance of three-phase winding

I_1 = positive-sequence excitation current

I_0 = zero-sequence excitation current

$$V_B = Z_s I_B$$

$$V_A = Z_m I_B = -V_B/2$$

$$V_C = Z_m I_B = -V_B/2$$

$$Z_s = \frac{2Z_1 + Z_0}{3}$$

$$Z_m = \frac{Z_0 - Z_1}{3}$$

$$V_A = V_C = \frac{Z_m}{Z_s} V_B = -\frac{\frac{Z_1}{Z_0} - 1}{2\frac{Z_1}{Z_0} + 1} V_B = -\frac{\frac{I_0}{I_1} - 1}{2\frac{I_0}{I_1} + 1} V_B = -k V_B,$$

where k = ratio of induced voltage (with k slightly lower than 0.5)

Therefore, the I_0/I_1 ratio can be deduced from k :

$$\frac{I_0}{I_1} = \frac{1 + k}{1 - 2k}.$$

Obviously k cannot be exactly 0.5 because this would lead to an infinite zero-sequence current. Also, when the three windings are excited with a zero-sequence voltage, the flux path should return through the air and tank surrounding the iron core. The high reluctance of the zero-sequence flux path results in a high zero-sequence current.

Let us assume $I_1 = 0.5\%$. A reasonable value for I_0 could be 100%. Therefore $I_0/I_1 = 200$. According to the equation for I_0/I_1 given above, one can deduce the value of k . $k = (200 - 1) / (2 * 200 + 1) = 199 / 401 = 0.496$.

Zero-sequence losses should be also higher than the positive-sequence losses because of the additional eddy current losses in the tank.

Finally, the value of the zero-sequence excitation current and the value of the zero-sequence losses are not critical if the transformer has a winding connected in Delta because this winding acts as a short circuit for zero-sequence.

Winding Connections

The three-phase windings of the transformer can be connected in the following manner:

- Y
- Y with accessible neutral
- Grounded Y
- Delta (D1), delta lagging Y by 30 degrees
- Delta (D11), delta leading Y by 30 degrees

Note The D1 and D11 notations refer to the following clock convention. It assumes that the reference Y voltage phasor is at noon (12) on a clock display. D1 and D11 refer respectively to 1 PM (delta voltages lagging Y voltages by 30 degrees) and 11 AM (delta voltages leading Y voltages by 30 degrees).

Parameters

Configuration Tab

Core type

Select the core geometry: Three single-phase cores or Three-limb or five-limb core (default). If you select the first option, only the positive-sequence parameters are used to compute the inductance matrix. If you select the second option, both the positive- and zero-sequence parameters are used.

Winding 1 connection

The winding connection for three-phase winding 1. Choices are Y, Yn, Yg (default), Delta (D1), and Delta (D11).

Winding 2 connection

The winding connection for three-phase winding 2. Choices are Y, Yn, Yg (default), Delta (D1), and Delta (D11).

Connect windings 1 and 2 in autotransformer

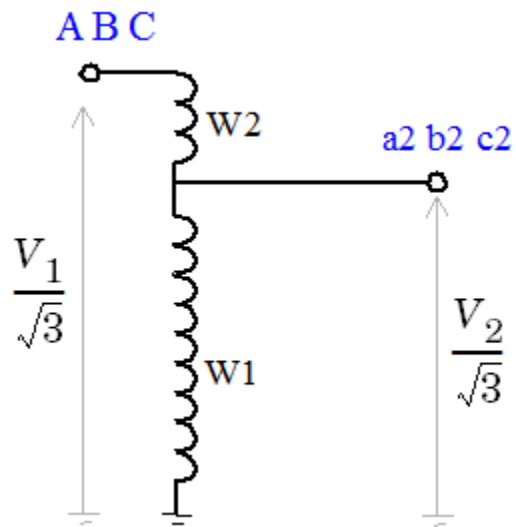
Select to connect the three-phase windings 1 and 2 in autotransformer (three-phase windings 1 and 2 in series with additive voltage). Default is cleared.

If the first voltage specified in the **Nominal line-line voltages** parameter is higher than the second voltage, the low voltage tap is connected on the right side (a2,b2,c2 terminals). Otherwise, the low voltage tap is connected on the left side (A,B,C terminals).

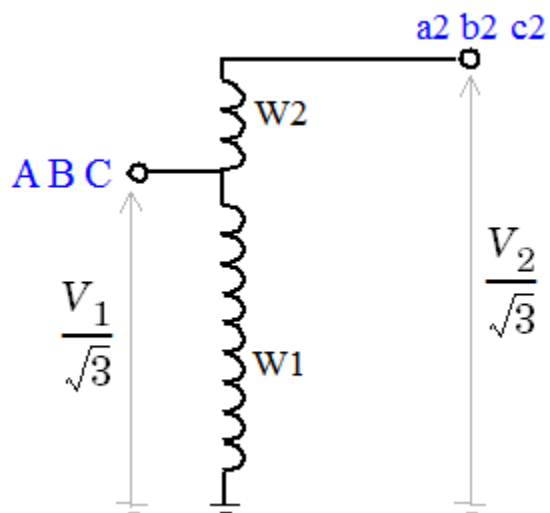
In autotransformer mode you must specify the same winding connections for the three-phase windings 1 and 2. If you select Yn connection for both winding 1 and winding 2, the common neutral N connector is displayed on the left side.

The following figure illustrates winding connections for one phase of an autotransformer when the three-phase windings are both connected in Yg.

If $V_1 > V_2$:



If $V_2 > V_1$:



Windings W1, W2, W3 correspond to the following phase winding numbers:

- Phase A: W1=1, W2=4
- Phase B: W1=2, W2=5
- Phase C: W1=3, W2=6

Measurements

Select Winding voltages to measure the voltage across the winding terminals of the Three-Phase Transformer block.

Select **Winding currents** to measure the current flowing through the windings of the Three-Phase Transformer block.

Select **All measurements** to measure the winding voltages and currents.

Default is **None**.

Place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurements** list box of the Multimeter block, the measurements are identified by a label followed by the block name.

If the **Winding 1 connection** parameter is set to **Y**, **Yn**, or **Yg**, the labels are as follows.

Measurement	Label
Winding 1 voltages	Uan_w1: or Uag_w1:
Winding 1 currents	Ian_w1: or Iag_w1:

If the **Winding 1 connection** parameter is set to **Delta (D11)** or **Delta (D1)**, the labels are as follows.

Measurement	Label
Winding 1 voltages	Uab_w1:
Winding 1 currents	Iab_w1:

The same labels apply for three-phase winding 2, except that 1 is replaced by 2 in the labels.

Parameters Tab

Nominal power and frequency

The nominal power rating, in volt-amperes (VA), and nominal frequency, in hertz (Hz), of the transformer. Default is [100e3, 60].

Nominal line-line voltages [V1 V2]

The phase-to-phase nominal voltages of windings 1 and 2 in volts RMS. Default is [2400, 600].

Winding resistances [R1 R2]

The resistances in pu for windings 1 and 2. Default is [0.01, 0.01].

Positive-sequence no-load excitation current

The no-load excitation current in percent of the nominal current when positive-sequence nominal voltage is applied at any three-phase winding terminals (ABC or abc2). Default is 2.

Positive-sequence no-load losses

The core losses plus winding losses at no-load, in watts (W), when positive-sequence nominal voltage is applied at any three-phase winding terminals (ABC or abc2). Default is 1000.

Positive-sequence short-circuit reactance

The positive-sequence short-circuit reactances X_{12} in pu. X_{12} is the reactance measured from winding 1 when winding 2 is short-circuited. Default is 0.06.

When the **Connect windings 1 and 2 in autotransformer** parameter is selected, the short-circuit reactances is labeled XHL. H and L indicate respectively the high voltage winding (either winding 1 or winding 2) and the low voltage winding (either winding 1 or winding 2).

Zero-sequence no-load excitation current with Delta windings opened

The no-load excitation current in percent of the nominal current when zero-sequence nominal voltage is applied at any three-phase winding terminals (ABC or abc2) connected in Yg or Yn. Default is 100.

Note If your transformer contains delta-connected windings (D1 or D11), the zero-sequence current flowing into the Yg or Yn winding connected to the zero-sequence voltage source does not represent the net excitation current because zero-sequence currents are also flowing in the delta winding. Therefore, you must specify the no-load zero-sequence circulation current obtained with the delta windings open.

If you want to measure this excitation current, you must temporarily change the delta windings connections from D1 or D11 to Y, Yg, or Yn, and connect the excited winding in Yg or Yn to provide a return path for the source zero-sequence currents.

Zero-sequence no-load losses with Delta windings opened

The core losses plus winding losses at no-load, in watts (W), when zero-sequence nominal voltage is applied at any group of winding terminals (ABC or abc2) connected in Yg or Yn. The Delta winding must be temporarily open to measure these losses. Default is 1500.

Note If your transformer contains delta-connected windings (D1 or D11), the zero-sequence current flowing into the Yg or Yn winding connected to the zero-sequence voltage source does not represent the net excitation current because zero-sequence currents are also flowing in the delta winding. Therefore, you must specify the no-load zero-sequence circulation current obtained with the delta windings open.

Zero-sequence short-circuit reactance

The zero-sequence short-circuit reactance X_{12} in pu. X_{12} is the reactance measured from winding 1 when winding 2 is short-circuited. Default is 0.03.

When the **Connect windings 1 and 2 in autotransformer** parameter is selected, the short-circuit reactances is labeled XHL. H and L indicate respectively the high voltage winding (either winding 1 or winding 2) and the low voltage winding (either winding 1 or winding 2).

Limitations

This transformer model does not include saturation. If you need modeling saturation, connect the primary winding of a saturable Three-Phase Transformer (Two Windings) in parallel with the primary winding of your model. Use the same connection (Yg, D1 or D11) and same winding resistance for the two windings connected in parallel. Specify the Y or Yg connection for the secondary winding and leave it open. Specify appropriate voltage, power ratings, and saturation characteristics that you

want. The saturation characteristic is obtained when the transformer is excited by a positive-sequence voltage.

If you are modeling a transformer with three single-phase cores or a five-limb core, this model produces acceptable saturation currents because flux stays trapped inside the iron core.

For a three-limb core, this saturation model still produces acceptable results, even if zero-sequence flux circulates outside of the core and returns through the air and the transformer tank surrounding the iron core. As the zero-sequence flux circulates in the air, the magnetic circuit is mainly linear and its reluctance is high (high magnetizing currents). These high zero-sequence currents (100% and more of nominal current) required to magnetize the air path are already taken into account in the linear model. Connecting a saturable transformer outside the three-limb linear model with a flux-current characteristic obtained in positive sequence produces currents required for magnetization of the iron core. This model gives acceptable results whether the three-limb transformer has a delta or not.

See Also

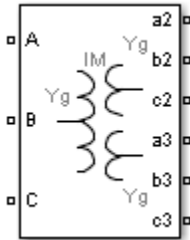
Linear Transformer, Multimeter, Three-Phase Transformer (Two Windings), Three-Phase Transformer (Three Windings), Three-Phase Transformer Inductance Matrix Type (Three Windings)

Version History

Introduced in R2008a

Three-Phase Transformer Inductance Matrix Type (Three Windings)

Implement three-phase three-winding transformer with configurable winding connections and core geometry

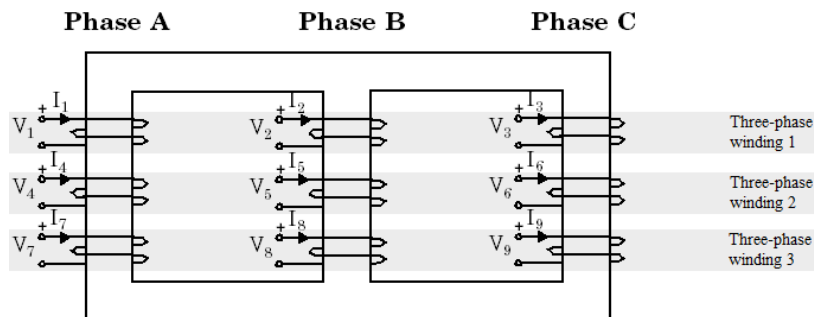


Library

Simscape / Electrical / Specialized Power Systems / Power Grid Elements

Description

The Three-Phase Transformer Inductance Matrix Type (Three Windings) block is a three-phase transformer with a three-limb core and three windings per phase. Unlike the Three-Phase Transformer (Three Windings) block, which is modeled by three separate single-phase transformers, this block takes into account the couplings between windings of different phases. The transformer core and windings are shown in the following illustration.



The phase windings of the transformer are numbered as follows:

- 1, 4, 7 on phase A
- 2, 5, 8 on phase B
- 3, 6, 9 on phase C

This core geometry implies that phase winding 1 is coupled to all other phase windings (2 to 9), whereas in Three-Phase Transformer (Three Windings) block (a three-phase transformer using three independent cores) winding 1 is coupled only with windings 4 and 7.

Note Do not confuse the phase winding numbers (1, 2, and 3) with the numbers used to identify the three-phase windings of the transformer. Three-phase winding 1 consists of phase windings 1, 2, 3, three-phase winding 2 consists of phase windings 4, 5, 6, and three-phase winding 3 consists of phase windings 7, 8, 9.

Transformer Model

The Three-Phase Transformer Inductance Matrix Type (Three-Windings) block implements the following matrix relationship:

$$\begin{bmatrix} V_1 \\ V_2 \\ \vdots \\ V_9 \end{bmatrix} = \begin{bmatrix} R_1 & 0 & \dots & 0 \\ 0 & R_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & R_9 \end{bmatrix} \cdot \begin{bmatrix} I_1 \\ I_2 \\ \vdots \\ I_9 \end{bmatrix} + \begin{bmatrix} L_{11} & L_{12} & \dots & L_{19} \\ L_{21} & L_{22} & \dots & L_{29} \\ \vdots & \vdots & \ddots & \vdots \\ L_{91} & L_{92} & \dots & L_{99} \end{bmatrix} \cdot \frac{d}{dt} \begin{bmatrix} I_1 \\ I_2 \\ \vdots \\ I_9 \end{bmatrix}.$$

R_1 to R_9 represent the winding resistances. The self-inductance terms L_{ii} and the mutual inductance terms L_{ij} are computed from the voltage ratios, the inductive component of the no load excitation currents and the short-circuit reactances at nominal frequency. Two sets of values in positive-sequence and in zero-sequence allow calculation of the 9 diagonal terms and 36 off-diagonal terms of the symmetrical inductance matrix.

When the parameter **Core type** is set to **Three single-phase cores**, the model uses three independent circuits with (3x3) R and L matrices. In this condition, the positive-sequence and zero-sequence parameters are identical and you need only specifying positive-sequence values.

The self and mutual terms of the (9x9) L matrix are obtained from excitation currents (one three-phase winding is excited and the other two three-phase windings are left open) and from short-circuit reactances.

The following short-circuit reactances are specified in the mask parameters:

$X_{1_{12}}, X_{0_{12}}$ — positive- and zero-sequence reactances measured with three-phase winding 1 excited and three-phase winding 2 short-circuited

$X_{1_{13}}, X_{0_{13}}$ — positive- and zero-sequence reactances measured with three-phase winding 1 excited and three-phase winding 3 short-circuited

$X_{1_{23}}, X_{0_{23}}$ — positive- and zero-sequence reactances measured with three-phase winding 2 excited and three-phase winding 3 short-circuited

Assuming the following positive-sequence parameters for three-phase windings i and j (where i=1, 2, or 3 and j=1, 2, or 3):

Q_{1i} = Three-phase reactive power absorbed by winding i at no load when winding i is excited by a positive-sequence voltage V_{nom_i} with winding j open

Q_{1j} = Three-phase reactive power absorbed by winding j at no load when winding j is excited by a positive-sequence voltage V_{nom_j} with winding i open

X_{1ij} = positive-sequence short-circuit reactance seen from winding i when winding j is short-circuited

V_{nom_i}, V_{nom_j} = nominal line-line voltages of windings i and j.

The positive-sequence self and mutual reactances are given by:

$$X_1(i, i) = \frac{V_{\text{nom}i}^2}{Q1_i}$$

$$X_1(j, j) = \frac{V_{\text{nom}j}^2}{Q1_j}$$

$$X_1(i, j) = X_1(j, i) = \sqrt{X_1(j, j) \cdot (X_1(i, i) - X_{1ij})}.$$

The zero-sequence self-reactances $X_0(i, i)$, $X_0(j, j)$, and mutual reactance $X_0(i, j) = X_0(j, i)$ are also computed using similar equations.

Extension from the following two (3x3) reactance matrices in positive-sequence and in zero-sequence

$$\begin{bmatrix} X_1(1, 1) & X_1(1, 2) & X_1(1, 3) \\ X_1(2, 1) & X_1(2, 2) & X_1(2, 3) \\ X_1(3, 1) & X_1(3, 2) & X_1(3, 3) \end{bmatrix}$$

$$\begin{bmatrix} X_0(1, 1) & X_0(1, 2) & X_0(1, 3) \\ X_0(2, 1) & X_0(2, 2) & X_0(2, 3) \\ X_0(3, 1) & X_0(3, 2) & X_0(3, 3) \end{bmatrix}$$

to a (9x9) matrix, is performed by replacing each of the nine $[X_1 \ X_0]$ pairs by a (3x3) submatrix of the form:

$$\begin{bmatrix} X_s & X_m & X_m \\ X_m & X_s & X_m \\ X_m & X_m & X_s \end{bmatrix}$$

where the self and mutual terms are given by:

$$X_s = (X_0 + 2X_1)/3$$

$$X_m = (X_0 - X_1)/3$$

In order to model the core losses (active power $P1$ and $P0$ in positive- and zero-sequences), additional shunt resistances are also connected to terminals of one of the three-phase windings. If winding i is selected, the resistances are computed as:

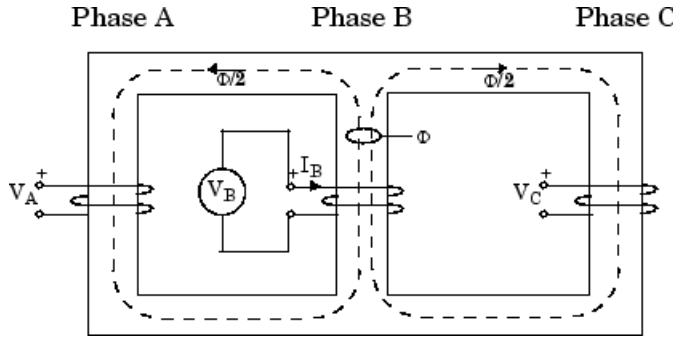
$$R1_i = \frac{V_{\text{nom}i}^2}{P1_i} \quad R0_i = \frac{V_{\text{nom}i}^2}{P0_i}.$$

The block takes into account the connection type you select, and the icon of the block is automatically updated. An input port labeled N is added to the block if you select the Y connection with accessible neutral for winding 1. If you ask for an accessible neutral on three-phase winding 2 or 3, an extra output port labeled $n2$ or $n3$ is generated.

Excitation Current in Zero Sequence

Often, the zero-sequence excitation current of a transformer with a 3-limb core is not provided by the manufacturer. In such a case, a reasonable value can be guessed as explained below.

The following figure shows a three-limb core with a single three-phase winding. Only phase B is excited and voltage is measured on phase A and phase C. The flux Φ produced by phase B shares equally between phase A and phase C so that $\Phi/2$ is flowing in limb A and in limb C. Therefore, in this particular case, if leakage inductance of winding B would be zero, voltage induced on phases A and C would be $-kV_B = -V_B/2$. In fact, because of the leakage inductance of the three windings, the average value of induced voltage ratio k when windings A, B, and C are successively excited must be slightly lower than 0.5



Assume:

- Z_s = average value of the three self-impedances
- Z_m = average value of mutual impedance between phases
- Z_1 = positive-sequence impedance of three-phase winding
- Z_0 = zero-sequence impedance of three-phase winding
- I_1 = positive-sequence excitation current
- I_0 = zero-sequence excitation current

$$\begin{aligned}
 V_B &= Z_s I_B \\
 V_A &= Z_m I_B = -V_B/2 \\
 V_C &= Z_m I_B = -V_B/2 \\
 Z_s &= \frac{2Z_1 + Z_0}{3} \\
 Z_m &= \frac{Z_0 - Z_1}{3} \\
 V_A = V_C &= \frac{Z_m}{Z_s} V_B = -\frac{\frac{Z_1}{Z_0} - 1}{2\frac{Z_1}{Z_0} + 1} V_B = -\frac{\frac{I_0}{I_1} - 1}{2\frac{I_0}{I_1} + 1} V_B = -kV_B,
 \end{aligned}$$

where k = ratio of induced voltage (with k slightly lower than 0.5)

Therefore, the I_0/I_1 ratio can be deduced from k :

$$\frac{I_0}{I_1} = \frac{1 + k}{1 - 2k}.$$

Obviously k cannot be exactly 0.5 because this would lead to an infinite zero-sequence current. Also, when the three windings are excited with a zero-sequence voltage the flux path returns through the air and tank surrounding the iron core. The high reluctance of the zero-sequence flux path results in a high zero-sequence current.

Let us assume $I_1 = 0.5\%$. A reasonable value for I_0 could be 100%. Therefore $I_0/I_1 = 200$. According to the equation for I_0/I_1 given above, one can deduce the value of k . $k = (200 - 1)/(2 * 200 + 1) = 199/401 = 0.496$.

Zero-sequence losses are also higher than the positive-sequence losses because of the additional eddy current losses in the tank.

Finally, the value of the zero-sequence excitation current and the value of the zero-sequence losses are not critical if the transformer has a winding connected in Delta because this winding acts as a short circuit for zero-sequence.

Winding Connections

The three-phase windings can be configured in the following manner:

- Y
- Y with accessible neutral
- Grounded Y
- Delta (D1), delta lagging Y by 30 degrees
- Delta (D11), delta leading Y by 30 degrees

Note The D1 and D11 notations refer to the following clock convention. It assumes that the reference Y voltage phasor is at noon (12) on a clock display. D1 and D11 refer respectively to 1 PM (delta voltages lagging Y voltages by 30 degrees) and 11 AM (delta voltages leading Y voltages by 30 degrees).

Parameters

Configuration Tab

Core type

Select the core geometry: Three single-phase cores or Three-limb or five-limb core (default). If you select the first option only the positive-sequence parameters are used to compute the inductance matrix. If you select the second option, both the positive- and zero-sequence parameters are used.

Winding 1 connection

The winding connection for three-phase winding 1. Choices are Y, Yn, Yg (default), Delta (D1), and Delta (D11).

Winding 2 connection

The winding connection for three-phase winding 2. Choices are Y, Yn, Yg (default), Delta (D1), and Delta (D11).

Winding 3 connection

The winding connection for three-phase winding 3. Choices are Y, Yn, Yg (default), Delta (D1), and Delta (D11).

Connect windings 1 and 2 in autotransformer

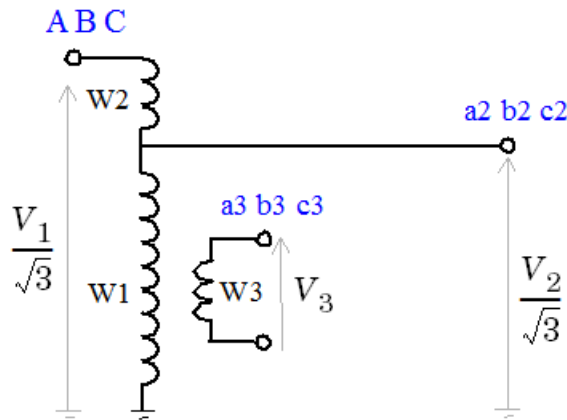
Select to connect the three-phase windings 1 and 2 in autotransformer (three-phase windings 1 and 2 in series with additive voltage). Default is cleared.

If the first voltage specified in the **Nominal line-line voltages** parameter is higher than the second voltage, the low voltage tap is connected on the right side (a2,b2,c2 terminals). Otherwise, the low voltage tap is connected on the left side (A,B,C terminals).

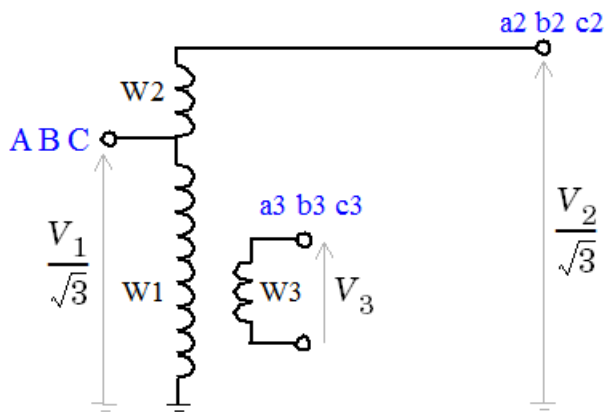
In autotransformer mode you must specify the same winding connections for the three-phase windings 1 and 2. If you select Yn connection for both winding 1 and winding 2, the common neutral N connector is displayed on the left side.

The following figure illustrates winding connections for one phase of an autotransformer when the three-phase windings are connected respectively in Yg,Yg, and Delta.

If $V_1 > V_2$:



If $V_2 > V_1$:



Windings W1,W2,W3 correspond to the following phase winding numbers:

- Phase A: W1=1, W2=4, W3=7
- Phase B: W1=2, W2=5, W3=8
- Phase C: W1=3, W2=6, W3=9

Measurements

Select **Winding voltages** to measure the voltage across the winding terminals of the Three-Phase Transformer block.

Select **Winding currents** to measure the current flowing through the windings of the Three-Phase Transformer block.

Select **All measurements** to measure the winding voltages and currents.

Default is None.

Place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurements** list box of the Multimeter block, the measurements are identified by a label followed by the block name.

If the **Winding 1 connection** parameter is set to Y, Yn, or Yg, the labels are as follows.

Measurement	Label
Winding 1 voltages	Uan_w1: or Uag_w1:
Winding 1 currents	Ian_w1: or Iag_w1:

If the **Winding 1 connection** parameter is set to Delta (D11) or Delta (D1), the labels are as follows.

Measurement	Label
Winding 1 voltages	Uab_w1:
Winding 1 currents	Iab_w1:

The same labels apply for three-phase windings 2 and 3, except that 1 is replaced by 2 or 3 in the labels.

Parameters Tab

Nominal power and frequency

The nominal power rating, in volt-amperes (VA), and nominal frequency, in hertz (Hz), of the transformer. Default is [260e6, 60].

Nominal line-line voltages [V1 V2 V3]

The phase-to-phase nominal voltages of windings 1, 2, 3 in volts RMS. Default is [315e3, 120e3, 43e3].

Winding resistances [R1 R2 R3]

The resistances in $\mu\Omega$ for windings 1, 2, and 3. Default is [0.005, 0.005, 0.005] .

Positive-sequence no-load excitation current

The no-load excitation current in percent of the nominal current when positive-sequence nominal voltage is applied at any three-phase winding terminals (ABC, abc2, or abc3). Default is 0.06.

Positive-sequence no-load losses

The core losses plus winding losses at no-load, in watts (W), when positive-sequence nominal voltage is applied at any three-phase winding terminals (ABC, abc2, or abc3). Default is $260e6 * 0.04 / 100$.

Positive-sequence short-circuit reactances

The positive-sequence short-circuit reactances X12, X23, and X13 in pu. Xij is the reactance measured from winding i when winding j is short-circuited. Default is [0.087, 0.166, 0.067].

When the **Connect windings 1 and 2 in autotransformer** parameter is selected, the short-circuit reactances are labeled XHL, XHT, and XLT. H, L, and T indicate the following terminals: H=high voltage winding (either winding 1 or winding 2), L=low voltage winding (either winding 1 or winding 2), and T=tertiary (winding 3).

Zero-sequence no-load excitation current with Delta windings opened

The no-load excitation current in percent of the nominal current when zero-sequence nominal voltage is applied at any three-phase winding terminals (ABC, abc2, or abc3) connected in Yg or Yn. Default is 100.

Note If your transformer contains delta-connected windings (D1 or D11), the zero-sequence current flowing into the Yg or Yn winding connected to the zero-sequence voltage source does not represent the net excitation current because zero-sequence currents are also flowing in the delta winding. Therefore, you must specify the no-load zero-sequence circulation current obtained with the delta windings open.

If you want to measure this excitation current, you must temporarily change the delta windings connections from D1 or D11 to Y, Yg, or Yn, and connect the excited winding in Yg or Yn to provide a return path for the source zero-sequence currents.

Zero-sequence no-load losses with Delta windings opened

The core losses plus winding losses at no-load, in watts (W), when zero-sequence nominal voltage is applied at any three-phase winding terminals (ABC, abc2, or abc3) connected in Yg or Yn. The Delta windings must be temporarily open to measure these losses. Default is $260e6 * 1 / 100$.

Note Note: If your transformer contains delta-connected windings (D1 or D11), the zero-sequence current flowing into the Yg or Yn winding connected to the zero-sequence voltage source does not represent the net excitation current because zero-sequence currents are also flowing in the delta winding. Therefore, you must specify the no-load zero-sequence circulation current obtained with the delta windings open.

Zero-sequence short-circuit reactances

The zero-sequence short-circuit reactances X12, X23, and X13 in pu. Xij is the reactance measured from winding i when winding j is short-circuited. If the **Zero-sequence X12 measured with winding 3 Delta connected** check box is not selected, X12 represents the short-circuit reactance when winding 3 is not connected in Delta. Default is [0.1, 0.2, 0.3].

When the **Connect windings 1 and 2 in autotransformer** check box is selected, the short-circuit reactances are labeled XHL, XHT, and XLT. H, L, and T indicate the following terminals: H=high voltage winding (either winding 1 or winding 2), L=low voltage winding (either winding 1 or winding 2), and T=tertiary (winding 3).

Zero-sequence X12 measured with winding 3 Delta connected

Select this check box if the available zero-sequence short circuit tests are obtained with tertiary winding (winding 3) connected in Delta. Default is cleared.

Limitations

This transformer model does not include saturation. If you need modeling saturation, connect the primary winding of a saturable Three-Phase Transformer (Two Windings) in parallel with the primary winding of your model. Use the same connection (Yg, D1 or D11) and same winding resistance for the two windings connected in parallel. Specify the Y or Yg connection for the secondary winding and leave it open. Specify appropriate voltage, power ratings, and the saturation characteristics that you want. The saturation characteristic is obtained when the transformer is excited by a positive-sequence voltage.

If you are modeling a transformer with three single-phase cores or a five-limb core, this model produces acceptable saturation currents because flux stays trapped inside the iron core.

For a three-limb core, the saturation model also gives acceptable results even if zero-sequence flux circulates outside of the core and returns through the air and the transformer tank surrounding the iron core. As the zero-sequence flux circulates in the air, the magnetic circuit is mainly linear and its reluctance is high (high magnetizing currents). These high zero-sequence currents (100% and more of nominal current) required to magnetize the air path are already taken into account in the linear model. Connecting a saturable transformer outside the three-limb linear model with a flux-current characteristic obtained in positive sequence produces currents required for magnetization of the iron core. This model gives acceptable results whether the three-limb transformer has a delta or not.

See Also

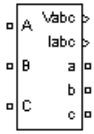
Linear Transformer, Multimeter, Three-Phase Transformer (Two Windings), Three-Phase Transformer (Three Windings), Three-Phase Transformer Inductance Matrix Type (Two Windings)

Version History

Introduced in R2008a

Three-Phase V-I Measurement

Measure three-phase currents and voltages in circuit



Libraries:

Simscape / Electrical / Specialized Power Systems / Sensors and Measurements

Description

The Three-Phase V-I Measurement block is used to measure instantaneous three-phase voltages and currents in a circuit. When connected in series with three-phase elements, it returns the three phase-to-ground or phase-to-phase peak voltages and currents.

The block can output the voltages and currents in per unit (pu) values or in volts and amperes.

If you choose to measure phase-to-ground voltages in per unit, the block converts the measured voltages based on peak value of nominal phase-to-ground voltage:

$$V_{abc}(pu) = \frac{V_{\text{phase to ground}}(V)}{V_{\text{base}}(V)}$$

where

$$V_{\text{base}} = \frac{V_{\text{nom}}(V_{\text{rms}})}{\sqrt{3}} \cdot \sqrt{2}$$

If you choose to measure phase-to-phase voltages in per unit, the block converts the measured voltages based on peak value of nominal phase-to-phase voltage:

$$V_{abc}(pu) = \frac{V_{\text{phase to phase}}(V)}{V_{\text{base}}(V)}$$

where

$$V_{\text{base}} = V_{\text{nom}}(V_{\text{rms}}) \cdot \sqrt{2}$$

If you choose to measure currents in per unit, the block converts the measured currents based on the peak value of the nominal current:

$$I_{abc}(pu) = \frac{I_{abc}(A)}{I_{\text{base}}(A)}$$

where

$$I_{\text{base}} = \frac{P_{\text{base}}}{V_{\text{nom}}} * \frac{\text{sqrt}(2)}{\text{sqrt}(3)}$$

V_{nom} and P_{base} are specified in the Three-Phase V-I Measurement block dialog box.

The steady-state voltage and current phasors measured by the Three-Phase V-I Measurement block can be obtained from the Powergui block by selecting **Steady-State Voltages and Currents**. The phasor magnitudes displayed in the Powergui stay in peak or RMS values even if the output signals are converted to pu.

Ports

Output

Vabc — Measured phase-to-ground or phase-to-phase voltages
Simulink signal

The three measured phase-to-ground or phase-to-phase voltages. The Vabc output disappears when the **Use a label** parameter is selected or when the **Voltage measurement** menu is set to no.

Iabc — Measured line currents
Simulink signal

The three measured line currents. The Iabc output disappears when the **Use a label** parameter is selected or when the **Current measurement** menu is set to no.

Conserving

A — Phase A
specialized electrical

Specialized electrical conserving port associated with the phase A voltage and current

B — Phase B
specialized electrical

Specialized electrical conserving port associated with the phase B voltage and current

C — Phase C
specialized electrical

Specialized electrical conserving port associated with the phase C voltage and current

a — Phase a
specialized electrical

Specialized electrical conserving port associated with the phase a voltage and current

b — Phase b
specialized electrical

Specialized electrical conserving port associated with the phase b voltage and current

c — Phase c
specialized electrical

Specialized electrical conserving port associated with the phase c voltage and current

Parameters

Voltage measurement — Voltage measurement option

no (default) | phase-to-ground | phase-to-phase

Select no if you do not want to measure three-phase voltage. Select phase-to-ground (default) if you want to measure the phase-to-ground voltages. Select phase-to-phase if you want to measure the phase-to-phase voltages.

Use a label — Label usage

off (default) | on

If selected, the voltage measurements are sent to a labeled signal. Use a From block to read the voltages. The Goto tag of the From block must correspond to the label specified by the **Signal label** parameter. If cleared, the voltage measurements are available via the Vabc output of the block. Default is cleared.

Signal label (use a From block to collect this signal) — Label tag

Vabc (default)

Specifies a label tag for the voltage measurements. default is Vabc.

Voltages in pu, based on peak value of nominal phase-to-ground voltage — Phase-to-ground voltages in pu

off (default) | on

If selected, the measured phase-to-ground voltages are converted in pu. Default is cleared.

Note For new models, this parameter is not available if the **Voltage measurement** parameter is set to phase-to-phase or no. However, this parameter is available and selected for a model saved from a previous release with the **Voltage measurement** parameter set to phase-to-phase and the **Voltages in pu** option selected. In such a case, you can choose to keep this parameter as is for compatibility purposes. You can also choose to clear this parameter and select the **Voltages in pu, based on peak value of nominal phase-to-phase voltage** parameter. You then need to take into account the impact of this new per unit conversion in your model

Voltages in pu, based on peak value of nominal phase-to-phase voltage — Phase-to-phase voltages in pu

off (default) | on

If selected, the measured phase-to-phase voltages are converted in pu. This parameter is not available if **Voltage measurement** parameter is set to phase-to-ground or no.

Current measurement — Current measurement option

no (default) | yes

Select yes (default) if you want to measure the three-phase currents that flow through the block.

Use a label — Label usage

off (default) | on

If selected, the current measurements are sent to a labeled signal. Use a From block to read the currents. The Goto tag of the From block must correspond to the label specified by the **Signal label**

parameter. If cleared, the current measurements are available via the `Iabc` output of the block. Default is cleared.

Signal label (use a From block to collect this signal) — Label tag
`Iabc` (default)

Specifies a label tag for the current measurements. Default is `Iabc`.

Currents in pu — Currents in pu
`off` (default) | `on`

If selected, the three-phase currents are measured in pu. Otherwise they are measured in amperes. Default is cleared.

Base power (VA 3 phase) — Three-phase base power
`100e6` (default)

The three-phase base power, in volt-ampere (VA), used to convert the measured currents in pu. The **Base power (VA 3 phase)** parameter is not available if **Currents in pu** is cleared.

Nominal voltage used for pu measurement (Vrms phase-phase) — Nominal voltage in RMS
`500e3` (default)

The nominal voltage, in volts RMS, used to convert the measured voltages in pu. This parameter is not available if the **Voltage measurement** parameter is set to `no`. Default is `500e3`.

Output signals in — Format of measured signals
`Complex` (default) | `Real-Imag` | `Magnitude-Angle` | `Magnitude`

Specifies the format of the measured signals when the block is used in a phasor simulation. The **Output signals in** parameter is enabled when the **Simulation type** parameter of the `powergui` block is set to `Phasor` or `Discrete phasor`.

Set to `Complex` (default) to output the measured voltages and currents as complex values. The outputs are complex signals.

Set to `Real-Imag` to output the real and imaginary parts of the measured voltages and currents.

Set to `Magnitude-Angle` to output the magnitudes and angles of the measured voltages and currents.

Set to `Magnitude` to output the magnitudes of the measured voltages and currents. The output is a scalar value.

Version History

Introduced before R2006a

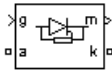
Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

Thyristor

Implement thyristor model

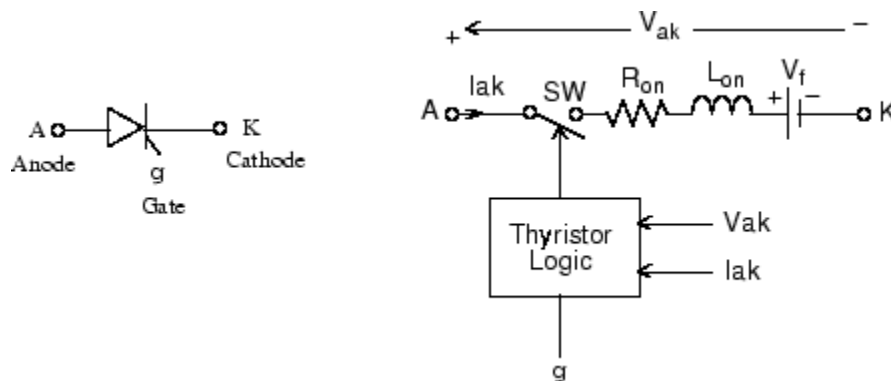


Libraries:

Simscape / Electrical / Specialized Power Systems / Power Electronics

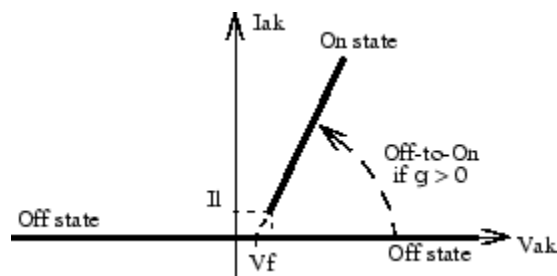
Description

The thyristor is a semiconductor device that can be turned on via a gate signal. The thyristor model is simulated as a resistor R_{on} , an inductor L_{on} , and a DC voltage source representing the forward voltage V_f , connected in series with a switch. The switch is controlled by a logical signal depending on the voltage V_{ak} , the current I_{ak} , and the gate signal g .



The Thyristor block also contains a series Rs-Cs snubber circuit that can be connected in parallel with the thyristor device.

The static VI characteristic of this model is shown below.



The thyristor device turns on when the anode-cathode V_{ak} voltage is greater than V_f and a positive pulse signal is applied at the gate input ($g > 0$). The pulse height must be greater than 0 and last long enough to allow the thyristor anode current to become larger than the latching current I_l .

The thyristor device turns off when the current flowing in the device becomes 0 ($I_{ak} = 0$) and a negative voltage appears across the anode and cathode for at least a period of time equal to the turnoff time T_q . If the voltage across the device becomes positive within a period of time less than T_q , the device turns on automatically even if the gate signal is low ($g = 0$) and the anode current is less

than the latching current. Furthermore, if during turn-on, the device current amplitude stays below the latching current level specified in the dialog box, the device turns off after the gate signal level becomes low ($g = 0$).

The turnoff time T_q represents the carrier recovery time: it is the time interval between the instant the anode current has decreased to 0 and the instant when the thyristor is capable of withstanding positive voltage V_{ak} without turning on again.

Examples

In the `power_thyristor` example, a single-pulse thyristor rectifier is used to feed an RL load. The gate pulses are obtained from a pulse generator synchronized on the source voltage. The following parameters are used:

R		1 Ω
L		10 mH
Thyristor block:	Ron	0.001 Ω
	Lon	0 H
	Vf	0.8 V
	Rs	20 Ω
	Cs	4e-6 F

The firing angle is varied by a pulse generator synchronized on the voltage source. Run the simulation and observe the load current and load voltage, as well as the thyristor current and voltage.

Assumptions and Limitations

- The Thyristor block implements a macro model of the real thyristor. It does not take into account either the geometry of the device or complex physical processes that model the behavior of the device [1, 2]. The forward breakover voltage and the critical value of the derivative of the reapplied anode-cathode voltage are not considered by the model.
- Depending on the value of the inductance L_{on} , the Thyristor block is modeled either as a current source ($L_{on} > 0$) or as a variable topology circuit ($L_{on} = 0$). The Thyristor block cannot be connected in series with an inductor, a current source, or an open circuit, unless its snubber circuit is in use.
- The inductance L_{on} is forced to 0 if you choose to discretize your circuit.

Ports

Input

g — Thyristor control signal
Simulink signal

Simulink signal to control the opening and closing of the thyristor.

Output

m — Switch output
Simulink signal

The Simulink output of the block is a vector containing two signals. You can demultiplex these signals by using the Bus Selector block provided in the Simulink library:

Signal	Definition	Units
1	Thyristor current	A
2	Thyristor voltage	V

Dependencies

To enable this port, select the **Show measurement port** parameter.

Conserving

A — Thyristor anode
specialized electrical

Specialized electrical conserving port associated with the thyristor anode.

k — Thyristor cathode
specialized electrical

Specialized electrical conserving port associated with the thyristor cathode.

Parameters

To optimize simulation speed, two models of thyristors are available: the thyristor model and the detailed thyristor model. For the thyristor model, the latching current I_l and recovery time T_q are assumed to be 0.

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Resistance Ron — Internal resistance of thyristor
0.001 (default) | nonnegative scalar

Internal resistance of the thyristor, in ohms (Ω). The **Resistance Ron** parameter cannot be set to 0 when the **Inductance Lon** parameter is set to 0..

Inductance Lon — Internal inductance of thyristor
0 (default) | 1e-3 | nonnegative scalar

Internal inductance L_{on} , in henries (H). Default is 0 for Thyristor blocks and 1e-3 for Detailed Thyristor blocks. The **Inductance Lon** parameter is normally set to 0 except when the **Resistance Ron** parameter is set to 0.

Forward voltage Vf — Forward voltage of thyristor
0.8 (default) | nonnegative scalar

Forward voltage of the thyristor device, in volts.

Initial current Ic — Initial current flowing in thyristor
0 (default) | nonnegative scalar

Initial current flowing in the thyristor. It is usually set to 0 to start the simulation with the device blocked.

You can specify an **Initial current Ic** value corresponding to a particular state of the circuit. In such a case all states of the linear circuit must be set accordingly. Initializing all states of a power electronic converter is a complex task. Therefore, this option is useful only with simple circuits.

Dependencies

To enable this parameter, set **Inductance Lon** to a value greater than 0.

Snubber resistance Rs — Snubber resistance

500 (default) | nonnegative scalar

Snubber resistance, in ohms (Ω). Set the **Snubber resistance Rs** parameter to `inf` to eliminate the snubber from the model.

Snubber capacitance Cs — Snubber capacitance

250e-9 (default) | nonnegative scalar

Snubber capacitance in farads (F). Set the **Snubber capacitance Cs** parameter to 0 to eliminate the snubber, or to `inf` to get a resistive snubber.

Show measurement port — Option to show measurement port

on (default) | off

If selected, add a Simulink output to the block returning the thyristor current and voltage.

Latching current Il — Latching current

0.1 (default) | nonnegative scalar

The latching current of the detailed thyristor model, in amperes (A). Default is 0.1. This parameter is specific to Detailed Thyristor blocks.

Turn-off time Tq — Snubber capacitance

100e-6 (default) | positive scalar

The turn-off time Tq of the detailed thyristor model, in seconds (s). Default is 100e-6. This parameter is specific to Detailed Thyristor blocks.

Version History

Introduced before R2006a

References

- [1] Rajagopalan, V., *Computer-Aided Analysis of Power Electronic Systems*, Marcel Dekker, Inc., New York, 1987.
- [2] Mohan, N., T.M. Undeland, and W.P. Robbins, *Power Electronics: Converters, Applications, and Design*, John Wiley & Sons, Inc., New York, 1995.

Extended Capabilities

C/C++ Code Generation

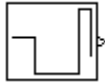
Generate C and C++ code using Simulink® Coder™.

See Also

Diode | Universal Bridge

Timer

Generate signal changing at specified transition times



Library

powerlib_extras/Control Blocks, powerlib_extras/Discrete Control Blocks

Note The Pulse & Signal Generators section of the Control and Measurements library contains the Stair Generator block. This is an improved version of the Timer block. The new block features a mechanism that eliminates duplicate continuous and discrete versions of the same block by basing the block configuration on the simulation mode. If your legacy models contain the Timer block, they continue to work. However, for best performance, use the Stair Generator block in your new models.

Description

The Timer block generates a signal changing at specified transition times. Use this block to generate a logical signal (0 or 1 amplitude) and control the opening and closing times of power switches like the Breaker block and the Ideal Switch block. You can also use this block to generate a signal whose amplitude changes by steps at specified transition times.

Parameters

Time(s)

The transition times, in seconds, when the output of the block changes its value as defined by the **Amplitude** parameter. The **Time(s)** parameter must be a vector of the same length as the vector defined in the **Amplitude** parameter. The definition of the time 0 is optional. If a signal is not specified at time 0, the output is kept at zero until the first transition time specified in the **Amplitude** vector.

Amplitude

The vector of amplitudes of signal to be generated by the Timer block. The amplitude is kept constant between transition times defined in the **Time(s)** vector.

Inputs and Outputs

The output is a signal changing by steps at specified transition times.

Version History

Introduced before R2006a

See Also

Stair Generator

Total Harmonic Distortion

Measure total harmonic distortion (THD) of signal



Library

powerlib_extras/Measurements

A discrete version of this block is available in the powerlib_extras/Discrete Measurements library.

Note The Sensors and Measurements library contains the THD block. This is an improved version of the Total Harmonic Distortion block. The new block features a mechanism that eliminates duplicate continuous and discrete versions of the same block by basing the block configuration on the simulation mode. If your legacy models contain the Total Harmonic Distortion block, they continue to work. However, for best performance, use the THD block in your new models.

Description

The Total Harmonic Distortion block measures the total harmonic distortion (THD) of a periodic distorted signal. The signal can be a measured voltage or current.

The THD is defined as the root mean square (RMS) value of the total harmonics of the signal, divided by the RMS value of its fundamental signal. For example, for currents, the THD is defined as

$$\text{total harmonic distortion (THD)} = I_H/I_F,$$

where

$$I_H = \sqrt{I_2^2 + I_3^2 + \dots + I_n^2},$$

where

I_n = RMS value of the harmonic n

I_F = RMS value of the fundamental current.

The THD has a null value for a pure sinusoidal voltage or current.

Parameters

Fundamental frequency

The frequency, in hertz, of the fundamental signal.

Inputs and Outputs

signal

Connect this input to the voltage or current you want to measure the total harmonic distortion.

THD

The output returns the THD of the input signal.

Version History

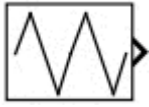
Introduced before R2006a

See Also

THD

Triangle Generator

Generate symmetrical triangle waveform at regular intervals

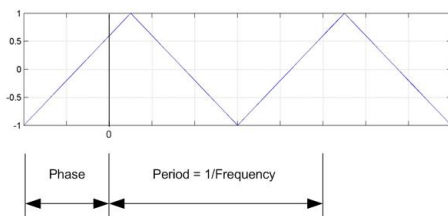


Libraries:

Simscape / Electrical / Specialized Power Systems / Sources / Signal Generator Sources

Description

The Triangle Generator block generates a symmetrical triangle waveform, with peak amplitude of ± 1 . The figure shows how the **Frequency** and **Phase** block parameters affect the output waveform.



Characteristics

Sample Time	Specified in the Sample Time parameter Continuous if Sample Time = 0
Scalar Expansion	No
Dimensionalized	No
Zero-Crossing Detection	No

Examples

The power_SignalGenerators example shows various uses of the Triangle Generator block.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Frequency (Hz) — Frequency of triangle waveform

1e3 (default) | scalar

Frequency of the triangle waveform, in hertz.

Phase (degrees) — Delay of triangle waveform

90 (default) | scalar

Delay of the triangle waveform, in degrees. When the phase is set to 0, the waveform starts with an amplitude of -1 and a positive slope.

Sample time — Block sample time

0 (default) | nonnegative scalar

Sample time of the block, in seconds. Set to 0 to implement a continuous block.

Version History

Introduced in R2013a

Extended Capabilities

C/C++ Code Generation

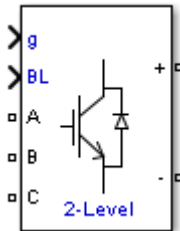
Generate C and C++ code using Simulink® Coder™.

See Also

Pulse Generator (Thyristor) | PWM Generator

Two-Level Converter

Implement three-phase two-level power converter



Libraries:

Simscape / Electrical / Specialized Power Systems / Power Electronics

Description

The Two-Level Converter block implements a three-phase, two-level power converter. You can choose from four model types:

- **Switching devices** — The converter is modeled with IGBT/diode pairs controlled by firing pulses produced by a PWM generator. This model provides the most accurate simulation results.
- **Switching function** — The converter is modeled by a switching-function model. The switches are replaced with two voltage sources and two diodes on the AC side and with two current sources on the DC side.

The converter is controlled by firing pulses produced by a PWM generator (0/1 signals) or by firing pulses averaged over a specified period (PWM averaging: signals between 0 and 1). Both modes of operation produce harmonics normally generated by a PWM-controlled converter and also correctly simulate rectifying operation as well as blanking time. This model type is well-suited for real-time simulation.

- **Average model (Uref-controlled)** — The converter is modeled using a switching-function model directly controlled by the reference voltage signals (Uref). A PWM generator is not required.
- **Average model (Uref-controlled, no rectifier mode)** — The block uses the voltage source directly controlled by the reference voltage to model the converter. The model does not require a PWM generator and does not simulate the rectifier mode.

When doing real-time simulation, use the **Switching function** setting with the firing pulses averaged, or the **Average model (Uref-controlled)** or **Average model (Uref-controlled, no rectifier mode)** settings.

Ports

Input

g — Gating signal to control converter
vector of six elements

Vectorized gating signal to control the converter. The gating signal contains six firing pulses. The first two pulses control the Q1 and Q2 switching devices (phase A of the converter), pulses three and four

control the Q3 and Q4 switching devices (phase B of the converter), and the last two pulses control the Q5 and Q6 switching devices (phase C of the converter).

Dependencies

To enable this port, set **Model type** to `Switching devices` or `Switching function`.

Uref — Signal to control converter
vector of three elements

Vectorized signal to control the converter. The signal contains three reference voltages (one for each phase).

Dependencies

To enable this port, set **Model type** to `Average model (Uref-controlled)` or `Average model (No rectifier mode)`.

BL — Input to block firing pulses
scalar

Input port to block all firing pulses to the converter. To block all firing pulses to the converter, apply a signal value of 1 to this port.

Conserving

+ — Positive terminal
specialized electrical

Specialized electrical conserving port associated with the positive terminal.

- — Negative terminal
specialized electrical

Specialized electrical conserving port associated with the negative terminal.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Model type — Model type
`Switching devices (default)` | `Switching function` | `Average model (Uref-controlled)` | `Average model (Uref-controlled, no rectifier mode)`

Select one of these values:

- **Switching devices** — The converter is modeled with IGBT/diode pairs controlled by firing pulses produced by a PWM generator.
- **Switching function** — The converter is modeled by a switching-function model. The switches are replaced with two voltage sources and two diodes on the AC side and with two current sources on the DC side.
- **Average model (Uref-controlled)** — The converter is modeled using a switching-function model directly controlled by the reference voltage signals (Uref).

- **Average model (Uref-controlled, no rectifier mode)** — The block uses the voltage source directly controlled by the reference voltage to model the converter. The model does not require a PWM generator and does not simulate the rectifier mode.

Device on-state resistance (Ohms) — Internal resistance of switching devices

1e-3 (default) | positive scalar

Internal resistance of the switching devices, in ohms.

Dependencies

To enable this parameter, set **Model type** to Switching devices.

Snubber resistance (Ohms) — Snubber resistance

1e6 (default) | positive scalar

Snubber resistance, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to Switching devices.

Snubber capacitance (F) — Snubber capacitance

inf (default) | nonnegative scalar

Snubber capacitance, in farads. Set the snubber capacitance to 0 to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to Switching devices.

Diode on-state resistance (Ohms) — Internal resistance of diodes

1e-3 (default) | positive scalar

Internal resistance of the diodes, in ohms.

Dependencies

To enable this parameter, set **Model type** to Switching function or Average model (Uref-controlled).

Diode snubber resistance (Ohms) — Snubber resistance of diodes

1e6 (default) | positive scalar

Snubber resistance of the diodes, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to Switching function or Average model (Uref-controlled).

Diode snubber capacitance (F) — Snubber capacitance of diodes

inf (default) | nonnegative scalar

Snubber capacitance of the diodes, in farads. Set the snubber capacitance to 0 to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to `Switching function` or `Average model (Uref-controlled)`.

Diode forward voltage (V) — Forward voltage across diode
`1e-3` (default) | `nonnegative scalar`

Forward voltage, in volts, across the diode when it is conducting.

Dependencies

To enable this parameter, set **Model type** to `Switching function` or `Average model (Uref-controlled)`.

Current source snubber resistance (Ohms) — Snubber resistance across current sources
`inf` (default) | `positive scalar`

Snubber resistance across the two current sources, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to `Switching function` or `Average model (Uref-controlled)`.

Version History

Introduced in R2015b

Extended Capabilities

C/C++ Code Generation

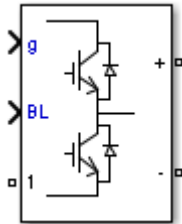
Generate C and C++ code using Simulink® Coder™.

See Also

Three-Level NPC Converter | Boost Converter

Two-Quadrant DC/DC Converter

Implement two-quadrant DC/DC power converter



Libraries:

Simscape / Electrical / Specialized Power Systems / Power Electronics

Description

The Two-Quadrant DC/DC Converter block implements a two-quadrant DC/DC power converter. You can choose from four model types :

- **Switching devices** — The converter is modeled with IGBT/diode pairs controlled by firing pulses produced by a PWM generator. This model provides the most accurate simulation results.
- **Switching function** — The converter is modeled by a switching-function model. The switches are replaced with two voltage sources and two diodes on the AC side and with two current sources on the DC side.

The converter is controlled by firing pulses produced by a PWM generator (0/1 signals) or by firing pulses averaged over a specified period (PWM averaging: signals between 0 and 1). Both modes of operation produce harmonics normally generated by a PWM-controlled converter and also correctly simulate rectifying operation as well as blanking time. This model type is well-suited for real-time simulation.

- **Average model (D-controlled)** — The converter is modeled using a switching-function model directly controlled by the duty cycle signal ($0 < D < 1$). A PWM generator is not required.
- **Average model (No rectifier mode)** — The block uses the voltage source directly controlled by the reference voltage to model the converter. The model does not require a PWM generator and does not simulate the rectifier mode. This setting provides the fastest simulations.

When doing real-time simulation, use the **Switching function** setting with the firing pulses averaged, or the **Average model (D-controlled)** or **Average model (No rectifier mode)** settings.

Ports

Input

g — Gating signal to control converter
vector of two elements

Vectorized gating signal to control the converter. The gating signal contains two firing pulses to control Q1 and Q2.

Dependencies

To enable this port, set **Model type** to `Switching devices` or `Switching function`.

D — Duty cycle signal to control converter
scalar in the range [0,1]

Duty cycle signal to control the converter.

Dependencies

To enable this port, set **Model type** to `Average model (D-controlled)` or `Average model (No rectifier mode)`.

BL — Input to block firing pulses
scalar

Input port to block all firing pulses to the converter. To block all firing pulses to the converter, apply a signal value of 1 to this port.

Conserving

+ — Positive terminal
specialized electrical

Specialized electrical conserving port associated with the positive terminal.

- — Negative terminal
specialized electrical

Specialized electrical conserving port associated with the negative terminal.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Model type — Model type

`Switching devices (default)` | `Switching function` | `Average model (D-controlled)` | `Average model (No rectifier mode)`

Select one of these values:

- **Switching devices** — The converter is modeled with IGBT/diode pairs controlled by firing pulses produced by a PWM generator.
- **Switching function** — The converter is modeled by a switching-function model. The switches are replaced with two voltage sources and two diodes on the AC side and with two current sources on the DC side.
- **Average model (D-controlled)** — The converter is modeled using a switching-function model directly controlled by the duty cycle signal ($0 < D < 1$).
- **Average model (No rectifier mode)** — The block uses the voltage source directly controlled by the reference voltage to model the converter. The model does not require a PWM generator and does not simulate the rectifier mode.

Device on-state resistance (Ohms) — Internal resistance of switching devices

1e-3 (default) | positive scalar

Internal resistance of the switching devices, in ohms.

Dependencies

To enable this parameter, set **Model type** to Switching devices.

Snubber resistance (Ohms) — Snubber resistance

1e6 (default) | positive scalar

Snubber resistance, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to Switching devices.

Snubber capacitance (F) — Snubber capacitance

`inf` (default) | nonnegative scalar

Snubber capacitance, in farads. Set the snubber capacitance to `0` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to Switching devices.

Diode on-state resistance (Ohms) — Internal resistance of diodes

1e-3 (default) | positive scalar

Internal resistance of the diodes, in ohms.

Dependencies

To enable this parameter, set **Model type** to Switching function or Average model (D-controlled).

Diode snubber resistance (Ohms) — Snubber resistance of diodes

1e6 (default) | positive scalar

Snubber resistance of the diodes, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to Switching function or Average model (D-controlled).

Diode snubber capacitance (F) — Snubber capacitance of diodes

`inf` (default) | nonnegative scalar

Snubber capacitance of the diodes, in farads. Set the snubber capacitance to `0` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to Switching function or Average model (D-controlled).

Diode forward voltage (V) — Forward voltage across diode

1e-3 (default) | nonnegative scalar

Forward voltage, in volts, across the diode when it is conducting.

Dependencies

To enable this parameter, set **Model type** to Switching function or Average model (D-controlled).

DC side snubber resistance (Ohms) — Snubber resistance across current sources

inf (default) | positive scalar

Snubber resistance across the two current sources, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to Switching function, Average model (D-controlled), or Average model (No rectifier mode).

AC side source resistance (Ohms) — Source resistance on AC side

1e-3 (default) | positive scalar

Source resistance on the AC side, in ohms. Set the snubber resistance to `inf` to eliminate the snubbers.

Dependencies

To enable this parameter, set **Model type** to Average model (No rectifier mode).

Version History

Introduced in R2015b

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

Full-Bridge Converter | Half-Bridge Converter

Universal Bridge

Implement universal power converter with selectable topologies and power electronic devices



Libraries:

Simscape / Electrical / Specialized Power Systems / Power Electronics



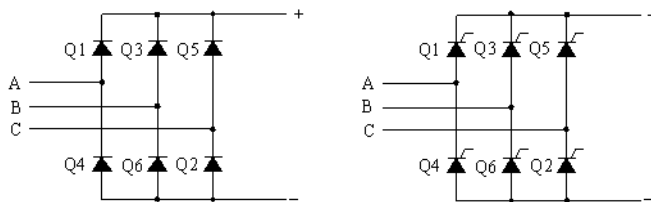
Description

The Universal Bridge block implements a universal three-phase power converter that consists of up to six power switches connected in a bridge configuration. The type of power switch and converter configuration are selectable from the dialog box.

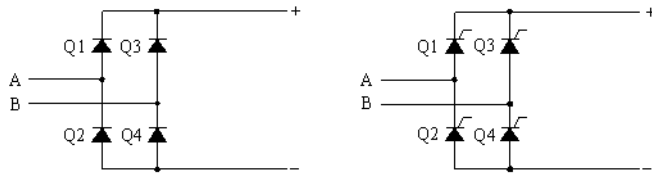
The Universal Bridge block allows simulation of converters using both naturally commutated (or line-commutated) power electronic devices (diodes or thyristors) and forced-commutated devices (GTO, IGBT, MOSFET).

The Universal Bridge block is the basic block for building two-level voltage-sourced converters (VSC).

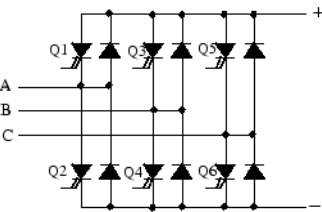
The device numbering is different if the power electronic devices are naturally commutated or forced-commutated. For a naturally commutated three-phase converter (diode and thyristor), numbering follows the natural order of commutation:



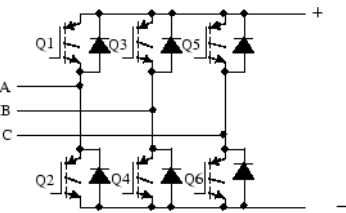
For the case of a two-phase diode or thyristor bridge, and for any other bridge configuration, the order of commutation is the following:



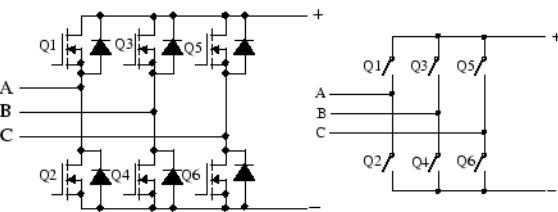
GTO-Diode bridge:



IGBT-Diode bridge:



MOSFET-Diode and Ideal Switch bridges:



Ports

Input

g — Gate input
Simulink signal

The gate input for the controlled switch devices. The pulse ordering in the vector of the gate signals corresponds to the switch number indicated in the six circuits shown in the Description section. For the diode and thyristor bridges, the pulse ordering corresponds to the natural order of commutation. For all other forced-commutated switches, pulses are sent to upper and lower switches of phases A, B, and C.

Topology	Pulse Vector of Input g
one arm	[Q1,Q2]

Topology	Pulse Vector of Input g
two arms	[Q1,Q2,Q3,Q4]
three arms	[Q1,Q2,Q3,Q4,Q5,Q6]

Conserving

A — Phase A

specialized electrical

Specialized electrical conserving port associated with phase A.

B — Phase B

specialized electrical

Specialized electrical conserving port associated with phase B.

C — Phase C

specialized electrical

Specialized electrical conserving port associated with phase C.

+ — Positive terminal

specialized electrical

Specialized electrical conserving port associated with the positive terminal.

- — Negative terminal

specialized electrical

Specialized electrical conserving port associated with the negative terminal.

Parameters

Number of bridge arms — Number of bridge arms

1 (default) | 2 | 3

Set to 1 or 2 to get a single-phase converter (two or four switching devices). Set to 3 to get a three-phase converter connected in Graetz bridge configuration (six switching devices). Default is 3.

Snubber resistance Rs (Ohms) — Snubber resistance

1e5 (default)

The snubber resistance, in ohms (Ω). Default is 1e5. Set the **Snubber resistance Rs** parameter to `inf` to eliminate the snubbers from the model.

Snubber capacitance Cs (F) — Snubber capacitance

`inf` (default)

The snubber capacitance, in farads (F). Default is `inf`. Set the **Snubber capacitance Cs** parameter to 0 to eliminate the snubbers, or to `inf` to get a resistive snubber.

When you are using the continuous solver you can eliminate snubbers in all power electronic devices if you select the **Disable snubbers in switching devices** option in the Preference tab of the Powergui block

When your system is discretized, you can simulate power electronic devices with virtually no snubbers by specifying purely resistive snubbers with a very large resistance, thus producing negligible leakage currents. The bridge operates satisfactorily with purely resistive snubbers.

Power Electronic device — Type of power electronic device

Diodes (default) | Thyristors | GTO / Diodes | MOSFET / Diodes | IGBT / Diodes | Ideal Switches | Switching-function based VSC | Average-model based VSC

Select the type of power electronic device to use in the bridge. Default is **Thyristors**.

When you select **Switching-function based VSC**, a switching-function voltage source converter type equivalent model is used, where switches are replaced by two voltage sources on the AC side and a current source on the DC side. This model uses the same firing pulses as for other power electronic devices and it correctly represents harmonics normally generated by the bridge. When the firing pulses are set to 0, the AC side voltage sources of 0 volts and the DC side current source of 0 A are implemented.

When you select **Average-model based VSC**, an average-model type of voltage source converter is used to represent the power-electronic switches. Unlike the other power electronic devices, this model uses the reference signals (uref) representing the average voltages generated at the ABC terminals of the bridge. This model does not represent harmonics. It can be used with larger sample times while preserving the average voltage dynamics.

See `power_sfavg` for an example comparing these two models to an Universal Bridge block using IGBT/Diode device.

Ron (Ohms) — Internal resistance

1e-3 (default)

Internal resistance of the selected device, in ohms (Ω). Default is 1e-3.

Lon (H) — Internal inductance

0 (default)

Internal inductance, in henries (H), for the diode or the thyristor device. Default is 0. When the bridge is discretized, the Lon parameter must be set to zero.

Forward voltages [Device Vf(V) , Diode Vfd(V)] — Forward voltages

[0 0] (default)

This parameter is available when the selected **Power electronic device** is GTO/Diodes or IGBT/Diodes.

Forward voltages, in volts (V), of the forced-commutated devices (GTO, MOSFET, or IGBT) and of the antiparallel diodes. Default is [0 0].

Forward voltage Vf (V) — Forward voltage across the device

0 (default)

This parameter is available only when the selected **Power electronic device** is Diodes or Thyristors.

Forward voltage, in volts (V), across the device when it is conducting. Default is 0.

Measurements — Measurement type

None (default) | Device voltages | Device currents | UAB UBC UCA UDC voltages | All voltages and currents

Default is None.

Select **Device voltages** to measure the voltages across the six power electronic device terminals.

Select **Device currents** to measure the currents flowing through the six power electronic devices. If antiparallel diodes are used, the measured current is the total current in the forced-commutated device (GTO, MOSFET, or IGBT) and in the antiparallel diode. A positive current therefore indicates a current flowing in the forced-commutated device and a negative current indicates a current flowing in the diode. If snubber devices are defined, the measured currents are the ones flowing through the power electronic devices only.

Select **UAB UBC UCA UDC voltages** to measure the terminal voltages (AC and DC) of the Universal Bridge block.

Select **All voltages and currents** to measure all voltages and currents defined for the Universal Bridge block.

Place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurements** menu of the Multimeter block, the measurement is identified by a label followed by the block name.

Measurement	Label
Device voltages	Usw1 :
Branch current	Isw1 :
Terminal voltages	Uab :

Version History

Introduced before R2006a

Extended Capabilities

C/C++ Code Generation

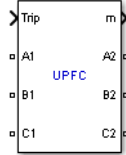
Generate C and C++ code using Simulink® Coder™.

See Also

Diode | GTO | Ideal Switch | IGBT | Mosfet | Multimeter | Three-Level Bridge | Thyristor

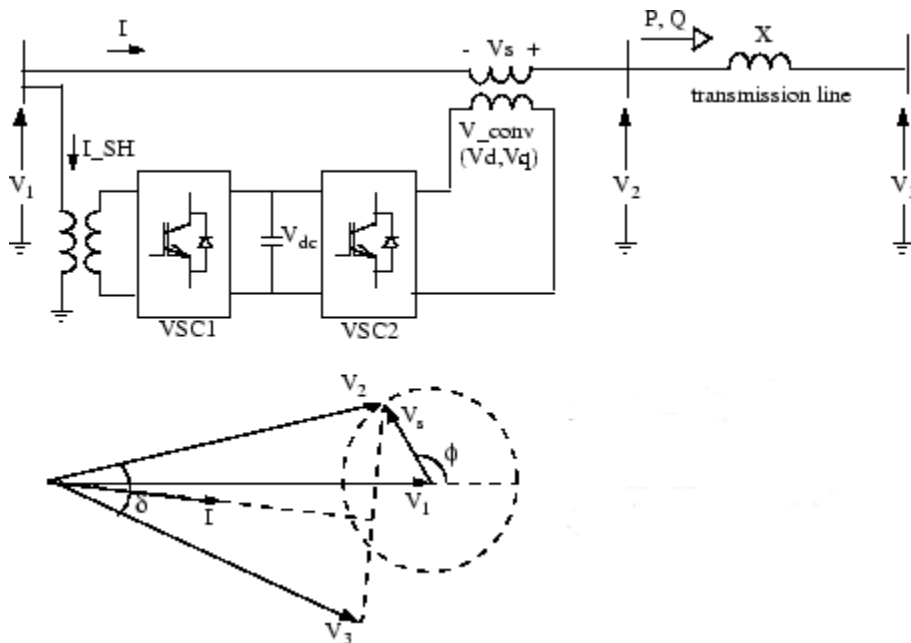
Unified Power Flow Controller (Phasor Type)

Implement phasor model of three-phase unified power flow controller



Description

The Unified Power Flow Controller (UPFC) is the most versatile member of the Flexible AC Transmission Systems (FACTS) family using power electronics to control power flow on power grids [1]. The UPFC uses a combination of a shunt controller (STATCOM) and a series controller (SSSC) interconnected through a common DC bus as shown on the figure below.

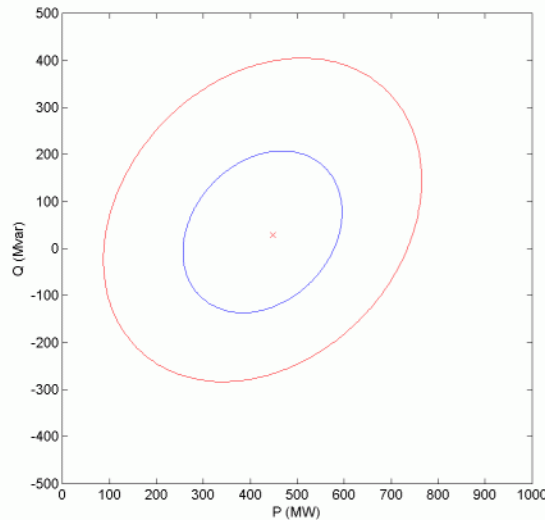


Single-line Diagram of a UPFC and Phasor Diagram of Voltages and Currents

$$P = \frac{V_2 V_3 \sin \delta}{X}, Q = \frac{V_2 (V_2 - V_3 \cos \delta)}{X}$$

This FACTS topology provides much more flexibility than the SSSC for controlling the line active and reactive power because active power can now be transferred from the shunt converter to the series converter, through the DC bus. Contrary to the SSSC where the injected voltage V_s is constrained to stay in quadrature with line current I , the injected voltage V_s can now have any angle with respect to line current. If the magnitude of injected voltage V_s is kept constant and if its phase angle ϕ with respect to V_1 is varied from 0 degrees to 360 degrees, the locus described by the end of vector V_2 ($V_2 = V_1 + V_s$) is a circle as shown on the phasor diagram. As ϕ is varying, the phase shift δ between

voltages V2 and V3 at the two line ends also varies. It follows that both the active power P and the reactive power Q transmitted at one line end can be controlled. The UPFC controllable region in the P-Q plane is the area enclosed by an ellipse, as shown on the figure below.



Controllable Region for a 100 MVA UPFC Connected on 500 kV, 200-km line

This figure was obtained with a 100 MVA UPFC controlling active and reactive power at one end of a 500 kV, 200-km transmission line. The following parameters have been used:

- Line: length = 200 km; reactance = $0.35 \Omega / \text{km}$
- System voltage: 500-kV infinite sources V1 and V3; V1=1.0 pu, 0 degrees; V3= 1.0 pu, -7.22 degrees
- Series and shunt converter rating: 100 MVA
- Series converter: nominal injected voltage = 10% of nominal line-to-ground voltage (28.9 kV); impedance (transformer leakage reactance and filters) = 0.15 pu

With V3 lagging V1 by 7.22 degrees, the natural power flow without compensation is 450 MW or 50% of the line surge impedance loading (SIL=900 MW). With an injected voltage $V_s = 0.1$ pu any operating point inside the larger ellipse can be obtained and active power can be varied by approximately ± 300 MW.

In addition to allow control of the line active and reactive power, the UPFC provides an additional degree of freedom. Its shunt converter operating as a STATCOM controls voltage V1 by absorbing or generating reactive power.

Both the series and shunt converters use a Voltage-Sourced Converter (VSC) connected on the secondary side of a coupling transformer. The VSCs use forced-commutated power electronic devices (GTOs, IGBTs, or IGCTs) to synthesize a voltage from a DC voltage source. The common capacitor connected on the DC side of the VSCs acts as a DC voltage source. Two VSC technologies can be used for the VSCs:

- VSC using GTO-based square-wave inverters and special interconnection transformers. Typically four three-level inverters are used to build a 48-step voltage waveform. Special interconnection

transformers are used to neutralize harmonics contained in the square waves generated by individual inverters. In this type of VSC, the fundamental component of voltage is proportional to the voltage V_{dc} . Therefore V_{dc} has to vary for controlling the injected voltage.

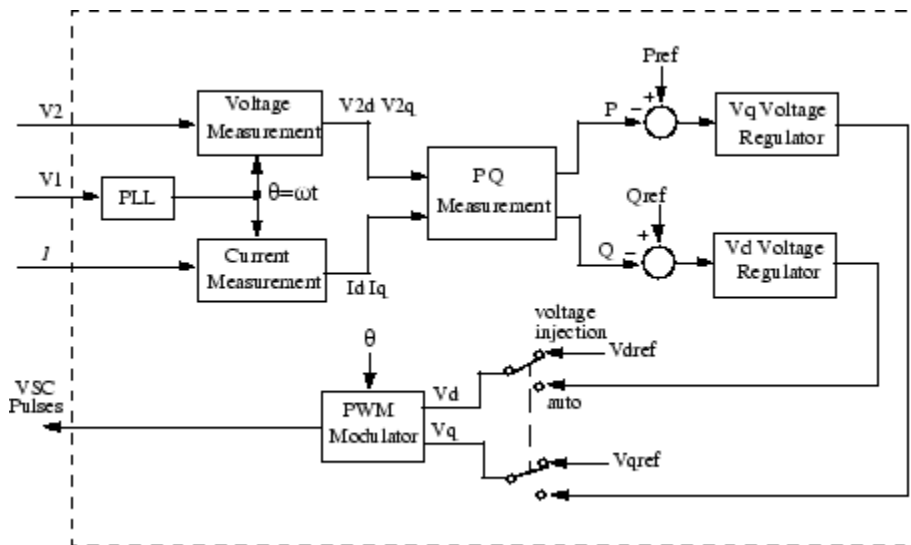
- VSC using IGBT-based PWM inverters. This type of inverter uses the Pulse-Width Modulation (PWM) technique to synthesize a sinusoidal waveform from a DC voltage with a typical chopping frequency of a few kilohertz. Harmonics are cancelled by connecting filters at the AC side of the VSC. This type of VSC uses a fixed DC voltage V_{dc} . Voltage is varied by changing the modulation index of the PWM modulator.

The UPFC (Phasor Type) block models an IGBT-based UPFC. However, as details of the inverter and harmonics are not represented, it can be also used to model a GTO-based UPFC in transient stability studies.

Control System

The shunt converter operates as a STATCOM. For a description of its control system, refer to the Static Synchronous Compensator (Phasor Type). In summary, the shunt converter controls the AC voltage at its terminals and the voltage of the DC bus. It uses a dual voltage regulation loop: an inner current control loop and an outer loop regulating AC and DC voltages.

Control of the series branch is different from the SSSC. In an SSSC the two degrees of freedom of the series converter are used to control the DC voltage and the reactive power. In case of a UPFC the two degrees of freedom are used to control the active power and the reactive power. A simplified block diagram of the series converter is shown below.



Simplified Block Diagram of the Series Converter Control System

The series converter can operate either in power flow control (automatic mode) or in manual voltage injection mode. In power control mode, the measured active power and reactive power are compared with reference values to produce P and Q errors. The P error and the Q error are used by two PI regulators to compute respectively the V_q and V_d components of voltage to be synthesized by the VSC. (V_q in quadrature with V_1 controls active power and V_d in phase with V_1 controls reactive power). In manual voltage injection mode, regulators are not used. The reference values of injected voltage V_{dref} and V_{qref} are used to synthesize the converter voltage.

The UPFC block is a phasor model which does not include detailed representation of the power electronics. You must use it with the phasor simulation method, activated with the Powergui block. It can be used in three-phase power systems together with synchronous generators, motors, dynamic loads, and other FACTS and Renewable Energy systems to perform transient stability studies and observe impact of the UPFC on electromechanical oscillations and transmission capacity at fundamental frequency.

Parameters

The UPFC parameters are grouped in three categories: Power data, Control parameters (shunt converter), and Control parameters (series converter). Use the **Display** listbox to select which group of parameters you want to visualize.

Power Tab

Nominal voltage and frequency

The nominal line-to-line voltage in Vrms and the nominal system frequency in hertz. Default is [500e3, 60].

Shunt Converter rating

The nominal rating of the shunt converter in VA. Default is 100e6.

Shunt Converter impedance

The positive-sequence resistance R and inductance L of the shunt converter, in pu based on the nominal converter rating and nominal voltage. R and L represent the resistance and leakage inductance of the shunt transformer plus the resistance and inductance of the series filtering inductors connected at the VSC output. Default is [0.22/30, 0.22].

Shunt Converter initial current

The initial value of the positive-sequence current phasor (Magnitude in pu and Phase in degrees). If you know the initial value of the shunt current corresponding to the UPFC operating point you may specify it in order to start simulation in steady state. If you don't know this value, you can leave [0 0]. The system will reach steady-state after a short transient. Default is [0, 0].

Series Converter rating

The ratings of the series converter in VA and the maximum value of the injected voltage V_{conv} on the VSC side of the transformer (see single line diagram), in pu of nominal phase-to-ground voltage. Default is [100e6, 0.1].

Series Converter impedance

The positive-sequence resistance and inductance of the converter, in pu, based on the converter rated power and voltage. R and L represent the resistance and leakage inductance of the series transformer plus the resistance and inductance of the series filtering inductors connected at the VSC output. Default is [0.16/30, 0.16].

Series Converter initial current

The initial value of the positive-sequence current phasor (Magnitude in pu and Phase in degrees). If you know the initial value of the series current corresponding to the UPFC operating point you may specify it in order to start simulation in steady state. If you don't know this value, you can leave [0 0]. The system will reach steady-state after a short transient. Default is [0 0].

DC link nominal voltage

The nominal voltage of the DC link in volts. Default is 40000.

DC link total equivalent capacitance

The total capacitance of the DC link in farads. This capacitance value is related to the UPFC converter ratings and to the DC link nominal voltage. The energy stored in the capacitance (in joules) divided by the converter rated power (in VA) is a time duration which is usually a fraction of a cycle at nominal frequency. For example, for the default parameters, ($C=750\ \mu\text{F}$, $V_{dc}=40\ 000\ \text{V}$, $S_{nom}=100\ \text{MVA}$) this ratio $1/2 \cdot C \cdot V_{dc}^2 / S_{nom}$ is 6.0 ms, which represents 0.36 cycle for a 60 Hz frequency. If you change the default values of the nominal power rating and DC voltage, you should change the capacitance value accordingly. Default is 750e-6.

Shunt Converter Tab

Mode

Specifies the shunt converter mode of operation. Select either **Voltage regulation** (default) or **Var Control**.

Reference voltage Vref

This parameter is not enabled when the **Mode** parameter is set to **Var Control**.

Reference voltage, in pu, used by the voltage regulator. Default is 1.00.

External

When **External** is selected, a Simulink input named Vref appears on the block, allowing you to control the reference voltage from an external signal (in pu). The **Reference voltage Vref** parameter is therefore unavailable. Default is cleared.

Maximum rate of change of reference voltage

This parameter is not enabled when the **Mode** parameter is set to **Var Control**.

Maximum rate of change of the reference voltage, in pu/s, when an external reference voltage is used. Default is 0.1.

Droop

This parameter is not enabled when the **Mode** parameter is set to **Var Control**.

Droop reactance, in pu/shunt converter rating S_{nom} , defining the slope of the V-I characteristic. Default is 0.03.

Vac Regulator gains: [Kp Ki]

This parameter is not enabled when the **Mode** parameter is set to **Var Control**.

Gains of the AC voltage PI regulator. Specify proportional gain Kp in (pu of I)/(pu of V), and integral gain Ki, in (pu of I)/(pu of V)/s, where V is the AC voltage error and I is the output of the voltage regulator. Default is [5, 1000].

Reactive power setpoint Qref

This parameter is not enabled when the **Mode** parameter is set to **Voltage regulation**.

Reference reactive power, in pu, when the shunt converter is in **Var Control** mode. Default is 0.

Maximum rate of change of reactive power setpoint

This parameter is not enabled when the **Mode** parameter is set to **Voltage regulation**.

Maximum rate of change of the reference reactive power, in pu/s. Default is 2.

Vdc Regulator gains: [Kp Ki]

Gains of the DC voltage PI regulator which controls the voltage across the DC bus capacitor. Specify proportional gain Kp in (pu of I)/Vdc, and integral gain Ki, in (pu of I)/Vdc/s, where Vdc is the DC voltage error and I is the output of the voltage regulator. Default is [0.1e-3, 20e-3].

Current Regulators gains: [Kp Ki]

Gains of the inner current regulation loop. Default is [0.5, 25]/3.

Specify proportional gain Kp in (pu of V)/(pu of I), integral gain Ki, in (pu of V)/(pu of I)/s, where V is the output of the d or q current regulator and I is the Id or Iq current error.

The current regulator is assisted by a feed forward regulator. The feed forward gain in (pu of V)/(pu of I) is the shunt converter reactance (in pu) given by parameter L in the **Shunt converter impedance [R L]** parameters.

Series Converter Tab**Bypass Breaker**

Specifies the status of the bypass breaker connected inside the block across terminals A1, B1, C1 and A2, B2, C2. Select either **External Control**, **Open** (default) or **Closed**. If the bypass breaker is in external control, a Simulink input named Bypass appears on the block, allowing to control the status of the bypass breaker from an external signal (0 or 1).

Mode

Specifies the series converter mode of operation. Select either **Power flow control** (default) or **Manual voltage injection**.

Reference powers

This parameter is not enabled when the **Mode** parameter is set to **Manual voltage injection**.

Specify references values, in pu. Default is [5.87 -0.27].

External

When **External** is selected, a Simulink input named PQref appears on the block, allowing you to control the active and reactive powers from an external signal (in pu). The **Reference powers** parameter is therefore unavailable. Default is cleared. When **External** is selected, a Simulink input named Vdqref appears on the block, allowing you to control the injected voltage from an external signal (in pu). The **Reference voltages** parameter is therefore unavailable.

Maximum rate of change of reference powers

This parameter is not enabled when the **Mode** parameter is set to **Manual voltage injection**.

Specify maximum rate of change of Pref and Qref, in pu/s. Default is 1.

Power Regulator gains: [Kp Ki]

This parameter is not enabled when the **Mode** parameter is set to **Manual voltage injection**.

Gains of the PI regulators which control the line active power and reactive power. Specify proportional gain Kp in (pu of Vdq)/(pu of PQ), and integral gain Ki, in (pu of Vdq)/(pu of PQ)/s, where Vdq is the Vd or Vq injected voltage and PQ is the P or Q voltage error. Default is [0.025, 1.5].

Reference voltages

This parameter is not enabled when the **Mode** parameter is set to **Power flow control**.

Specify the direct-axis and quadrature-axis components of the voltage injected on the VSC side of the series transformer, in pu. Default is [0.05, 0.05].

Maximum rate of change for references voltages

This parameter is not enabled when the **Mode** parameter is set to **Power flow control**.

Specify maximum rate of change of the Vdref and Vqref voltages, in pu/s. Default is 0.2.

Inputs and Outputs

A1 B1 C1

The three input terminals of the UPFC.

A2 B2 C2

The three output terminals of the UPFC.

Trip

Apply a Simulink logical signal (0 or 1) to this input. When this input is high the shunt converter is disconnected and the series converter is bypassed. In addition, when the trip signal is high the shunt and series control systems are disabled. Use this input to implement a simplified version of the protection system.

Bypass

This input is visible only when the **Bypass Breaker** parameter is set to **External Control**.

Apply a Simulink logical signal (0 or 1) to this input. When this input is high the bypass breaker is closed.

Vdqref

This input is visible only when the **External control of injected voltage Vdref_Vqref** parameter is checked.

Apply a Simulink vectorized signal specifying the reference voltages Vdref and Vqref, in pu.

PQref

This input is visible only when the **External control of power references Pref_Qref** parameter is checked.

Apply a Simulink vectorized signal specifying the reference powers Pref and Qref, in pu.

m

Simulink output vector containing 34 UPFC internal signals. These signals are either voltage and current phasors (complex signals) or control signals. They can be individually accessed by using the Bus Selector block. They are, in order:

Signal	Signal Group	Signal Names	Definition
1-3	Power Vabc1 (cmplx)	Va1 (pu) Vb1 (pu) Vc1 (pu)	Phasor voltages (phase to ground) Va, Vb, Vc at the UPFC input terminals A1, B1, C1 (pu)

Signal	Signal Group	Signal Names	Definition
4-6	Power Iabc_SH (cmplx)	Ia_SH (pu) Ib_SH (pu) Ic_SH (pu)	Phasor currents Ia, Ib, Ic flowing into the shunt converter (pu)
7	Power	Vdc (V)	DC voltage (V)
8-10	Power Vabc2 (cmplx)	Va2 (pu) Vb2 (pu) Vc2 (pu)	Phasor voltages (phase to ground) Va, Vb, Vc at the UPFC output terminals A2, B2, C2 (pu)
11-13	Power Vabc_Inj (cmplx)	Va_Inj (pu) Vb_Inj (pu) Vc_Inj (pu)	Phasors of injected voltages Vs=V2-V1 (pu)
14_16	Power Iabc_SE (cmplx)	Ia_SE (pu) Ib_SE (pu) Ic_SE (pu)	Phasor currents Ia, Ib, Ic flowing in the series transformer from ABC1 to ABC2 (pu)
17	Control (Shunt)	Vm (pu)	Positive-sequence value of the measured voltage (pu)
18	Control (Shunt)	Vref (pu)	Reference voltage (pu)
19	Control (Shunt)	Qm (pu)	Shunt converter reactive power. A positive value indicates inductive operation.
20	Control (Shunt)	Qref (pu)	Reference reactive power (pu)
21	Control (Shunt)	Id (pu)	Direct-axis component of current (active current) flowing into the shunt converter (pu). A positive value indicates active power flowing into shunt converter.
22	Control (Shunt)	Iq (pu)	Quadrature-axis component of current (reactive current) flowing into the shunt converter (pu). A positive value indicates capacitive operation.
23	Control (Shunt)	Idref (pu)	Reference value of direct-axis component of current flowing into shunt converter (pu)
24	Control (Shunt)	Iqref (pu)	Reference value of quadrature-axis component of current flowing into shunt converter (pu)

Signal	Signal Group	Signal Names	Definition
25	Control (Shunt)	modindex	The modulation index m of the shunt converter PWM modulator. A positive number $0 < m < 1$. $m=1$ corresponds to the maximum voltage which can be generated by the VSC without overmodulation.
26	Control (Series)	P (pu)	Measured active power flowing out of terminals A2, B2, C2 (pu)
27	Control (Series)	Q (pu)	Measured reactive power flowing out of terminals A2, B2, C2 (pu)
28	Control (Series)	Pref (pu)	Reference active power (pu)
29	Control (Series)	Qref (pu)	Reference reactive power (pu)
30	Control (Series)	Vd_conv (pu)	Direct-axis injected voltage on VSC side of series transformer (pu)
31	Control (Series)	Vq_conv (pu)	Quadrature-axis injected voltage on VSC side of series transformer (pu)
32	Control (Series)	Vdref (pu)	Reference value of direct-axis injected voltage (pu)
33	Control (Series)	Vqref (pu)	Reference value of quadrature-axis injected voltage (pu)
34	Control (Series)	modindex	The modulation index m of the series converter PWM modulator. A positive number $0 < m < 1$. $m=1$ corresponds to the maximum voltage V_{conv} which can be generated by the series converter without overmodulation.

Examples

See the “Unified Power Flow Controller (UPFC) Phasor Model” example, which illustrates the use of a UPFC controlling power on a 500 kV, 60 Hz, system.

References

- [1] N. G. Hingorani, L. Gyugyi, “Understanding FACTS; Concepts and Technology of Flexible AC Transmission Systems,” IEEE Press book, 2000

See Also

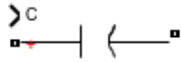
Static Synchronous Compensator (Phasor Type), Static Synchronous Series Compensator (Phasor Type)

Version History

Introduced in R2006a

Variable Capacitor

Discrete variable capacitor



Libraries:

Simscape / Electrical / Specialized Power Systems / Passives

Description

The Variable Capacitor block represents a linear time-varying capacitor. It implements a discrete variable capacitor as a voltage source. The capacitance is specified by the Simulink input signal. The capacitance value can be negative.

When you use a Variable Capacitor block in your model, set the powergui block **Simulation type** to **Discrete** and select the **Automatically handle Discrete solver and Advanced tab solver settings of blocks** parameter in the **Preferences** tab. The robust discrete solver is used to discretize the electrical model. Simulink signals an error if the robust discrete solver is not used.

Equations

The block uses the following equations for the relationship between the voltage, v , across the device and the current through the capacitor, i , when the capacitance at port **C** is C :

$$i = C \frac{dv}{dt}.$$

Ports

Input

C — Capacitance

Simulink signal

Input port associated with the capacitance. The capacitance can be negative and must be finite.

Conserving

+ — Positive terminal

specialized electrical

Specialized electrical conserving port associated with the capacitor positive voltage.

- — Negative terminal

specialized electrical

Specialized electrical conserving port associated with the capacitor negative voltage.

Parameters

Discrete Solver — Solver type

Backward Euler (default) | Trapezoidal

Robust integration method used by the block. The discrete solver method is automatically set to Trapezoidal when, in the powergui block, in the **Preferences** settings, you select **Automatically handle discrete solver and Advanced tab solver settings of blocks**.

The Trapezoidal robust solver is slightly more accurate than the Backward Euler robust solver, especially when the model is simulated at larger sample times. The Trapezoidal robust solver may produce slight damped numerical oscillations on machine voltage in no-load conditions, while the Backward Euler robust solver prevents oscillations and maintains good accuracy.

Minimal capacitance absolute value (F) — Minimal capacitance

1e-6 (default) | nonnegative scalar

Lower limit on the absolute value of the signal at port **C**. This limit prevents the signal from reaching a value that has no physical meaning. The value of this parameter must be greater than 0.

Version History

Introduced in R2019b

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

Nonlinear Inductor | Nonlinear Resistor | Variable Inductor | Variable Resistor | Variable-Ratio Transformer

Variable Inductor

Discrete variable inductor



Libraries:

Simscape / Electrical / Specialized Power Systems / Passives

Description

The Variable Inductor block represents a linear time-varying inductor. It implements a discrete variable inductor as a current source. The impedance is specified by the Simulink input signal. The inductance value can be negative.

When you use a Variable Inductor block in your model, set the powergui block **Simulation type** to **Discrete** and select the **Automatically handle Discrete solver and Advanced tab solver settings of blocks** parameter in the **Preferences** tab. The robust discrete solver is used to discretize the electrical model. Simulink signals an error if the robust discrete solver is not used.

Equations

The block uses the following equations for the relationship between the voltage, v , across the device and the current through the inductor, i . The block input specifies the value of the inductance. The flux linkage of the inductor is equal to the specified inductance multiplied by the inductor current:

$$v = \frac{d(\phi)}{dt},$$

$$\phi = L * i.$$

Ports

Input

L — Inductance
Simulink signal

Input port associated with the inductance. The inductance can be negative and must be finite.

Conserving

+ — Positive terminal
specialized electrical

Specialized electrical conserving port associated with the inductor positive voltage.

- — Negative terminal
specialized electrical

Specialized electrical conserving port associated with the inductor negative voltage.

Parameters

Discrete Solver — Solver type

Backward Euler (default) | Trapezoidal

Robust integration method used by the block. The discrete solver method is automatically set to Trapezoidal when, in the powergui block, in the **Preferences** settings, you select **Automatically handle discrete solver and Advanced tab solver settings of blocks**.

The Trapezoidal robust solver is slightly more accurate than the Backward Euler robust solver, especially when the model is simulated at larger sample times. The Trapezoidal robust solver may produce slight damped numerical oscillations on machine voltage in no-load conditions, while the Backward Euler robust solver prevents oscillations and maintains good accuracy.

Minimal inductance absolute value (H) — Minimal inductance

1e-6 (default) | nonnegative scalar

Lower limit on the absolute value of the signal at port **L**. This limit prevents the signal from reaching a value that has no physical meaning. The value of this parameter must be greater than 0.

Version History

Introduced in R2019b

Extended Capabilities

C/C++ Code Generation

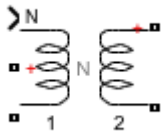
Generate C and C++ code using Simulink® Coder™.

See Also

Nonlinear Inductor | Nonlinear Resistor | Variable Capacitor | Variable Resistor | Variable-Ratio Transformer

Variable-Ratio Transformer

Discrete variable-ratio two-winding ideal transformer



Libraries:

Simscape / Electrical / Specialized Power Systems / Passives

Description

The Variable-Ratio Transformer block implements a discrete, variable-ratio, two-winding ideal transformer.

When you use a Variable-Ratio Transformer block, set the powergui block **Simulation type** to **Discrete** and select the **Automatically handle Discrete solver and Advanced tab solver settings of blocks** parameter in the **Preferences** tab. The robust discrete solver is used to discretize the electrical model. Simulink signals an error if the robust discrete solver is not used.

Equations

The voltage ratio between windings 1 and 2 is determined by the input signal, N , such that:

$$N = \frac{N_2}{N_1},$$

where:

- N_1 is the voltage at winding 1.
- N_2 is the voltage at winding 2.

Assumptions and Limitations

- Windings cannot be unconnected.

Ports

Input

N — Voltage ratio

Simulink signal

Signal representing the voltage ratio.

Conserving

+1 — Positive terminal of winding 1

specialized electrical

Specialized electrical conserving port associated with the positive voltage at winding 1.

-1 — Negative terminal of winding 1

specialized electrical

Specialized electrical conserving port associated with the negative voltage at winding 1.

+2 — Positive terminal of winding 2

specialized electrical

Specialized electrical conserving port associated with the positive voltage at winding 2.

-2 — Negative terminal of winding 2

specialized electrical

Specialized electrical conserving port associated with the negative voltage at winding 2.

Version History

Introduced in R2019b

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

Nonlinear Inductor | Nonlinear Resistor | Variable Capacitor | Variable Inductor | Variable Resistor

Variable Resistor

Discrete variable resistor



Libraries:

Simscape / Electrical / Specialized Power Systems / Passives

Description

The Variable Resistor block represents a linear time-varying resistor. It implements a discrete variable resistor as a current source. The resistance is specified by the Simulink input signal. The resistance value can be negative.

Equations

The block uses the following equations for the relationship between the voltage, v , across the device and the current through the inductor, i , when the resistance at port **R** is R :

$$v = R * i.$$

When you use a Variable Resistor block in your model, set the powergui block **Simulation type** to **Discrete** and select the **Automatically handle Discrete solver and Advanced tab solver settings of blocks** parameter in the **Preferences** tab. The robust discrete solver is used to discretize the electrical model. Simulink signals an error if the robust discrete solver is not used.

Ports

Input

R — Resistance
Simulink signal

Input port associated with the resistor. The resistance can be negative.

Conserving

+ — Positive terminal
specialized electrical

Specialized electrical conserving port associated with the resistor positive voltage.

- — Negative terminal
specialized electrical

Specialized electrical conserving port associated with the resistor negative voltage.

Parameters

Minimal resistance absolute value (Ohms) — Minimal resistance
1e-4 (default) | nonnegative scalar

Lower limit on the absolute value of the signal at port **R**. This limit prevents the signal from reaching a value that has no physical meaning. The value of this parameter must be greater than 0.

Version History

Introduced in R2019b

Extended Capabilities

C/C++ Code Generation

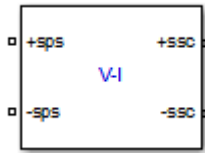
Generate C and C++ code using Simulink® Coder™.

See Also

Nonlinear Inductor | Nonlinear Resistor | Variable Capacitor | Variable Inductor | Variable-Ratio Transformer

Voltage-Current Simscape Interface

Ideal coupling between Simscape Electrical Specialized Power Systems and Simscape electrical circuits



Libraries:

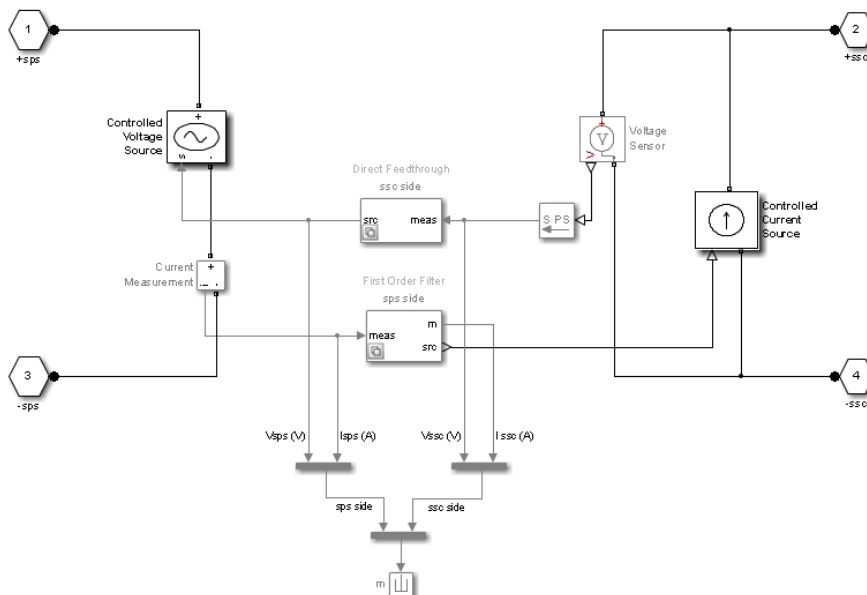
Simscape / Electrical / Specialized Power Systems / Utilities

Description

The Voltage-Current Simscape Interface block connects Simscape Electrical Specialized Power Systems circuits with electrical elements from the Simscape Foundation library. This block preserves current through the block and voltage across the block, conserving electrical power.

- On the Simscape Electrical Specialized Power Systems side, the electrical connector ports +sps and -sps act like a controlled voltage source, feeding voltage from the Simscape side to the connected Simscape Electrical Specialized Power Systems elements, while maintaining the current through the block.
- On the Simscape side, the electrical connector ports +ssc and -ssc act like a controlled current source, feeding current from the Simscape Electrical Specialized Power Systems side to the connected Simscape elements, while maintaining the voltage across the block.

The following figure gives an overview of the interface.



The block measures the voltage between the **+ssc** and **-ssc** ports and propagates it to the **+sps** and **-sps** ports, preserving the polarity. The block also measures the current flowing from the **+sps** to the **-sps** port and propagates it from the **-ssc** to the **+ssc** port.

You can specify a filter to break the algebraic loop between Simscape Electrical Specialized Power Systems and Simscape ports. You can specify the filter as a first order continuous transfer function or as a discrete unit delay block. You can apply it on the Simscape Electrical Specialized Power Systems side, to the measured voltage, or on the Simscape side, to the measured current. Depending on the measured signal where the filter is applied, the other signal is always a direct feedthrough.

If you do not specify a filter, direct feedthrough is applied on both measured signals. However, this configuration results in an algebraic loop containing the Voltage Sensor, Controlled Voltage Source, Current Measurement, and Controlled Current Source blocks, which might cause numerical issues.

Limitations

When simulating Simscape Electrical Specialized Power Systems models along with Simscape blocks, select the Simulink solver carefully.

Use the **ode23tb** solver for solving Simscape Electrical Specialized Power Systems circuits containing switches or nonlinear element. If you use other stiff solvers, Simscape Electrical Specialized Power Systems software gives you the following warning:

```
Warning: You have required continuous-time simulation of a system containing
switches or nonlinear elements. The ode23tb variable-step stiff solver with
relative tolerance set to 1e-4 is recommended in order to get best accuracy
and simulation performance. For some highly nonlinear models it may be
necessary to set the "Solver reset method" parameter to "Robust".
See "Improving Simulation Performance" chapter in Simscape Electrical
Specialized Power Systems documentation for additional information on how to
select an appropriate integration method.
'To ignore Simscape Electrical Specialized Power Systems warnings, select
"Disable Simscape Electrical SPS warnings" in the Powergui Preferences tab.';
```

The Simscape Electrical Specialized Power Systems warning can be disabled, as described in the warning message. However, if you use the **ode23tb** solver for solving models containing Simscape blocks, Simscape software gives you the following warning:

```
Warning: The solver chosen is not recommended if the model is stiff, which is
typical for models containing Simscape components. These solvers are typically
preferred for Simscape models:ode23t, ode15s or ode14x (fixed-step). To disable
this diagnostic,change the explicit solver diagnostic setting on the Simscape
panel of the model Configuration Parameters.
```

The Simscape warning can also be disabled, as described in the warning message.

Ports

Conserving

+sps — Positive terminal of Specialized Power Systems circuit
specialized electrical

Specialized electrical conserving port associated with the positive terminal of the Specialized Power Systems circuit.

-sps — Negative terminal of Specialized Power Systems circuit
specialized electrical

Specialized electrical conserving port associated with the negative terminal of the Specialized Power Systems circuit.

+ssc — Positive terminal of Simscape circuit
specialized electrical

Specialized electrical conserving port associated with the positive terminal of the Simscape circuit.

-ssc — Negative terminal of Simscape circuit
specialized electrical

Specialized electrical conserving port associated with the negative terminal of the Simscape circuit.

Output

m — Measurement port
Simulink signal

Bus signal that contains the Simscape Electrical Specialized Power Systems current and voltage and the Simscape current and voltages.

Dependencies

To enable this port, select the **Show measurement ports** parameter.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Direct feedthrough (no filter) — Direct feedthrough
off (default) | on

When you select this parameter, the block applies direct feedthrough between Simscape Electrical Specialized Power Systems and Simscape ports on both sides.

Filter type — Type of filter
Low pass filter (continuous) (default) | Unit delay (discrete) | First-order input filtering | Second-order input filtering

Set this parameter to **Low pass filter (continuous)** (default) to implement a first order continuous transfer function filter between Simscape Electrical Specialized Power Systems and Simscape ports.

Set this parameter to **Unit delay (discrete)** to implement a discrete unit delay filter between Simscape Electrical Specialized Power Systems and Simscape ports.

Set this parameter to **First-order input filtering** to implement a first order continuous transfer function filter with first-order filtering of the Simulink-PS Converter block that provides one derivative. You can apply this filter only on the Simscape Electrical Specialized Power Systems side.

Set this parameter to **Second-order input filtering** to implement a first order continuous transfer function filter with second-order filtering of the Simulink-PS Converter block that provides derivatives one and two. You can apply this filter only on the Simscape Electrical Specialized Power Systems side.

Dependencies

To enable this parameter, clear the **Direct feedthrough (no filter)** parameter.

Low pass filter time constant (s) — Time constant of low pass filter

1e-6 (default) | positive scalar

Time constant of low pass filter, in seconds.

Dependencies

To enable this parameter, set **Filter type** to Low pass filter (continuous), First-order input filtering, or Second-order input filtering.

Sample time (s) — Sample time of unit delay filter

50e-6 (default) | positive scalar

Sample time of the unit delay filter, in seconds.

Dependencies

To enable this parameter, set **Filter type** to Unit delay (discrete).

Initial voltage (V) — Initial voltage

0 (default) | nonnegative scalar

Initial voltage value of the measured Simscape Electrical Specialized Power Systems voltage.

Dependencies

To enable this parameter, set **Apply filter on** to Simscape side.

Initial current (A) — Initial current

0 (default) | nonnegative scalar

Initial current value of the measured Simscape Electrical current.

Dependencies

To enable this parameter, set **Apply filter on** to Specialized Power Systems side.

Apply filter on — Side to apply the filter on

inf (default) | nonnegative scalar

Set this parameter to Specialized Power Systems side (default) to apply the filter on the measured Simscape Electrical Specialized Power Systems voltage.

Set this parameter to Simscape side to apply the filter on the measured Simscape current.

Dependencies

To enable this parameter, clear the **Direct feedthrough (no filter)** parameter and set **Filter type** to Low pass filter (continuous) or Unit delay (discrete).

Show measurement port — Option to show measurement port

off (default) | on

If selected, the block exposes a Simulink output that returns the Simscape Electrical Specialized Power Systems current and voltage and to the Simscape current and voltages in a bus signal.

Version History

Introduced in R2011b

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

Current-Voltage Simscape Interface | Current-Voltage Simscape Interface (gnd) | Voltage-Current Simscape Interface (gnd)

Voltage-Current Simscape Interface (gnd)

Ideal coupling between Simscape Electrical Specialized Power Systems and Simscape electrical circuits



Libraries:

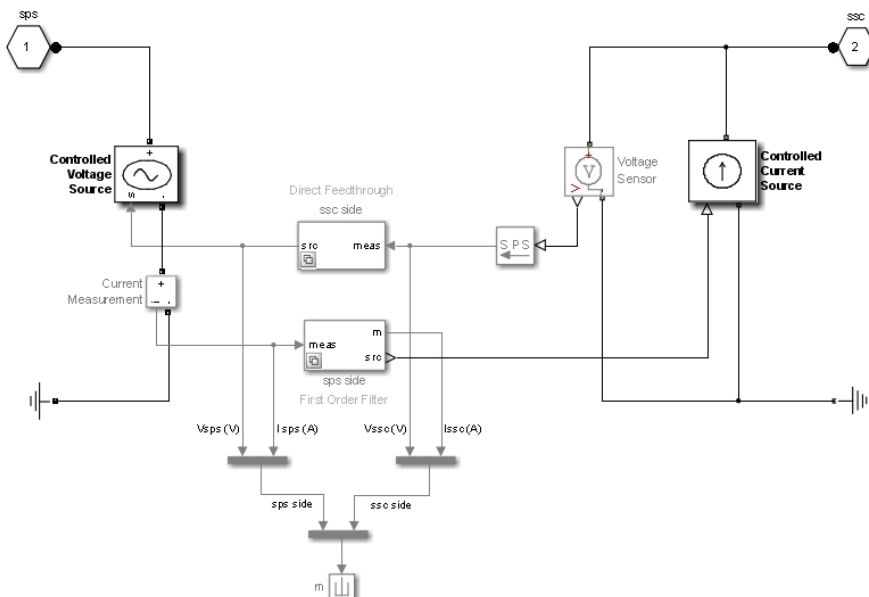
Simscape / Electrical / Specialized Power Systems / Utilities

Description

The Voltage-Current Simscape Interface (gnd) block connects Simscape Electrical Specialized Power Systems circuits with electrical elements from the Simscape Foundation library. This block preserves current through the block and voltage across the block, conserving electrical power.

- On the Simscape Electrical Specialized Power Systems side, the electrical connector port **sps** acts like a controlled voltage source referred to the ground, feeding voltage Simscape Electrical Specialized Power Systems side to the connected Simscape Electrical Specialized Power Systems elements.
- On the Simscape side, the electrical connector port **ssc** acts like a controlled current source referred to the ground, feeding current from the Simscape Electrical Specialized Power Systems side to the connected Simscape elements.

The following figure gives an overview of the interface.



The block measures the voltage between the **ssc** port and the Simscape electrical reference and propagates it to the **sps** port, preserving the polarity with respect to the Simscape Electrical

Specialized Power Systems electrical reference. The block also measures the current entering the `sps` port and propagates it to the `ssc` port.

You can specify a filter to break the algebraic loop between Simscape Electrical Specialized Power Systems and Simscape ports. You can specify the filter as a first order continuous transfer function or as a discrete unit delay block. You can apply it on the Simscape Electrical Specialized Power Systems side, to the measured voltage, or on the Simscape side, to the measured current. Depending on the measured signal where the filter is applied, the other signal is always a direct feedthrough.

If you do not specify a filter, direct feedthrough is applied on both measured signals. However, this configuration results in an algebraic loop containing the Voltage Measurement, Controlled Voltage Source, Current Sensor, and Controlled Current Source blocks, which might cause numerical issues.

Limitations

When simulating Simscape Electrical Specialized Power Systems models along with Simscape blocks, select the Simulink solver carefully.

Use the `ode23tb` solver for solving Simscape Electrical Specialized Power Systems circuits containing switches or nonlinear element. If you use other stiff solvers, Simscape Electrical Specialized Power Systems software gives you the following warning:

```
Warning: You have required continuous-time simulation of a system containing
switches or nonlinear elements. The ode23tb variable-step stiff solver with
relative tolerance set to 1e-4 is recommended in order to get best accuracy
and simulation performance. For some highly nonlinear models it may be
necessary to set the "Solver reset method" parameter to "Robust".
See "Improving Simulation Performance" chapter in Simscape Electrical
Specialized Power Systems documentation for additional information on how to
select an appropriate integration method.
'To ignore Simscape Electrical Specialized Power Systems warnings, select
"Disable Simscape Electrical SPS warnings" in the Powergui Preferences tab.';
```

The Simscape Electrical Specialized Power Systems warning can be disabled, as described in the warning message. However, if you use the `ode23tb` solver for solving models containing Simscape blocks, Simscape software gives you the following warning:

```
Warning: The solver chosen is not recommended if the model is stiff, which is
typical for models containing Simscape components. These solvers are typically
preferred for Simscape models:ode23t, ode15s or ode14x (fixed-step). To disable
this diagnostic,change the explicit solver diagnostic setting on the Simscape
panel of the model Configuration Parameters.
```

The Simscape warning can also be disabled, as described in the warning message.

Ports

Conserving

sps — Terminal of Specialized Power Systems circuit
specialized electrical

Specialized electrical conserving port associated with the terminal of the Specialized Power Systems circuit.

ssc — Terminal of Simscape circuit
specialized electrical

Specialized electrical conserving port associated with the terminal of the Simscape circuit.

Output

m — Measurement port

Simulink signal

Bus signal that contains the Simscape Electrical Specialized Power Systems current and voltage and the Simscape current and voltages.

Dependencies

To enable this port, select the **Show measurement ports** parameter.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Direct feedthrough (no filter) — Direct feedthrough

off (default) | on

When you select this parameter, the block applies direct feedthrough between Simscape Electrical Specialized Power Systems and Simscape ports on both sides.

Filter type — Type of filter

Low pass filter (continuous) (default) | Unit delay (discrete) | First-order input filtering | Second-order input filtering

Set this parameter to **Low pass filter (continuous)** (default) to implement a first order continuous transfer function filter between Simscape Electrical Specialized Power Systems and Simscape ports.

Set this parameter to **Unit delay (discrete)** to implement a discrete unit delay filter between Simscape Electrical Specialized Power Systems and Simscape ports.

Set this parameter to **First-order input filtering** to implement a first order continuous transfer function filter with first-order filtering of the Simulink-PS Converter block that provides one derivative. You can apply this filter only on the Simscape Electrical Specialized Power Systems side.

Set this parameter to **Second-order input filtering** to implement a first order continuous transfer function filter with second-order filtering of the Simulink-PS Converter block that provides derivatives one and two. You can apply this filter only on the Simscape Electrical Specialized Power Systems side.

Dependencies

To enable this parameter, clear the **Direct feedthrough (no filter)** parameter.

Low pass filter time constant (s) — Time constant of low pass filter

1e-6 (default) | positive scalar

Time constant of low pass filter, in seconds.

Dependencies

To enable this parameter, set **Filter type** to Low pass filter (continuous), First-order input filtering, or Second-order input filtering.

Sample time (s) — Sample time of unit delay filter

50e-6 (default) | positive scalar

Sample time of the unit delay filter, in seconds.

Dependencies

To enable this parameter, set **Filter type** to Unit delay (discrete).

Initial voltage (V) — Initial voltage

0 (default) | nonnegative scalar

Initial voltage value of the measured Simscape Electrical Specialized Power Systems voltage.

Dependencies

To enable this parameter, set **Apply filter on** to Simscape side.

Initial current (A) — Initial current

0 (default) | nonnegative scalar

Initial current value of the measured Simscape Electrical current.

Dependencies

To enable this parameter, set **Apply filter on** to Specialized Power Systems side.

Apply filter on — Side to apply the filter on

inf (default) | nonnegative scalar

Set this parameter to Specialized Power Systems side (default) to apply the filter on the measured Simscape Electrical Specialized Power Systems voltage.

Set this parameter to Simscape side to apply the filter on the measured Simscape current.

Dependencies

To enable this parameter, clear the **Direct feedthrough (no filter)** parameter and set **Filter type** to Low pass filter (continuous) or Unit delay (discrete).

Show measurement port — Option to show measurement port

off (default) | on

If selected, the block exposes a Simulink output that returns the Simscape Electrical Specialized Power Systems current and voltage and to the Simscape current and voltages in a bus signal.

Version History

Introduced in R2011b

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using Simulink® Coder™.

See Also

Current-Voltage Simscape Interface | Current-Voltage Simscape Interface (gnd) | Voltage-Current Simscape Interface

Voltage Measurement

Measure voltage in circuit

**Libraries:**

Simscape / Electrical / Specialized Power Systems / Sensors and Measurements

Description

The Voltage Measurement block measures the instantaneous voltage between two electric nodes. The Simulink signal output provides a signal that other Simulink blocks can use.

Ports

Output

v — Instantaneous voltage measured between + and - conserving ports
scalar

Outputs the instantaneous voltage measured between the + and - conserving ports, specified as a scalar.

Conserving

+ — Positive terminal
specialized electrical

Specialized electrical conserving port associated with the positive terminal of the Voltage Measurement block.

- — Negative terminal
specialized electrical

Specialized electrical conserving port associated with the negative terminal of the Voltage Measurement block.

Parameters

Output signal — Output signal format
Complex (default) | Real-Imag | Magnitude-Angle | Magnitude

Format of the output signal to use when you use the block in a phasor simulation.

Set this parameter to **Complex** to output the measured current as a complex value. The output is a complex signal.

Set this parameter to **Real - Imag** to output the real and imaginary parts of the measured current. The output is a vector of two elements.

Set this parameter to **Magnitude - Angle** to output the magnitude and angle of the measured current. The output is a vector of two elements.

Set this parameter to **Magnitude** to output the magnitude of the measured current. The output is a scalar value.

Dependencies

To enable this parameter, set the **Simulation type** parameter of the powergui block to **Phasor** or **Discrete phasor**.

Version History

Introduced before R2006a

Extended Capabilities**C/C++ Code Generation**

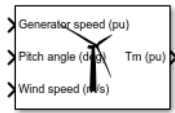
Generate C and C++ code using Simulink® Coder™.

See Also

[Current Measurement](#) | [powergui](#) | [Three-Phase V-I Measurement](#)

Wind Turbine

Implement model of variable pitch wind turbine



Libraries:

Simscape / Electrical / Specialized Power Systems / Electrical Machines

Description

The Wind Turbine block models the steady-state power characteristics of a wind turbine. The stiffness of the drive train is infinite and the friction factor and the inertia of the turbine must be combined with those of the generator coupled to the turbine. The output power of the turbine is given by the following equation.

$$P_m = c_p(\lambda, \beta) \frac{\rho A}{2} v_{\text{wind}}^3 \quad (1-16)$$

where:

P_m	Mechanical output power of the turbine (W)
c_p	Performance coefficient of the turbine
ρ	Air density (kg/m ³)
A	Turbine swept area (m ²)
v_{wind}	Wind speed (m/s)
λ	Tip speed ratio of the rotor blade tip speed to wind speed
β	Blade pitch angle (deg)

“Equation 1-16” can be normalized. In the per unit (pu) system we have:

$$P_{m\_pu} = k_p c_{p\_pu} v_{\text{wind\_pu}}^3$$

where:

P_{m_pu}	Power in pu of nominal power for particular values of ρ and A
c_{p_pu}	Performance coefficient in pu of the maximum value of c_p
$v_{\text{wind\_pu}}$	Wind speed in pu of the base wind speed. The base wind speed is the mean value of the expected wind speed in m/s.
k_p	Power gain for $c_{p\_pu}=1$ pu and $v_{\text{wind\_pu}}=1$ pu, k_p is less than or equal to 1

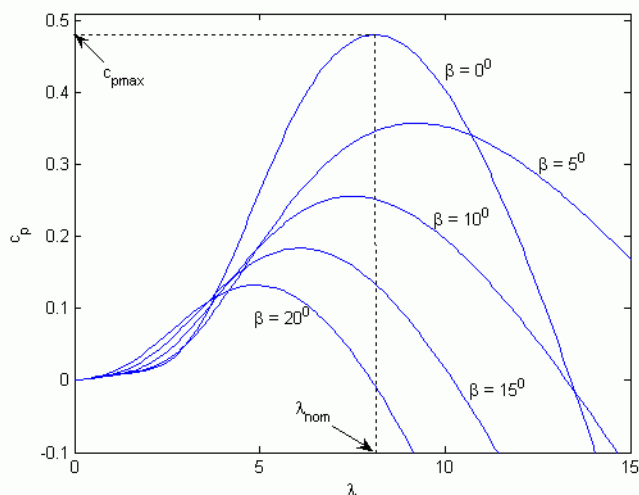
A generic equation is used to model $c_p(\lambda, \beta)$. This equation, based on the modeling turbine characteristics of [1], is:

$$c_p(\lambda, \beta) = c_1(c_2/\lambda_i - c_3\beta - c_4)e^{-c_5/\lambda_i} + c_6\lambda,$$

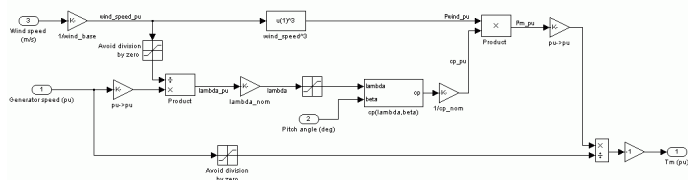
with:

$$\frac{1}{\lambda_i} = \frac{1}{\lambda + 0.08\beta} - \frac{0.035}{\beta^3 + 1}.$$

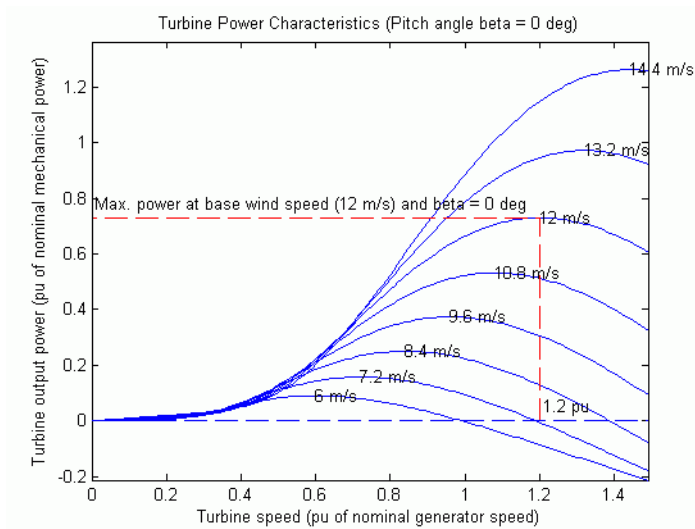
The coefficients c_1 to c_6 are: $c_1 = 0.5176$, $c_2 = 116$, $c_3 = 0.4$, $c_4 = 5$, $c_5 = 21$ and $c_6 = 0.0068$. The c_p - λ characteristics, for different values of the pitch angle β , are illustrated below. The maximum value of c_p ($c_{pmax} = 0.48$) is achieved for $\beta = 0$ degrees and for $\lambda = 8.1$. This particular value of λ is defined as the nominal value (λ_{nom}).



This figure shows the Simulink model of the turbine. The three inputs are the generator speed (ω_r pu) in pu of the nominal speed of the generator, the pitch angle in degrees, and the wind speed in m/s. The tip speed ratio λ in pu of λ_{nom} is obtained by the division of the rotational speed in pu of the base rotational speed (defined below) and the wind speed in pu of the base wind speed. The output is the torque applied to the generator shaft.



The illustration below shows the mechanical power P_m as a function of generator speed, for different wind speeds and for blade pitch angle $\beta = 0$ degrees. This figure is obtained with the default parameters (base wind speed = 12 m/s, maximum power at base wind speed = 0.73 pu ($k_p = 0.73$), and base rotational speed = 1.2 pu).



Ports

Input

Generator speed (pu) — Generator speed, pu
scalar

Generator speed based on the nominal speed of the generator, specified as a scalar, in pu.

Pitch angle (deg) — Pitch angle, deg
scalar

Pitch angle, specified as a scalar

Wind speed (m/s) — Wind speed, m/s
nonnegative scalar

Wind speed, specified as a nonnegative scalar, in m/s.

Output

Tm (pu) — Mechanical torque of wind turbine, pu
scalar

Mechanical torque of the wind turbine, returned as a scalar, in pu of the nominal generator torque. The nominal torque of the generator is based on the nominal generator power and speed.

Parameters

Nominal mechanical output power — Nominal output power
1.5e6 (default) | nonnegative scalar

The nominal output power in watts (W).

Base power of the electrical generator — Nominal power of electrical generator coupled to wind turbine

1.5e6/0.9 (default) | positive scalar

The nominal power of the electrical generator coupled to the wind turbine, in VA. This parameter is used to compute the output torque in pu of the nominal torque of the generator.

Base wind speed (m/s) — Base value of wind speed

12 (default) | positive scalar

The base value of the wind speed, in m/s, used in the per-unit system. The base wind speed is the mean value of the expected wind speed. This base wind speed produces a mechanical power that is usually lower than the turbine nominal power.

Maximum power at base wind speed — Maximum power

0.73 (default) | positive scalar

Power gain k_p at base wind speed in pu of the nominal mechanical power. k_p is less than or equal to 1.

Base rotational speed — Rotational speed at maximum power for base wind speed

1.2 (default) | positive scalar

The rotational speed at maximum power for the base wind speed. The base rotational speed is in pu of the base generator speed. For a synchronous or asynchronous generator, the base speed is the synchronous speed. For a permanent-magnet generator, the base speed is defined as the speed producing nominal voltage at no load.

Pitch angle beta to display wind turbine power characteristics — Pitch angle beta

0 (default) | nonnegative scalar

The pitch angle beta, in degrees, used to display the power characteristics. Beta must be greater than or equal to zero.

Display wind turbine power characteristics — Plot wind turbine power characteristics

button

Click to plot the turbine power characteristics for different wind speeds and for the specified pitch angle beta.

Version History

Introduced in R2006a

References

[1] Siegfried Heier, "Grid Integration of Wind Energy Conversion Systems," John Wiley & Sons Ltd, 1998, ISBN 0-471-97143-X

Extended Capabilities

C/C++ Code Generation

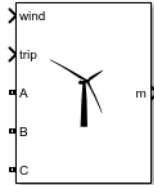
Generate C and C++ code using Simulink® Coder™.

See Also

Wind Turbine Induction Generator (Phasor Type) | Wind Turbine Doubly-Fed Induction Generator (Phasor Type)

Wind Turbine Doubly-Fed Induction Generator (Phasor Type)

Implement phasor model of variable speed doubly-fed induction generator driven by wind turbine

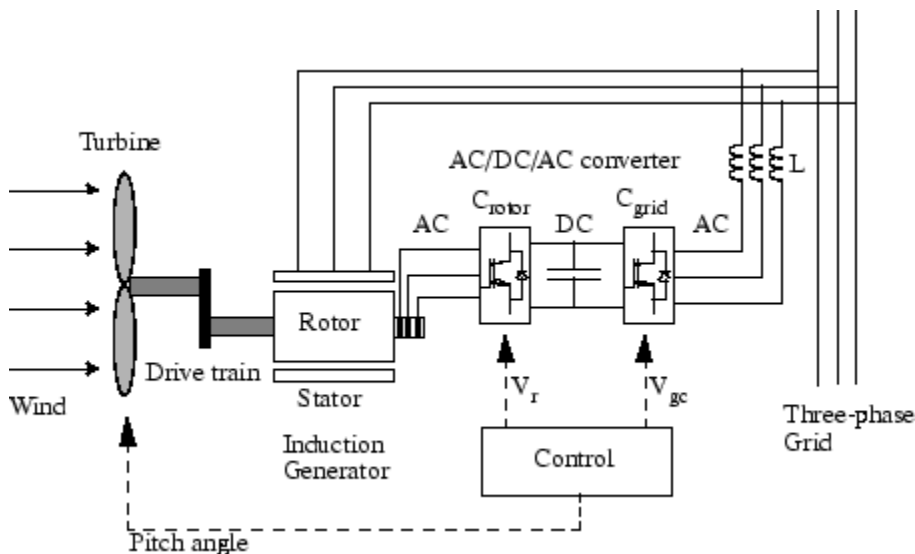


Library

Simscape / Electrical / Specialized Power Systems / Electrical Machines

Description

The wind turbine and the doubly-fed induction generator (WTDFIG) are shown in the figure.



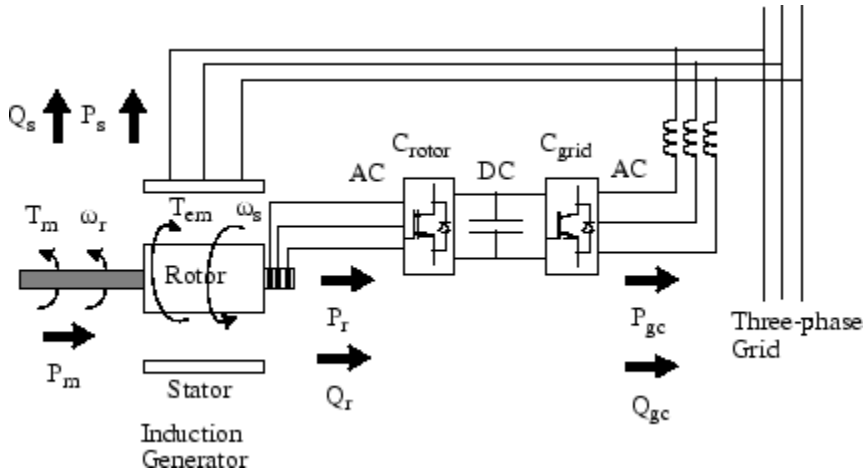
The Wind Turbine and the Doubly-Fed Induction Generator System

The AC/DC/AC converter is divided into two components: the rotor-side converter (C_{rotor}) and the grid-side converter (C_{grid}). C_{rotor} and C_{grid} are Voltage-Sourced Converters that use forced-commutated power electronic devices (IGBTs) to synthesize an AC voltage from a DC voltage source. A capacitor connected on the DC side acts as the DC voltage source. A coupling inductor L is used to connect C_{grid} to the grid. The three-phase rotor winding is connected to C_{rotor} by slip rings and brushes and the three-phase stator winding is directly connected to the grid. The power captured by the wind turbine is converted into electrical power by the induction generator and it is transmitted to the grid by the stator and the rotor windings. The control system generates the pitch angle command and the voltage

command signals V_r and V_{gc} for C_{rotor} and C_{grid} respectively in order to control the power of the wind turbine, the DC bus voltage, and the reactive power or the voltage at the grid terminals.

Operating Principle

Power flow, as illustrated in the figure, describes the operating principle of the Wind Turbine Doubly-Fed Induction Generator.



The Power Flow

The parameters for the power flow figure are:

P_m	Mechanical power captured by the wind turbine and transmitted to the rotor
P_s	Stator electrical power output
P_r	Rotor electrical power output
P_{gc}	C_{grid} electrical power output
Q_s	Stator reactive power output
Q_r	Rotor reactive power output
Q_{gc}	C_{grid} reactive power output
T_m	Mechanical torque applied to rotor
T_{em}	Electromagnetic torque applied to the rotor by the generator
ω_r	Rotational speed of rotor
ω_s	Rotational speed of the magnetic flux in the air-gap of the generator; this speed is named synchronous speed. It is proportional to the frequency of the grid voltage and to the number of generator poles.
J	Combined rotor and wind turbine inertia coefficient

The mechanical power and the stator electric power output are computed as follows:

$$P_m = T_m \omega_r, P_s = T_{em} \omega_s.$$

For a lossless generator the mechanical equation is:

$$J \frac{d\omega_r}{dt} = T_m - T_{em}.$$

In steady-state at fixed speed for a lossless generator $T_m = T_{em}$ and $P_m = P_s + P_r$.

It follows that:

$$P_r = P_m - P_s = T_m \omega_r - T_{em} \omega_s = -T_m \frac{\omega_s - \omega_r}{\omega_s} \omega_s = -s T_m \omega_s = -s P_s,$$

where s is defined as the slip of the generator: $s = (\omega_s - \omega_r)/\omega_s$.

Generally the absolute value of slip is much lower than 1 and, consequently, P_r is only a fraction of P_s . Since T_m is positive for power generation and since ω_s is positive and constant for a constant frequency grid voltage, the sign of P_r is a function of the slip sign. P_r is positive for negative slip (speed greater than synchronous speed) and it is negative for positive slip (speed lower than synchronous speed). For super-synchronous speed operation, P_r is transmitted to DC bus capacitor and tends to rise the DC voltage. For subsynchronous speed operation, P_r is taken out of DC bus capacitor and tends to decrease the DC voltage. C_{grid} is used to generate or absorb the power P_{gc} in order to keep the DC voltage constant. In steady-state for a lossless AC/DC/AC converter P_{gc} is equal to P_r and the speed of the wind turbine is determined by the power P_r absorbed or generated by C_{rotor} . The power control is explained below.

The phase-sequence of the AC voltage generated by C_{rotor} is positive for subsynchronous speed and negative for super-synchronous speed. The frequency of this voltage is equal to the product of the grid frequency and the absolute value of the slip.

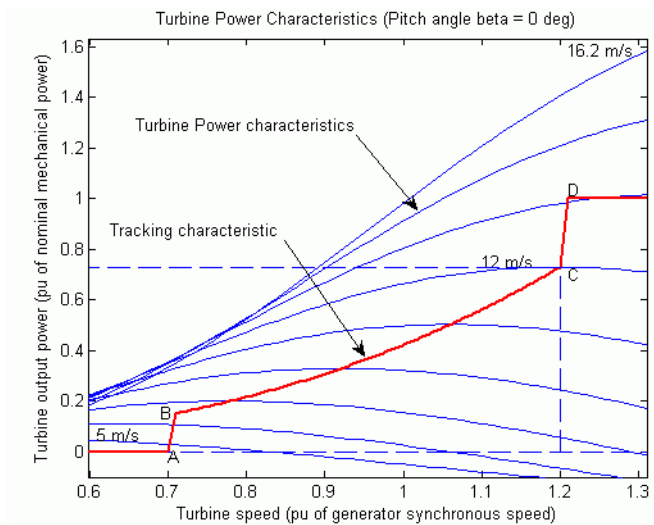
C_{rotor} and C_{grid} have the capability of generating or absorbing reactive power and could be used to control the reactive power or the voltage at the grid terminals.

C_rotor Control System

The rotor-side converter is used to control the wind turbine output power and the voltage (or reactive power) measured at the grid terminals.

Power Control

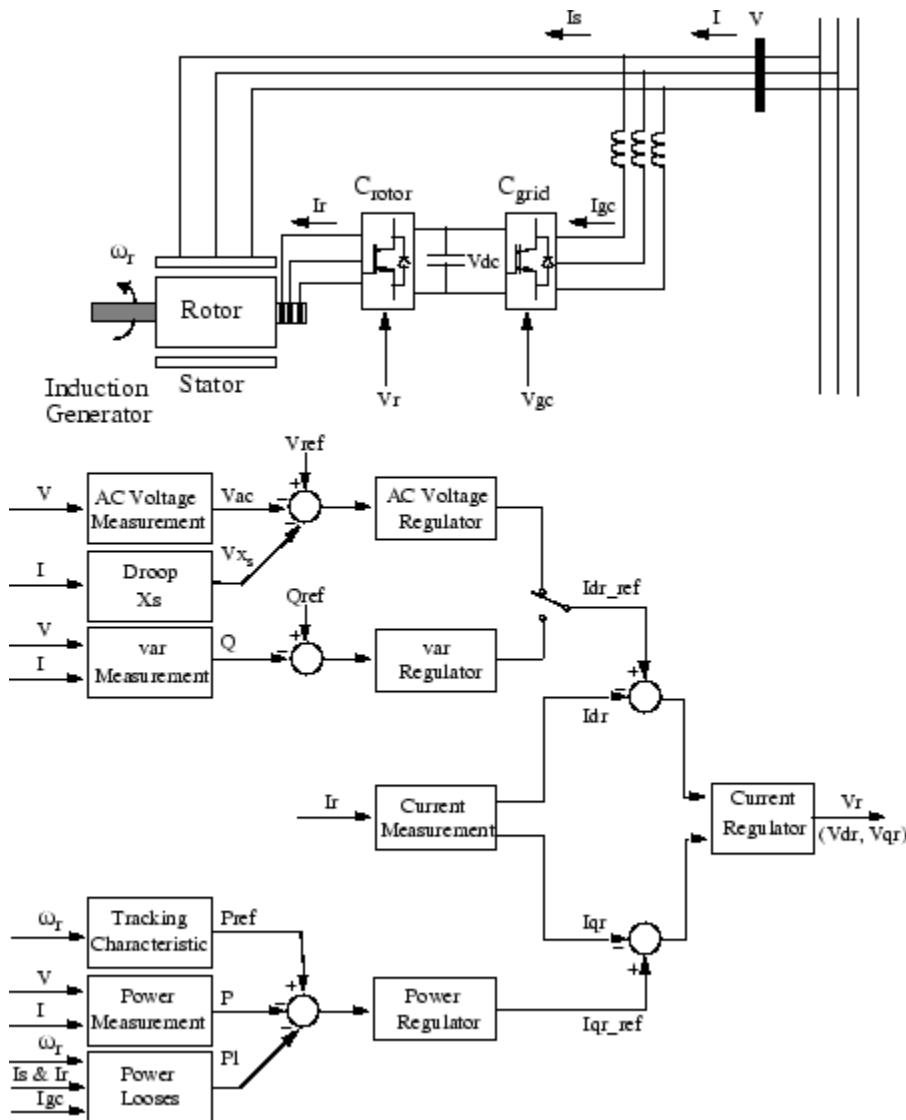
The power is controlled in order to follow a pre-defined power-speed characteristic, named tracking characteristic. An example of such a characteristic is illustrated by the ABCD curve superimposed to the mechanical power characteristics of the turbine obtained at different wind speeds.



Turbine Characteristics and Tracking Characteristic

The actual speed of the turbine ω_r is measured and the corresponding mechanical power of the tracking characteristic is used as the reference power for the power control loop. The tracking characteristic is defined by four points: A, B, C, and D. From zero speed to speed of point A, the reference power is zero. Between point A and point B the tracking characteristic is a straight line, the speed of point B must be greater than the speed of point A. Between point B and point C the tracking characteristic is the locus of the maximum power of the turbine (maxima of the turbine power versus turbine speed curves). The tracking characteristic is a straight line from point C and point D. The power at point D is one per unit (1 pu) and the speed of the point D must be greater than the speed of point C. Beyond point D the reference power is a constant equal to one per unit (1 pu).

The generic power control loop is illustrated in the figure.

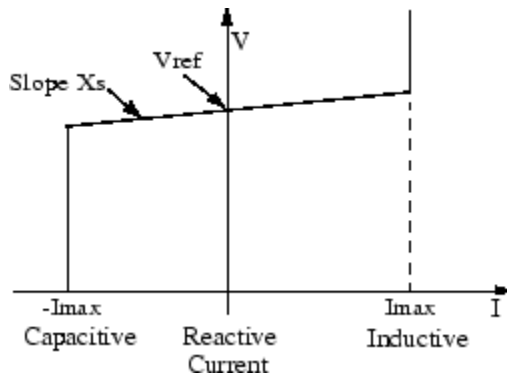


Rotor-Side Converter Control System

The actual electrical output power, measured at the grid terminals of the wind turbine, is added to the total power losses (mechanical and electrical) and is compared with the reference power obtained from the tracking characteristic. A Proportional-Integral (PI) regulator is used to reduce the power error to zero. The output of this regulator is the reference rotor current I_{qr_ref} that must be injected in the rotor by converter C_{rotor} . This is the current component that produces the electromagnetic torque T_{em} . The actual I_{qr} component of positive-sequence current is compared to I_{qr_ref} and the error is reduced to zero by a current regulator (PI). The output of this current controller is the voltage V_{qr} generated by C_{rotor} . The current regulator is assisted by feed forward terms which predict V_{qr} .

Voltage Control and Reactive Power Control

The voltage or the reactive power at grid terminals is controlled by the reactive current flowing in the converter C_{rotor} . The generic control loop is illustrated in the figure.



Wind Turbine V-I Characteristic

When the wind turbine is operated in voltage regulation mode, it implements the following V-I characteristic.

As long as the reactive current stays within the maximum current values ($-I_{max}$, I_{max}) imposed by the converter rating, the voltage is regulated at the reference voltage V_{ref} . However, a voltage droop is normally used (usually between 1% and 4% at maximum reactive power output), and the V-I characteristic has the indicated slope. In the voltage regulation mode, the V-I characteristic is described by the following equation:

$$V = V_{ref} + X_s I,$$

where:

V	Positive sequence voltage (pu)
I	Reactive current (pu/Pnom) ($I > 0$ indicates an inductive current)
X_s	Slope or droop reactance (pu/Pnom)
Pnom	Three-phase nominal power of the converter specified in the block dialog box

When the wind turbine is operated in var regulation mode, the reactive power at grid terminals is kept constant by a var regulator.

The output of the voltage regulator or the var regulator is the reference d-axis current I_{dr_ref} that must be injected in the rotor by converter C_{rotor} . The same current regulator as for the power control is used to regulate the actual I_{dr} component of positive-sequence current to its reference value. The output of this regulator is the d-axis voltage V_{dr} generated by C_{rotor} . The current regulator is assisted by feed forward terms which predict V_{dr} .

V_{dr} and V_{qr} are respectively the d-axis and q-axis of the voltage V_r .

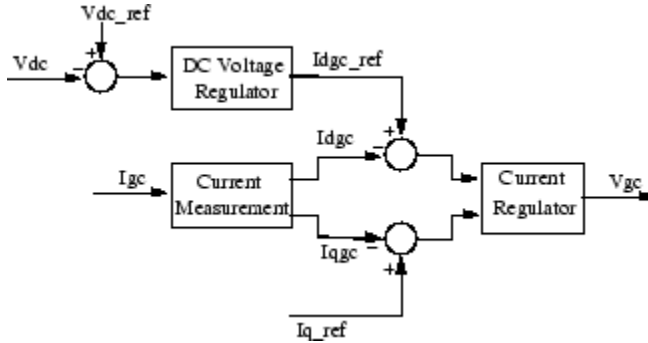
Note:

- for C_{rotor} control system and measurements the d-axis of the d-q rotating reference frame is locked on the generator mutual flux by a PLL which is assumed to be ideal in this phasor model.
- the magnitude of the reference rotor current I_{r_ref} is equal to $\sqrt{I_{dr\_ref}^2 + I_{qr\_ref}^2}$. The maximum value of this current is limited to 1 pu. When I_{dr_ref} and I_{qr_ref} are such that the magnitude is higher than 1 pu, the I_{qr_ref} component is reduced in order to bring back the magnitude to 1 pu.

C_grid Control System

The converter C_{grid} is used to regulate the voltage of the DC bus capacitor. In addition, this model allows using C_{grid} converter to generate or absorb reactive power.

The control system is illustrated in the figure.



Grid-Side Converter Control System

The control system consists of:

- Measurement systems measuring the d and q components of AC positive-sequence currents to be controlled as well as the DC voltage V_{dc} .
- An outer regulation loop consisting of a DC voltage regulator. The output of the DC voltage regulator is the reference current I_{dgc_ref} for the current regulator (I_{dgc} = current in phase with grid voltage which controls active power flow).
- An inner current regulation loop consisting of a current regulator. The current regulator controls the magnitude and phase of the voltage generated by converter C_{grid} (V_{gc}) from the I_{dgc_ref} produced by the DC voltage regulator and specified I_{q_ref} reference. The current regulator is assisted by feed forward terms which predict the C_{grid} output voltage.

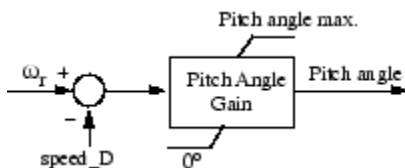
The magnitude of the reference grid converter current I_{gc_ref} is equal to

$$\sqrt{I_{dgc\_ref}^2 + I_{qr\_ref}^2}$$

. The maximum value of this current is limited to a value defined by the converter maximum power at nominal voltage. When I_{dgc_ref} and I_{q_ref} are such that the magnitude is higher than this maximum value, the I_{q_ref} component is reduced in order to bring back the magnitude to its maximum value.

Pitch Angle Control System

The pitch angle is kept constant at zero degrees until the speed reaches point D speed of the tracking characteristic. Beyond point D the pitch angle is proportional to the speed deviation from point D speed. The control system is illustrated in the following figure.



Pitch Control System

Turbine Model

The turbine model uses the Wind Turbine bloc of the Renewables/Wind Generation library. See documentation of this model for more details.

Induction Generator

The doubly-fed induction generator phasor model is the same as the wound rotor asynchronous machine (see the Machines library) with the following two points of difference:

- 1 Only the positive-sequence is taken into account, the negative-sequence has been eliminated.
- 2 A trip input has been added. When this input is high, the induction generator is disconnected from the grid and from C_{rotor} .

Parameters

The WTDFIG parameters are grouped in four categories: Generator data, Converters data, Turbine data, and Control parameters. Use the Display listbox to select which group of parameters you want to visualize.

Generator Tab

External turbine (Tm mechanical torque input)

When you select this parameter, the **Turbine** tab is not visible, and a Simulink input named Tm appears on the block, allowing to use an external signal for the generator input mechanical torque. This external torque must be in pu based on the nominal electric power and synchronous speed. For example, the external torque may come from a user defined turbine model. Following the convention used in the induction machine, the torque must be negative for power generation. Default is cleared.

Nominal power, line-to-line voltage and frequency

The nominal power in VA, the nominal line-to-line voltage in Vrms and the nominal system frequency in hertz. Default is [1.5e6/0.9 575 60].

Stator

The stator resistance Rs and leakage inductance Lls in pu based on the generator rating. Default is [0.00706 0.171].

Rotor

The rotor resistance Rr' and leakage inductance Llr', both referred to the stator, in pu based on the generator rating. Default is [0.005 0.156].

Magnetizing inductance

The magnetizing inductance Lm in pu based on the generator rating. Default is 2.90.

Inertia constant, friction factor and pairs of poles

Combined generator and turbine inertia constant H in seconds, combined viscous friction factor F in pu based on the generator rating and number of pole pairs p. Default is [5.04 0.01 3].

You may need to use your own turbine model, in order for example, to implement different power characteristics or to implement the shaft stiffness. Your model must then output the mechanical torque applied to the generator shaft. If the inertia and the friction factor of the turbine are implemented inside the turbine model you specify only the generator inertia constant H and the generator friction factor F.

Initial conditions

The initial slip s , electrical angle θ in degrees, stator phasor current magnitude in pu, stator phasor current phase angle in degrees, rotor phasor current magnitude in pu and rotor phasor current phase angle in degrees. Default is [0.2 0 0 0 0 0].

Turbine Tab**Nominal wind turbine mechanical output power**

This parameter is not visible when the **External turbine (Tm mechanical torque input)** parameter is selected.

The nominal turbine mechanical output power in watts. Default is 1.5e6.

Tracking characteristic speeds

This parameter is not visible when the **External turbine (Tm mechanical torque input)** parameter is selected.

Specify the speeds of point A to point D of the tracking characteristic in pu of the synchronous speed. speed_B must be greater than speed_A and speed_D must be greater than speed_C. Default is [0.7 0.71 1.2 1.21].

Power at point C

This parameter is not visible when the **External turbine (Tm mechanical torque input)** parameter is selected.

Specify the power of point C of the tracking characteristic in pu of the **Nominal wind turbine mechanical output power**. Default is 0.73.

Wind speed at point C

This parameter is not visible when the **External turbine (Tm mechanical torque input)** parameter is selected.

Specify wind speed in m/s for point C. The power at point C is the maximum turbine output power for the specified wind speed. Default is 12.

Pitch angle controller gain [Kp]

This parameter is not visible when the **External turbine (Tm mechanical torque input)** parameter is selected.

Proportional gain Kp of the pitch controller. Specify Kp in degrees/(speed deviation pu). The speed deviation is the difference between actual speed and speed of point D in pu of synchronous speed. Default is 500.

Maximum pitch angle

This parameter is not visible when the **External turbine (Tm mechanical torque input)** parameter is selected.

The maximum pitch angle in degrees. Default is 45.

Maximum rate of change of pitch angle

This parameter is not visible when the **External turbine (Tm mechanical torque input)** parameter is selected.

The maximum rate of change of the pitch angle in degrees/s. Default is 2.

Display wind turbine power characteristics

Click to plot the turbine power characteristics at zero degree of pitch angle for different wind speeds. The tracking characteristic is also displayed on the same figure.

Converters Tab

Converter maximum power

The maximum power of both C_{grid} and C_{rotor} in pu of the nominal power. This parameter is used to compute the maximum current at 1 pu of voltage for C_{grid} . The maximum current for C_{rotor} is 1 pu. Default is 0.5.

Grid-side coupling inductor

The coupling inductance L and its resistance R in pu based on the generator rating. Default is [0.15 0.15/100].

Coupling inductor initial currents

The coupling inductor initial phasor current in positive-sequence. Enter magnitude IL in pu and phase ph_IL in degrees. If you know the initial value of the current corresponding to the WTDFIG operating point you may specify it in order to start simulation in steady state. If you don't know this value, you can leave [0 90]. The system will reach steady-state after a short transient. Default is [0 90].

Nominal DC bus voltage

The nominal DC bus voltage in volts. Default is 1200.

DC bus capacitor

The total capacitance of the DC link in farads. This capacitance value is related to the WTDFIG rating and to the DC link nominal voltage. The energy stored in the capacitance (in joules) divided by the WTDFIG rating (in VA) is a time duration which is usually a fraction of a cycle at nominal frequency. For example, for the default parameters, ($C=10000 \mu\text{F}$, $V_{\text{dc}}=1200 \text{ V}$, $P_n=1.67 \text{ MVA}$) this ratio $1/2 \cdot C \cdot V_{\text{dc}}^2 / P_n$ is 4.3 ms, which represents 0.26 cycle for a 60 Hz frequency. If you change the default values of the nominal power rating and DC voltage, you should change the capacitance value accordingly. Default is 10000e-6.

Control Tab

Mode

Specifies the **WTDFIG** mode of operation. Select **Voltage regulation** (default) or **Var regulation**.

Reference grid voltage Vref

This parameter is not visible when the **Mode** parameter is set to **Var regulation**.

Reference voltage, in pu, used by the voltage regulator. Default is 1.0. When **External** is selected, a Simulink input named Vref appears on the block, allowing you to control the reference voltage from an external signal (in pu). The **Reference grid voltage** parameter is therefore unavailable.

Generated reactive power Qref

This parameter is not visible when the **Mode** parameter is set to **Voltage regulation**.

Reference generated reactive power at grid terminals, in pu, used by the var regulator. Default is 0. When **External** is selected, a Simulink input named Qref appears on the block, allowing you to control the reference reactive power from an external signal (in pu). The **Generated reactive power Qref** parameter is therefore unavailable.

Grid-side converter generated reactive current reference (Iq_ref)

Reference grid-side converter reactive current, in pu, used by the current regulator. Specify a positive value of Iq_ref for generated reactive power. Default is 0. When **External** is selected, a Simulink input named Iq_ref appears on the block, allowing you to control the grid-side converter reactive current from an external signal (in pu). The **Grid-side converter generated reactive current reference** parameter is therefore unavailable.

Grid voltage regulator gains [Kp Ki]

This parameter is not visible when the **Mode** parameter is set to Var regulation.

Gains of the AC voltage regulator. Specify proportional gain Kp in (pu of I)/(pu of V), and integral gain Ki, in (pu of I)/(pu of V)/s, where V is the AC voltage error and I is the output of the voltage regulator. Default is [1.25 300].

Droop Xs

This parameter is not visible when the **Mode** parameter is set to Var regulation.

Droop reactance, in pu/nominal power, defining the slope of the V-I characteristic. Default is 0.02.

Reactive power regulator gains [Kp Ki]

This parameter is not visible when the **Mode** parameter is set to Voltage regulation.

Gains of the var regulator. Specify proportional gain Kp in (pu of I)/(pu of Q), and integral gain Ki, in (pu of I)/(pu of Q)/s, where Q is the reactive power error and I is the output of the var regulator. Default is [0.05 5].

Power regulator gains [Kp Ki]

Gains of the power regulator. Specify proportional gain Kp in (pu of I)/(pu of P), and integral gain Ki, in (pu of I)/(pu of P)/s, where P is the power error and I is the output of the power regulator. Default is [1 100].

DC bus voltage regulator gains [Kp Ki]

Gains of the DC voltage regulator which controls the voltage across the DC bus capacitor. Specify proportional gain Kp in (pu of I)/(Vdc), and integral gain Ki, in (pu of I)/(Vdc)/s, where Vdc is the DC voltage error and I is the output of the voltage regulator. Default is [0.002 0.05].

Grid-side converter current regulator gains [Kp Ki]

Gains of the grid-side converter current regulator.

Specify proportional gain Kp in (pu of V)/(pu of I) and integral gain Ki, in (pu of V)/(pu of I)/s, where V is the output Vgc of the current regulator and I is the current error. Default is [1 100].

Rotor-side converter current regulator gains [Kp Ki]

Gains of the rotor-side converter current regulator. Default is [0.3 8].

Specify proportional gain Kp in (pu of V)/(pu of I) and integral gain Ki, in (pu of V)/(pu of I)/s, where V is the output Vr of the current regulator and I is the current error.

Maximum rate of change of reference grid voltage

This parameter is not visible when the **Mode** parameter is set to Var regulation.

Maximum rate of change of the reference voltage, in pu/s, when an external reference voltage is used. Default is 100.

Maximum rate of change of reference reactive power

This parameter is not visible when the **Mode** parameter is set to **Voltage regulation**.

Maximum rate of change of the reference reactive power, in pu/s, when an external reference reactive power is used. Default is 100.

Maximum rate of change of reference power

Maximum rate of change of the reference power in pu/s. Default is 1.

Maximum rate of change of converter reference currents

Maximum rate of change of the reference current in pu/s for both the rotor-side and the grid-side converters. Default is 200.

Inputs and Outputs**A B C**

The three terminals of the WTDFIG.

Trip

Apply a Simulink logical signal (0 or 1) to this input. When this input is high the WTDFIG is disconnected and its control system is disabled. Use this input to implement a simplified version of the protection system.

Wind (m/s)

This input is not visible when the **External mechanical torque** parameter is checked.

Simulink input of the wind speed in m/s.

Tm

This input is visible only when the **External mechanical torque** parameter is checked.

Simulink input of the mechanical torque. Tm must be negative for power generation. Use this input when using an external turbine model.

Vref

This input is visible only when the **Mode of operation** parameter is set to **Voltage regulation** and the **External grid voltage reference** parameter is checked.

Simulink input of the external reference voltage signal.

Qref

This input is visible only when the **Mode of operation** parameter is set to **Var regulation** and the **External generated reactive power reference** parameter is checked.

Simulink input of the external reference generated reactive power signal at grid terminals.

Iq_ref

This input is visible only when the **External reactive current Iq_ref for grid-side converter** parameter is checked.

Simulink input of the external reference grid-side converter reactive current signal.

m

Simulink output vector containing 29 WTDFIG internal signals. These signals can be individually accessed by using the Bus Selector block. They are, in order:

Signal	Signal Group	Signal Names	Definition
1-3	Iabc (cmplx) (pu)	Ia (pu) Ib (pu) Ic (pu)	Phasor currents Ia, Ib, Ic flowing into the WTDFIG terminals in pu based on the generator rating.
4-6	Vabc (cmplx) (pu)	Va (pu) Vb (pu) Vc (pu)	Phasor voltages (phase to ground) Va, Vb, Vc at the WTDFIG terminals in pu based on the generator rating.
7-8	Vdq_stator (pu)	Vd_stator (pu) Vq_stator (pu)	Direct-axis and quadrature-axis component of stator voltage in pu based on the generator rating. Vd_stator and Vq_stator are respectively the real and imaginary parts of the positive-sequence stator phasor voltage.
9-11	Iabc_stator (cmplx) (pu)	Ia_stator (pu) Ib_stator (pu) Ic_stator (pu)	Phasor currents Ia, Ib, Ic flowing into the stator in pu based on the generator rating.
12-13	Idq_stator (pu)	Id_stator (pu) Iq_stator (pu)	Direct-axis and quadrature-axis component of stator current in pu based on the generator rating. Id_stator and Iq_stator are respectively the real and imaginary parts of the positive-sequence stator phasor current.
14-15	Vdq_rotor (pu)	Vd_rotor (pu) Vq_rotor (pu)	Direct-axis and quadrature-axis component of rotor voltage in pu based on the generator rating. Vd_rotor and Vq_rotor are respectively the real and imaginary parts of the positive-sequence rotor phasor voltage.
16-17	Idq_rotor (pu)	Id_rotor (pu) Iq_rotor (pu)	Direct-axis and quadrature-axis component of currents flowing into the rotor in pu based on the generator rating. Id_rotor and Iq_rotor are respectively the real and imaginary parts of the positive-sequence rotor phasor current.
18		wr (pu)	Generator rotor speed (pu)
19		Tm (pu)	Mechanical torque applied to the generator (pu)
20		Te (pu)	Electromagnetic torque in pu based on the generator rating.

Signal	Signal Group	Signal Names	Definition
21-22	Vdq_grid_conv (pu)	Vd_grid_conv (pu) Vq_grid_conv (pu)	Direct-axis and quadrature-axis component of grid-side converter voltage in pu based on the generator rating. Vd_grid_conv and Vq_grid_conv are respectively the real and imaginary parts of the grid-side converter phasor voltage.
23-25	Iabc_grid_conv (cmplx) (pu)	Ia_grid_conv (pu) Ib_grid_conv (pu) Ic_grid_conv (pu)	Phasor currents Ia, Ib, Ic flowing into the grid-side converter in pu based on the generator rating.
26		P (pu)	WTDFIG output power. A positive value indicates power generation.
27		Q (pu)	WTDFIG output reactive power. A positive value indicates reactive power generation.
28		Vdc (V)	DC voltage (V).
29		Pitch_angle (deg)	Blade pitch angle in degrees.

Examples

See the “Wind Farm - Doubly-Fed Induction Generator (DFIG) Phasor Model” example, which illustrates the steady-state and dynamic performance of the WTDFIG in a 9 MW Wind Farm connected on a 25 kV, 60 Hz, system.

References

- [1] R. Pena, J.C. Clare, G.M. Asher, “Doubly fed induction generator using back-to-back PWM converters and its application to variable-speed wind-energy generation,” IEEE Proc.-Electr. Power Appl., Vol. 143, No. 3, May 1996
- [2] Vladislav Akhmatov, “Variable-Speed Wind Turbines with Doubly-Fed Induction Generators, Part I: Modelling in Dynamic Simulation Tools,” Wind Engineering Volume 26, No. 2, 2002
- [3] Nicholas W. Miller, Juan J. Sanchez-Gasca, William W. Price, Robert W. Delmerico, “DYNAMIC MODELING OF GE 1.5 AND 3.6 MW WIND TURBINE-GENERATORS FOR STABILITY SIMULATIONS,” GE Power Systems Energy Consulting, IEEE WTG Modeling Panel, Session July 2003

See Also

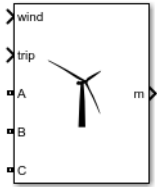
Wind Turbine, Wind Turbine Induction Generator (Phasor Type)

Version History

Introduced in R2006a

Wind Turbine Induction Generator (Phasor Type)

Implement phasor model of squirrel-cage induction generator driven by variable pitch wind turbine

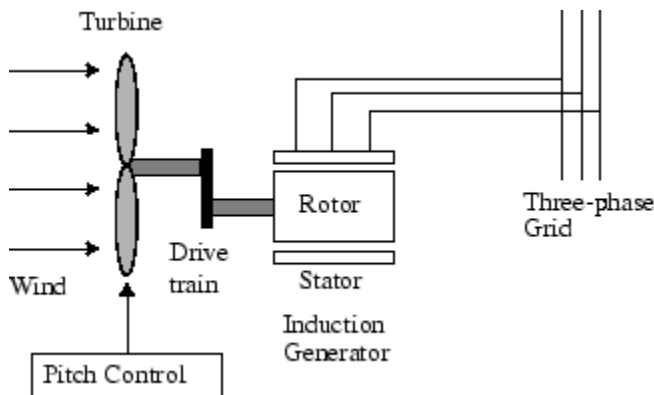


Libraries:

Simscape / Electrical / Specialized Power Systems / Electrical Machines

Description

The wind turbine and the induction generator (WTIG) are shown below. The stator winding is connected directly to the grid and the rotor is driven by the wind turbine. The power captured by the wind turbine is converted into electrical power by the induction generator and is transmitted to the grid by the stator winding. The pitch angle is controlled in order to limit the generator output power to its nominal value for high wind speeds. In order to generate power the induction generator speed must be slightly above the synchronous speed. But the speed variation is typically so small that the WTIG is considered to be a fixed-speed wind generator. The reactive power absorbed by the induction generator is provided by the grid or by some devices like capacitor banks, SVC, STATCOM, or synchronous condenser.

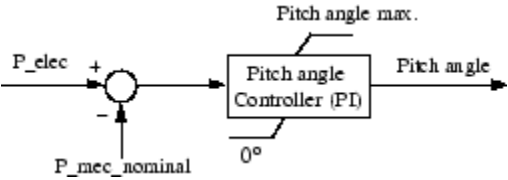


Pitch Angle Control System

The wind turbine model uses the Wind Turbine block of the Renewables/Wind Generation library. See documentation of this block for details.

A Proportional-Integral (PI) controller is used to control the blade pitch angle in order to limit the electric output power to the nominal mechanical power. The pitch angle is kept constant at zero degrees when the measured electric output power is under its nominal value. When it increases above its nominal value, the PI controller increases the pitch angle to bring back the measured power

to its nominal value. The control system is illustrated in the figure below:



Ports

Input

wind — Wind signal
Simulink signal

Simulink input of the wind speed in m/s.

Dependencies

To enable this port, clear the **External turbine (Tm mechanical torque input)** parameter.

Tm — Mechanical load torque
Simulink signal

Simulink input of the mechanical torque. Tm must be negative for power generation. Use this input when using an external turbine model.

Dependencies

To enable this port, select the **External turbine (Tm mechanical torque input)** parameter.

Trip — Mechanical load torque
0 | 1

Apply a Simulink logical signal (0 or 1) to this input. When this input is high the WTIG is disconnected. Use this input to implement a simplified version of the protection system.

Output

m — Measurement signal
vector

Simulink output vector containing 8 WTIG internal signals. These signals can be individually accessed by using the Bus Selector block. They are, in order:

Signal	Signal Names	Definition
1	Vabc (cmplx) (pu)	Phasor voltages (phase to ground) Va, Vb, Vc at the WTIG terminals in pu based on the generator ratings.
2	Iabc (cmplx) (pu)	Phasor currents Ia, Ib, Ic flowing into the WTIG terminals in pu based on the generator ratings.
3	P (pu)	WTIG output power in pu based on the generator ratings. A positive value indicates power generation.

Signal	Signal Names	Definition
4	Q (pu)	WTIG output reactive power in pu based on the generator ratings. A positive value indicates reactive power generation.
5	wr (pu)	Generator rotor speed (pu)
6	Tm (pu)	Mechanical torque applied to the generator in pu based on the generator ratings.
7	Te (pu)	Electromagnetic torque in pu based on the generator ratings.
8	Pitch_angle (deg)	Blade pitch angle in degrees.

Conserving

A — Terminal A
specialized electrical

Specialized electrical conserving port associated with the terminal A of the WTIG.

B — Terminal B
specialized electrical

Specialized electrical conserving port associated with the terminal B of the WTIG.

C — Terminal C
specialized electrical

Specialized electrical conserving port associated with the terminal C of the WTIG.

Parameters

To edit block parameters interactively, use the Property Inspector. From the Simulink Toolstrip, on the **Simulation** tab, in the **Prepare** gallery, select **Property Inspector**.

Generator

External turbine (Tm mechanical torque input) — Option to use external turbine
off (default) | on

When this parameter is selected, the **Turbine** tab is not visible in the mask dialog box, and a Simulink input named Tm appears on the block, allowing to use an external signal for the generator input mechanical torque. This external torque must be in pu based on the nominal electric power and synchronous speed of the generator. For example, the external torque may come from a user defined turbine model. By convention for the induction machine, the torque must be negative for power generation

Nominal power, line-to-line voltage and frequency [Pn(VA), Vn(Vrms), fn(Hz)] — Nominal power, line-to-line voltage, and frequency
[1.5e6/0.9, 575, 60] (default)

Nominal power in VA, the nominal line-to-line voltage in Vrms and the nominal system frequency in hertz.

Stator [Rs, Lls] (pu) — Stator resistance Rs and leakage inductance Lls
[0.004843, 0.1248] (default)

Stator resistance Rs and leakage inductance Lls in pu based on the generator ratings.

Rotor [Rr', Llr'] (pu) — Rotor resistance Rr' and leakage inductance Llr'
[0.004377, 0.1791] (default)

Rotor resistance Rr' and leakage inductance Llr', both referred to the stator, in pu based on the generator ratings.

Magnetizing inductance Lm (pu) — Magnetizing inductance Lm
6.77 (default)

Magnetizing inductance Lm in pu based on the generator ratings.

Inert constant, friction factor, and pairs of poles [H(s), F(pu), p] — Inert constant, friction factor, and pairs of poles
[5.04, 0.01, 3] (default)

Combined generator and turbine inertia constant H in seconds, combined viscous friction factor F in pu based on the generator ratings and number of pole pairs p.

You may need to use your own turbine model, in order for example, to implement different power characteristics or to implement the shaft stiffness. Your model must then output the mechanical torque applied to the generator shaft. If the inertia and the friction factor of the turbine are implemented inside the turbine model you specify only the generator inertia constant H and the generator friction factor F.

Initial conditions [s, th(deg), ias, ibs, ics (pu), ph_as, ph_bs, ph_cs (pu)] — Initial slip, electrical angle, stator current magnitude, and phase angle
[-0.01, 0, 0, 0, 0, 0, 0, 0] (default)

Initial slip s, electrical angle Θ in degrees, stator current magnitude in pu, and phase angle in degrees.

Turbine

This tab is not visible when you select the **External turbine (Tm mechanical torque input)** parameter.

Pitch angle beta used to display characteristics (beta >= 0) — Pitch angle used to display the wind turbine power characteristics
0 (default)

Pitch angle used to display the wind turbine power characteristics.

Dependencies

To enable this parameter, clear the **External turbine (Tm mechanical torque input)** parameter.

Nominal wind turbine mechanical output power (W) — Nominal turbine mechanical output power
1.5e6 (default)

Nominal turbine mechanical output power in watts.

Dependencies

To enable this parameter, clear the **External turbine (Tm mechanical torque input)** parameter.

Base wind speed (m/s) — Base value of wind speed

9 (default)

Base value of the wind speed, in m/s, used in the per unit system. The base wind speed is the mean value of the expected wind speed. This base wind speed produces a mechanical power which is usually lower than the turbine nominal power.

Dependencies

To enable this parameter, clear the **External turbine (Tm mechanical torque input)** parameter.

Base rotational speed (pu of base generator speed) — Rotational speed at maximum power for base wind speed

1 (default)

Rotational speed at maximum power for the base wind speed. The base rotational speed is in pu of the base generator speed.

Dependencies

To enable this parameter, clear the **External turbine (Tm mechanical torque input)** parameter.

Maximum power at base wind speed (pu of nominal mechanical power) — Maximum power at base wind speed of nominal mechanical power

1 (default)

Maximum power at base wind speed in pu of the nominal mechanical power.

Dependencies

To enable this parameter, clear the **External turbine (Tm mechanical torque input)** parameter.

Pitch angle controller gain [Kp Ki] — Proportional and Integral gains of pitch controller

[5 25] (default)

Proportional and Integral gains Kp and Ki of the pitch controller. Specify Kp in degrees/(power deviation pu) and Ki in degrees/(power deviation pu)/s. The power deviation is the difference between actual electrical output power and the nominal mechanical power in pu of the generator nominal power.

Dependencies

To enable this parameter, clear the **External turbine (Tm mechanical torque input)** parameter.

Maximum pitch angle (deg) — Maximum pitch angle

45 (default)

Maximum pitch angle in degrees.

Dependencies

To enable this parameter, clear the **External turbine (Tm mechanical torque input)** parameter.

Maximum rate of change of pitch angle (deg/s) — Maximum rate of change of pitch angle
2 (default)

Maximum rate of change of the pitch angle in degrees/s.

Dependencies

To enable this parameter, clear the **External turbine (Tm mechanical torque input)** parameter.

Display wind turbine power characteristics — Plot turbine power characteristics at specified pitch angle for different wind speeds

Click to plot the turbine power characteristics at the specified pitch angle for different wind speeds.

Dependencies

To enable this parameter, clear the **External turbine (Tm mechanical torque input)** parameter.

Version History

Introduced in R2006a

References

[1] Siegfried Heier, "Grid Integration of Wind Energy Conversion Systems," John Wiley & Sons Ltd, 1998, ISBN 0-471-97143-X

Extended Capabilities

C/C++ Code Generation

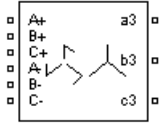
Generate C and C++ code using Simulink® Coder™.

See Also

Wind Turbine | Wind Turbine Doubly-Fed Induction Generator (Phasor Type)

Zigzag Phase-Shifting Transformer

Implement zigzag phase-shifting transformer with configurable secondary winding connection



Library

Simscape / Electrical / Specialized Power Systems / Power Grid Elements

Description

The Zigzag Phase-Shifting Transformer block implements a three-phase transformer with a primary winding connected in a zigzag configuration and a configurable secondary winding. The model uses three single-phase, three-winding transformers. The primary winding connects the windings 1 and 2 of the single-phase transformers in a zigzag configuration. The secondary winding uses the windings 3 of the single phase transformers, and they can be connected in one of the following ways:

- Y
- Y with accessible neutral
- Grounded Y
- Delta (D1), delta lagging Y by 30 degrees
- Delta (D11), delta leading Y by 30 degrees

Note The D1 and D11 notations refer to the following clock convention. It assumes that the reference Y voltage phasor is at noon (12) on a clock display. D1 and D11 refer respectively to 1 PM (lagging Y by 30 degrees) and 11 AM (leading Y by 30 degrees).

If the secondary winding is connected in Y, the secondary phase voltages are leading or lagging the primary voltages by the Phi phase angle specified in the parameters of the block. If the secondary winding is connected in delta (D11), an additional phase shift of +30 degrees is added to the phase angle. If the secondary winding is connected in delta (D1), a phase shift of -30 degrees is added to the phase angle.

The block takes into account the connection type you have selected and the icon of the block is automatically updated. An output port labeled N is added to the block if you select the Y connection with accessible neutral for the secondary winding.

The saturation characteristic, when activated, is the same as the one described for the Saturable Transformer block.

Parameters

Configuration Tab

Secondary winding (abc) connection

The winding connection for the secondary winding. Choices are Y, Yn, Yg (default), **Delta** (D1), and **Delta** (D11).

Saturable core

If selected, implements a saturable core. Default is cleared.

Specify initial fluxes

If selected, the initial fluxes are defined by the **Initial fluxes** parameter on the Parameters tab. This parameter is available only if the **Saturable core** parameter is selected. Default is cleared.

When the **Specify initial fluxes** parameter is not selected upon simulation, Simscape Electrical Specialized Power Systems software automatically computes the initial fluxes to start the simulation in steady state. The computed values are saved in the **Initial Fluxes** parameter and will overwrite any previous values.

Measurements

Select **Winding voltages** to measure the voltage across the winding terminals of the Three-Phase Transformer block.

Select **Winding currents** to measure the current flowing through the windings of the Three-Phase Transformer block.

Select **Fluxes** and **excitation currents** ($I_m + I_{Rm}$) to measure the flux linkage, in volt-seconds (V.s), and the total excitation current including iron losses modeled by R_m .

Select **Fluxes** and **magnetization currents** (I_m) to measure the flux linkage, in volt-seconds (V.s), and the magnetization current, in amperes (A), not including iron losses modeled by R_m .

Select **All measurements** (V, I, Flux) to measure the winding voltages, currents, magnetization currents, and the flux linkages.

Default is None.

Place a Multimeter block in your model to display the selected measurements during the simulation. In the **Available Measurements** list box of the Multimeter block, the measurements are identified by a label followed by the block name.

The labels used in the Multimeter are as follows.

Measurement	Label
Winding voltages of primary (zigzag)	Uprim_A:
Winding currents of primary (zigzag)	Iprim_A:
Winding voltages of secondary (Y, Yn, or Yg)	Usec_A:
Winding voltages of secondary (delta)	Usec_AB:

Measurement	Label
Winding currents of secondary (Y, Yn, or Yg)	Isec_A:
Winding currents of secondary (delta)	Isec_AB:
Fluxes (windings 1 of zigzag)	Flux_A:
Excitation currents (windings 1 of zigzag)	Iexc_A:
Magnetization currents (windings 1 of zigzag)	Imag_A:

Parameters Tab

Units

Specify the units used to enter the parameters of the Zigzag Phase-Shifting Transformer block. Select **pu** to use per unit. Select **SI** to use SI units. Changing the **Units** parameter from **pu** to **SI**, or from **SI** to **pu**, will automatically convert the parameters displayed in the mask of the block. The per unit conversion is based on the transformer rated power P_n in VA, nominal frequency f_n in Hz, and nominal voltage V_n , in Vrms, of the windings.

Nominal power and frequency

The nominal power rating, in volt-amperes (VA), and nominal frequency, in hertz (Hz), of the transformer. Note that the nominal parameters have no impact on the transformer model when the **Units** parameter is set to **SI**. Default is [100e6 60].

Primary (zigzag) nominal voltage Vp

The phase-to-phase nominal voltage in volts RMS, for the primary winding of the transformer. Default is 10e3.

Secondary nom. voltage phase shift

The phase-to-phase nominal voltage, in volts RMS, and the phase shift, in degrees, for the secondary winding of the transformer. Default is [30e3 +15].

Winding 1 zig-zag [R1 L1]

The resistance and leakage inductance of the windings 1 of the single-phase transformers used to implement the primary winding of the Zigzag Phase-Shifting Transformer. Default is [0.002 0.08] when the **Units** parameter is **pu** and [0.002488 0.00026399] when the **Units** parameter is **SI**.

Winding 2 zig-zag [R2 L2]

The resistance and leakage inductance of the windings 2 of the single-phase transformers used to implement the primary winding of the Zigzag Phase-Shifting Transformer. Default is [0.002 0.08] when the **Units** parameter is **pu** and [0.00017863 1.8954e-05] when the **Units** parameter is **SI**.

Winding 3 secondary [R3 L3]

The resistance and leakage inductance of the windings 3 of the single-phase transformers used to implement the secondary winding of the Zigzag Phase-Shifting Transformer. Default is [0.002 0.08] when the **Units** parameter is **pu** and [0.018 0.0019099] when the **Units** parameter is **SI**.

Magnetizing branch [Rm Lm]

The **Magnetizing branch** parameter is not accessible if the **Saturable core** check box is selected. Default is [500 500] when the **Units** parameter is pu and [622.01 622.01] when the **Units** parameter is SI.

The magnetization resistance Rm and inductance Lm, in pu, when the saturation is not simulated.

Magnetization resistance Rm

This parameter is accessible only if the **Saturable core** parameter on the **Configuration** tab is selected. Default is 500 when the **Units** parameter is pu and 622.01 when the **Units** parameter is SI.

The magnetization resistance Rm, in pu, when the saturation is simulated.

Saturation characteristic

This parameter is accessible only if the **Saturable core** parameter on the **Configuration** tab is selected. Default is [0,0 ; 0.0024,1.2 ; 1.0,1.52] when the **Units** parameter is pu and [0 0;17.569 28.988;7320.5 36.718] when the **Units** parameter is SI.

The saturation characteristic for the saturable core. Specify a series of current/ flux pairs (in pu) starting with the pair (0,0).

Initial fluxes

Specify initial fluxes for each phase of the transformer. This parameter is accessible only if the **Specify initial fluxes** parameter on the **Configuration** tab is selected. Default is [0,0,0].

When the **Specify initial fluxes** parameter is not selected upon simulation, Simscape Electrical Specialized Power Systems software automatically computes the initial fluxes to start the simulation in steady state. The computed values are saved in the **Initial Fluxes** parameter and will overwrite any previous values.

Advanced Tab

The **Advanced** tab of the block is not visible when you set the **Simulation type** parameter of the powergui block to Continuous, or when you select the **Automatically handle discrete solver** parameter of the powergui block. The tab is visible when you set the **Simulation type** parameter of the powergui block to Discrete, and when the **Automatically handle discrete solver** parameter of the powergui block is cleared.

Break Algebraic loop in discrete saturation model

When selected, a delay is inserted at the output of the saturation model computing magnetization current as a function of flux linkage (the integral of input voltage computed by a Trapezoidal method). This delay eliminates the algebraic loop resulting from trapezoidal discretization methods and speeds up the simulation of the model. However, this delay introduces a one simulation step time delay in the model and can cause numerical oscillations if the sample time is too large. The algebraic loop is required in most cases to get an accurate solution.

When cleared (default), the **Discrete solver model** parameter specifies the discretization method of the saturation model.

Discrete solver model

Select one of these methods to resolve the algebraic loop.

- **Trapezoidal iterative**—Although this method produces correct results, it is not recommended because Simulink tends to slow down and may fail to converge (simulation

stops), especially when the number of saturable transformers is increased. Also, because of the Simulink algebraic loop constraint, this method cannot be used in real time. In R2018b and previous releases, you used this method when the **Break Algebraic loop in discrete saturation model** parameter was cleared.

- **Trapezoidal robust**—This method is slightly more accurate than the Backward Euler robust method. However, it may produce slightly damped numerical oscillations on transformer voltages when the transformer is at no load.
- **Backward Euler robust**—This method provides good accuracy and prevents oscillations when the transformer is at no load.

The maximum number of iterations for the robust methods is specified in the **Preferences** tab of the powergui block, in the **Solver details for nonlinear elements** section. For real time applications, you may need to limit the number of iterations. Usually, limiting the number of iterations to 2 produces acceptable results. The two robust solvers are the recommended methods for discretizing the saturation model of the transformer.

For more information on what method to use in your application, see “Simulating Discretized Electrical Systems”.

Examples

See the help text of the “Three-Phase 48-Pulse GTO Converter” example.

In this model, a 48-pulse GTO converter is built with four Three-Level Bridge blocks and four Zigzag Phase-Shifting Transformer blocks. Harmonic neutralization is obtained by use of appropriate phase shifts introduced by the Zigzag connections (+7.5/–7.5 degrees) and of secondary winding connections (Y or Delta).

See Also

Multimeter, Three-Phase Transformer (Three Windings)

Version History

Introduced before R2006a

Functions

This table indicates the tasks performed by the commands described in this chapter.

<code>power_analyze</code>	Analyze electric circuit
<code>power_AynchronousMachineParams</code>	Estimate electrical parameters of double cage asynchronous machine based on standard manufacturer specifications
<code>power_customize</code>	Prepare a Simscape Electrical Specialized Power Systems block for customization
<code>power_flicker</code>	Perform statistical analysis of instantaneous flicker level signal
<code>power_hysteresis</code>	View and edit hysteresis characteristic for saturable core of Saturable Transformer blocks
<code>power_init</code>	Set initial state values of model built with Simscape Electrical Specialized Power Systems software
<code>power_initstates</code>	Compute and modify initial values of voltages and currents of model
<code>power_lineparam</code>	Compute RLC parameters of overhead transmission line from its conductor characteristics and tower geometry
<code>power_ltvview</code>	Create state-space object of model for use with the Linear System Analyzer tool
<code>power_new</code>	Create new Simscape Electrical Specialized Power Systems model populated by required and commonly used blocks
<code>power_PMSynchronousMachineParams</code>	Convert manufacturer specifications to mask parameters of Permanent Magnet Synchronous Machine block
<code>power_report</code>	Generate report of steady state variables, initial states, and machine load flow for model
<code>power_statespace</code>	Compute state-space model of linear electrical circuit
<code>power_steadystate</code>	Compute steady-state values of voltages and currents of model
<code>power_zmeter</code>	Compute impedance versus frequency of Impedance Measurement blocks of model

power_analyze

Analyze electric circuit

Syntax

```
sps = power_analyze('sys','structure')  
[A,B,C,D,x0,states,inputs,outputs,uss,xss,yss,frequencies,Hlin] =...  
power_analyze('sys')  
sps = power_analyze('sys','sort')  
sps = power_analyze('sys','ss')  
power_analyze('sys','net')  
SW = power_analyze('sys','getSwitchStatus')  
sps = power_analyze('sys','setSwitchStatus',SW)
```

Description

The `power_analyze` command computes the equivalent state-space model of the specified electrical model built with Simscape Electrical Specialized Power Systems software. It evaluates the A , B , C , D standard matrices of the state-space system described by the equations

$$\begin{aligned}\dot{x} &= Ax + Bu \\ y &= Cx + Du\end{aligned}$$

where the state vector x represents the inductor currents and capacitor voltages, the input vector u represents the voltage and current sources, and the output vector y represents the voltage and current measurements of the model.

Nonlinear elements, such as the motors or machines, are simulated either by current sources driven by the voltages across the nonlinear element terminals, or by voltage sources driven by the currents through the nonlinear element terminals. The nonlinear elements produce additional current and voltage source inputs to the u vector, and additional voltage and current measurements outputs to the y vector.

For the switching devices elements, such as the breakers, diodes, and thyristors, the state-space calculations include only the resistance and capacitance elements of the snubber devices.

The Simulink blocks of the model, as well as the internal Simulink models of the Simscape Electrical Specialized Power Systems nonlinear elements, are not represented in the state-space matrices.

State-Space Matrices

The A , B , C , D matrices are computed for the particular circuit topology where all the switch devices, if any, are considered to be open status (that is, with infinite impedance).

`power_analyze` also computes the `Aswitch`, `Bswitch`, `Cswitch`, and `Dswitch` matrices for the circuit topology that take into account the initial state (open/closed) of the Breaker and Ideal Switch blocks present in the model. The initial state of power electronic devices (Diodes, Thyristors, and so on) is considered to be open, unless a nonzero initial current value is specified in the mask of the device when `Lon` parameter is different from zero.

For a circuit that contains no switches, the `Aswitch`, `Bswitch`, `Cswitch`, and `Dswitch` matrices have exactly the same values as the `A`, `B`, `C`, `D` matrices.

States

In Simscape Electrical Specialized Power Systems software, each state variable name begins with a prefix `Uc_` for capacitor voltages or `Il_` for inductor currents, followed by the name of the block in which the element (C or L) is found.

A character vector is added to this prefix for blocks containing more than one inductance or capacitor. For example, the Linear Transformer block is represented with four state variables, one for each of the three leakage inductances, defined with the prefixes `Il_winding_x:`, where `x` is the winding number of the transformer, and one state for the magnetization inductance defined with the prefix `Il_Lm:`.

Inputs

Each input state variable name begins with a prefix `U_` for voltage sources or `I_` for current sources, followed by the name of the source block. Text can be added to the prefix for blocks containing more than one source. For example, the Synchronous Machine block produces two current inputs with prefixes `I_A:` and `I_B:` (phase A and phase B machine currents).

Outputs

Each output state variable name begins with a prefix `U_` for voltage outputs or `I_` for current outputs, followed by the name of the block that produces the output. Text can be added to the prefix for blocks containing more than one output. For example, the Synchronous Machine block produces two voltage outputs with prefixes `U_AB:` and `U_BC:` (two machine phase-to-phase voltages).

Sign Conventions

The following conventions are used for inputs:

- Source current flowing in the arrow direction is positive.
- Positive source voltage is indicated by a + sign on the icon.

The sign conventions used for voltages and currents of state variables and measurement outputs are described in “Measuring Voltages and Currents” in the Simscape Electrical *User's Guide (Specialized Power Systems)*.

Output Arguments: Structure

`sps = power_analyze('sys','structure')` creates a structure array `sps` with fields and values describing the model `sys`.

The fields of the structure array are defined in the following order.

Field	Description
<code>circuit</code>	Name of the model
<code>states</code>	char array of state variable names
<code>inputs</code>	char array of system input names
<code>outputs</code>	char array of system output names

Field	Description
A	nstates-by-nstates state-space <i>A</i> matrix
B	nstates-by-ninput state-space <i>B</i> matrix
C	noutput-by-nstates state-space <i>C</i> matrix
D	noutput-by-ninput state-space <i>D</i> matrix
x0	nstates-by-1 vector of initial conditions of the state variables listed in states
xss	nstates-by-nfreq steady-state values of states. A set of values is computed for every frequency specified in the frequencies vector.
uss	ninput-by-nfreq steady-state values of inputs. A set of values is computed for every frequency specified in the frequencies vector.
yss	noutput-by-nfreq steady-state values of outputs. A set of values is computed for every frequency specified in the frequencies vector.
frequencies	1-by-nfreq vector of input source frequencies ordered by increasing values
DependentStates	char array of dependent state variable names. The dependent states are not included in the state-space equations.
x0DependentStates	Vector of initial conditions of dependent states
xssDependentStates	nstates-by-nfreq steady-state values of dependent states
Adiscrete	Discrete state-space <i>A</i> matrix. Returns an empty value when the Powergui is in continuous or in phasor mode.
Bdiscrete	Discrete state-space <i>B</i> matrix. Returns an empty value when the Powergui is in continuous or in phasor mode.
Cdiscrete	Discrete state-space <i>C</i> matrix. Returns an empty value when the Powergui is in continuous or in phasor mode.
Ddiscrete	Discrete state-space <i>D</i> matrix. Returns an empty value when the Powergui is in continuous or in phasor mode.
x0discrete	Vector of discrete initial conditions. Returns an empty value when the Powergui is in continuous or in phasor mode.
SampleTime	Sample time value used to compute discrete state-space matrices
Aswitch	<i>A</i> matrix taking into account the initial status of switch devices
Bswitch	<i>B</i> matrix taking into account the initial status of switch devices
Cswitch	<i>C</i> matrix taking into account the initial status of switch devices
Dswitch	<i>D</i> matrix taking into account the initial status of switch devices
x0switch	Vector of initial values of switch currents
Hlin	noutput-by-ninput-by-nfreq complex transfer function of impedances of the linear system corresponding to the frequencies contained in the frequencies vector. For a particular frequency, Hlin is defined by $yss(:,i) = Hlin(:,:,i) * uss(:,i)$
OscillatoryModes	Display the oscillatory modes of the state-space system

The table uses the following conventions:

- `nstates` is the number of states.
- `ninput` is the number of inputs.
- `noutput` is the number of outputs.
- `nfreq` is the number of input source frequencies.

`[A,B,C,D,x0,states,inputs,outputs,uss,xss,yss,frequencies,Hlin] = ...`
`power_analyze('sys')` returns the state-space calculations in separate variables.

Output Arguments: Sort

`sps = power_analyze('sys','sort')` returns a structure array `sps` with the following fields related to the interconnection of Simscape Electrical Specialized Power Systems blocks in a model. The fields are defined in the following order.

Field	Description
<code>circuit</code>	Name of the model
<code>SampleTime</code>	Sample time for discrete systems
<code>RlcBranch</code>	<code>rlc</code> matrix in the <code>power_statespace</code> format
<code>RlcBranchNames</code>	List of blocks containing the state variable
<code>SourceBranch</code>	Source matrix in the <code>power_statespace</code> format
<code>SourceBranchNames</code>	Names of the blocks defined as sources
<code>InputNames</code>	Names of the inputs of the system
<code>OutputNames</code>	Names of the outputs of the system
<code>OutputExpressions</code>	Output expression in the <code>power_statespace</code> format
<code>OutputMatrix</code>	Output expression in matrix format (internal)
<code>MeasurementBlocks</code>	Names of the voltage and current measurement blocks

Output Arguments: ss

`sps = power_analyze('sys','ss')` creates a continuous state-space model of the model `sys` with matrices A , B , C , D . You must have Control System Toolbox software installed for this option. The output is a state-space object.

Output Arguments: Net

`power_analyze('sys','net')` generates a netlist stored in a file, `sys.net`. The file contains the node numbers automatically generated by `power_analyze`, as well as parameter values of all linear elements. See the formats described in the `power_statespace` reference page.

Output Arguments: getSwitchStatus

`SW = power_analyze('sys','getSwitchStatus')` returns a structure array with switch names and their initial status. You can use the `SW` structure to specify switch statuses for a particular circuit topology and to compute the corresponding state-space matrices using the command `sps = power_analyze('sys','setSwitchStatus',SW)`. The `SW` structure contains the following fields.

Field	Description
SwitchNames	Names of the switches of the system
SwitchStatus	Vector of initial states of switches

Output Arguments: setSwitchStatus

`sps = power_analyze('sys', 'setSwitchStatus', SW)` creates a structure array `sps` with fields and values describing the state-space matrices of model `sys` for the switch status defined in `SW`. Use the command

`SW = power_analyze('sys', 'getSwitchStatus')` to obtain the `SW` structure array.

Field	Description
A	nstates-by-nstates state-space <i>A</i> matrix
B	nstates-by-ninput state-space <i>B</i> matrix
C	noutput-by-nstates state-space <i>C</i> matrix
D	noutput-by-ninput state-space <i>D</i> matrix
x0	nstates-by-1 vector of initial conditions of the state variables listed in <code>states</code>

Examples

Obtain the state-space matrices and steady-state voltages and currents for the `power_netsim2` model.

The command

```
sps = power_analyze('power_netsim2','structure');
```

returns the state-space model in the `sps` structure variable.

```
sps.A =
    1.0e+04 *
         0          6.2500
    -0.0083     -1.4250
```

```
sps.uss =
         0
       1000
```

```
sps.xss =
    1.0e+02 *
    4.8392 - 5.1314i
    0.0310 + 0.0292i
```

```
sps.yss =
    1.0e+02 *
    8.5535 - 1.6287i
         0
```

```
sps.inputs =
    I_Breaker
    U_Source
```

```
sps.outputs =  
    U_Breaker  
    I_Current Measurement
```

The inductor current of the 51-ohm, 12-mH block and the capacitor voltage of the 120-ohm, 16- μ F block are the two state variables in this circuit. The Breaker block is a nonlinear element that is represented by a current source (the first input) driven by the voltage across its terminals (the first output).

See Also

[power_statespace](#) | [power_init](#) | [Powergui](#)

power_AsymynchronousMachineParams

Estimate electrical parameters of double cage asynchronous machine based on standard manufacturer specifications

Syntax

```
power_AsymynchronousMachineParams
params = power_AsymynchronousMachineParams(spec)
params = power_AsymynchronousMachineParams(spec, options)
[params, spec2] = power_AsymynchronousMachineParams(spec, options)
[params, spec2, errors] = power_AsymynchronousMachineParams(spec, options)
```

Description

`power_AsymynchronousMachineParams` opens a graphical user interface to compute the mask parameters required by the double-cage asynchronous machine block based on standard manufacturer specifications.

`params = power_AsymynchronousMachineParams(spec)` computes the machine parameters from the input structure `spec`, which contains the manufacturer data. The solution is returned in the output structure `params`.

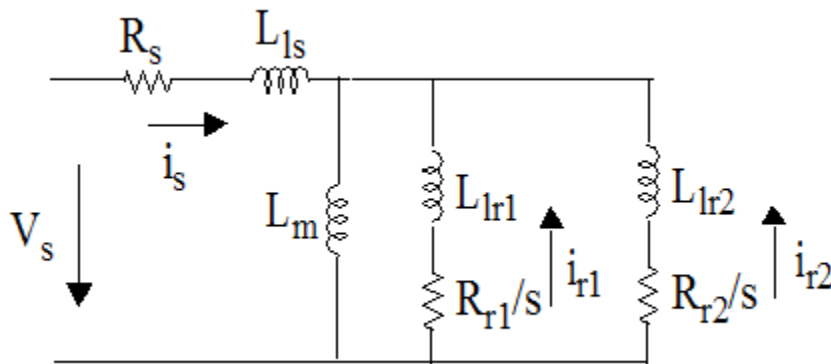
`params = power_AsymynchronousMachineParams(spec, options)` lets you specify more options (display details, draw graphs, parameter units, and graph units) in the input structure `options`.

`[params, spec2] = power_AsymynchronousMachineParams(spec, options)` returns the input manufacturer data along with some additional derived data (synchronous speed, number of pole pairs, nominal slip, starting torque, and so on) in the structure `spec2`.

`[params, spec2, errors] = power_AsymynchronousMachineParams(spec, options)` returns the relative errors between the input manufacturer data and the equivalent data obtained with the computed parameters in the structure `errors`.

Electrical Parameters of a Double-Cage Asynchronous Machine

The following figure shows the steady-state equivalent circuit of a double-cage asynchronous machine.



$R_s, L_{ls}, R_{r1}, L_{lr1}, R_{r2}, L_{lr2}, L_m$ are the seven electrical parameters of the machine. They form the fields of the `params` structure, as described in the “Output Arguments” on page 2-10 section.

The `power_AynchronousMachineParams` function computes the seven machine parameters by solving the nonlinear equations in the form $f(x) = 0$ with:

$$f_1(x) = \frac{T_n - T(s_n)}{T_n}$$

$$f_2(x) = \frac{I_n - I(s_n)}{I_n}$$

$$f_3(x) = \frac{pf - pf(s_n)}{pf}$$

$$f_4(x) = \frac{I_{st} - I(1)}{I_{st}}$$

$$f_5(x) = \frac{T_{br} - T(s_{br})}{T_{br}}$$

$$f_6(x) = \frac{T_{st} - T(1)}{T_{st}}$$

where $f = (f_1, f_2, f_3, f_4, f_5, f_6)$ and $x = (R_s, L_m, R_{r1}, L_{lr1}, R_{r2}, L_{lr2})$. The L_{ls} parameter is obtained by assuming that the stator and outer cage leakage inductances are equal ($L_{ls} = L_{lr2}$). s_n and s_{br} are the slip at nominal and breakdown torque, respectively. For details on the functions used in these equations, see [1 on page 2-15].

$T_n, I_n, pf, I_{st}, I_{br}, T_{st}$ are the standard manufacturer specifications of the machine. They form the fields of the `spec` structure, as described in the “Input Arguments” on page 2-9 section.

Input Arguments

`spec` contains the manufacturer data in a structure with the following fields.

Field	Description
<code>Vn</code>	Nominal line-to-line rms voltage (V)
<code>fn</code>	Nominal frequency (Hz)
<code>In</code>	Nominal (full load) line current (A)
<code>Tn</code>	Nominal (full load) torque (N.m)
<code>Ns, or p</code>	You can either specify the Synchronous speed (rpm) or the pole pairs
<code>Nn</code>	Nominal (full load) mechanical speed (rpm)
<code>Ist_In</code>	Starting current to nominal current ratio
<code>Tst_Tn</code>	Starting torque to full load torque ratio
<code>Tbr_Tn</code>	Breakdown torque to full load torque ratio
<code>pf</code>	Nominal power factor (%)

`options` contains additional options to the function in a structure with the following fields.

Field	Description
DisplayDetails	If set to 1, displays detailed information (results, errors between the specified and the computed data) in the MATLAB Command window. By default, this field is set to 0.
DrawGraphs	If set to 1, plots torque versus speed and stator current versus speed characteristics. By default, this field is set to 0.
units	Specifies the units of machine parameters in 'SI' or 'p.u.'. By default, this field is set to 'SI'.
graphUnits	Specifies the graph units in 'SI' or 'p.u.'. By default, this field is set to 'SI'.

Output Arguments

`params` returns the machine parameters in a structure with the following fields.

Field	Description
<code>Rs</code>	Stator resistance (ohm or pu)
<code>Lls</code>	Stator leakage inductance (H or pu)
<code>Rr1</code>	Cage 1 (inner cage) resistance (ohm or pu)
<code>Llr1</code>	Cage 1 (inner cage) leakage inductance (H or pu)
<code>Rr2</code>	Cage 2 (outer cage) resistance (ohm or pu)
<code>Llr2</code>	Cage 2 (outer cage) leakage inductance (H or pu)
<code>Lm</code>	Magnetizing inductance (H or pu)

`spec2` returns the input manufacturer data structure along with the following additional derived data fields.

Field	Description
<code>cosphi</code>	Nominal power factor (0-1)
<code>Ns</code>	Synchronous speed (rpm)
<code>p</code>	Number of pole pairs
<code>we</code>	Nominal electrical angular speed (rad/s)
<code>Vin</code>	Nominal phase voltage (V)
<code>sn</code>	Nominal (full-load) slip
<code>Ist</code>	Starting current (A)
<code>Tst</code>	Starting torque (N.m)
<code>Tbr</code>	Breakdown torque (N.m)
<code>Pn</code>	Nominal (full load) mechanical power (W)

`errors` returns the relative errors between the input manufacturer data and the equivalent data obtained with the computed parameters in a structure with the following fields.

Field	Description
<code>In</code>	Relative error with respect to specified nominal current (%)

Field	Description
Tn	Relative error with respect to specified nominal torque (%)
pf	Relative error with respect to specified nominal power factor (%)
Ist	Relative error with respect to specified starting current (%)
Ist_Iin	Relative error with respect to specified starting current to nominal current ratio (%)
Tst	Relative error with respect to specified starting torque (%)
Tst_Tn	Relative error with respect to specified starting torque to full load torque ratio (%)
Tbr	Relative error with respect to specified breakdown torque (%)
Tbr_Tn	Relative error with respect to specified breakdown torque to full load torque ratio (%)
maxError	Maximum relative error (maximum of above fields) (%)

Dialog Box

`power_AynchronousMachineParams` command opens a user interface (UI) that you can use to estimate parameters of the double-cage Asynchronous Machine block.

Specifications

Enter the manufacturer specifications. If you loaded a motor preset in the interface, the name of the MAT file is displayed in the upper section of the panel.

Compute Block Parameters

Compute the electrical parameters of the double-cage asynchronous machine and display the solution in the **Block Parameters** section. The button is grayed out when the block parameters have been estimated.

The button is reenabled when you enter new values in the **Specifications** section, to indicate that the displayed block parameters no longer correspond to the displayed specifications.

If the **Display detailed results in the command window** check box is selected, the errors between the specified manufacturer parameters and the obtained parameters are displayed in the command window. This allows you to check that the estimated RL machine parameters are satisfactory. For example, the errors obtained with the `Baldor_2,2_kw_208_V.mat` preset motor are shown below:

Asynchronous machine parameter estimation results

Parameter	Specified	Obtained	Error (%)
In (A)	8.3	8.26016	-0.48
Tn (N.m)	12.27	12.3656	0.78
Ist (A)	66.483	66.0075	-0.72
Ist/In ()	8.01	7.99106	-0.24
Tst (N.m)	43.3131	43.2625	-0.12
Tst/Tn ()	3.53	3.49861	-0.89
Tmax (N.m)	47.7303	47.9587	0.48
Tbr/Tn ()	3.89	3.87839	-0.30
pf (%)	87	86.4635	-0.62

Block Parameters

This section displays the asynchronous machine block parameters corresponding to the given manufacturer specifications. These fields are empty the first time you open the interface, or when the function failed to find a solution. The block parameters are noneditable. They can be loaded in your Asynchronous Machine block by using the **Apply to selected block** button.

Save spec to MATLAB structure

Click **Save spec to MATLAB structure** to save the motor specifications in a MATLAB structure in the base workspace.

Apply to selected block

Open your model, select the machine you want to parametrize, then click the **Apply to selected block** button to load the displayed block parameters in the block. The block is grayed out when the **Block Parameters** fields are empty. The button is enabled when a valid solution is displayed in the **Block Parameters** section.

The button has no effect on a selected block that is not an Asynchronous Machine block.

Presets

Select **New** to clear the **Block Parameters** and **Specifications** sections. The **Compute Block Parameters** and **Apply to selected block** buttons are then disabled.

Select **Open preset motor** to load in the interface a given set of manufacturer specifications. The corresponding block parameters are also loaded when they are available in the preset file.

Select **Save preset motor** to save the specifications displayed in the interface. The block parameters are also saved when available.

Options

If the **Draw graph after computation** option in the **Compute Block Parameters** submenu is selected, then when you click the **Compute Block Parameters** button the function displays the graphs of torque versus speed and stator current versus speed, corresponding to the block parameters. This option is not selected by default.

If the **Display detailed results in the command window** option in the **Compute Block Parameters** submenu is selected, then when you click the **Compute Block Parameters** button the function displays the errors between the specified manufacturer parameters and the obtained parameters are displayed in the command window. This option is selected by default.

If the **Ask me before overwriting rotor type and preset model of selected block** option in the **Apply button** submenu is selected, then when you click the **Apply to selected block** button it gives you a warning if you try to download double squirrel-cage parameters in a selected block that has a squirrel-cage or wound rotor, or if you try to overwrite a selected block using a preset model. This option is selected by default.

Examples

Consider a three-phase 400V, 110 kW asynchronous machine with the following manufacturer specification:

Data	Values
Vn	400 V

Data	Values
fn	50 Hz
In	194 A
Tn	352 N.m
Nn	2982 rpm
Ist/In	7.6
Tst/Tn	2
Tbr/Tn	3
pf	86%

To obtain the electrical parameters (R_s , L_{ls} , L_m , R_{r1} , L_{lr1} , R_{r2} , L_{lr2}) in per unit (pu) for the double cage asynchronous machine model based on this specification, enter the following commands in the MATLAB Command window:

```
spec.Vn = 400;
spec.fn = 50;
spec.In = 194;
spec.Tn = 352;
spec.Ns = 3000;
spec.Nn = 2982;
spec.Ist_In = 7.6;
spec.Tst_Tn = 2;
spec.Tbr_Tn = 3;
spec.pf = 86;
```

```
options.DisplayDetails = 1;
options.units = 'p.u.';
```

```
[params,spec2,errors] = power_AynchronousMachineParams(spec,options)
```

You should get the following results:

Asynchronous machine parameter estimation results

Parameter	Specified	Obtained	Error (%)
In (A)	194	193.991	-0.00
Tn (N.m)	352	352.023	0.01
Ist (A)	1474.4	1474.35	-0.00
Ist/In ()	7.6	7.6001	0.00
Tst (N.m)	704	703.8	-0.03
Tst/Tn ()	2	1.9993	-0.03
Tmax (N.m)	1056	1056	0.00
Tbr/Tn ()	3	2.99981	-0.01
pf (%)	86	85.9949	-0.01

```
params =
```

```
Rs: 0.0303
Lls: 0.0506
Lm: 1.9066
Llr1: 0.0868
Rr1: 0.0056
```

```
Llr2: 0.0506
Rr2: 0.0762
```

```
spec2 =
```

```
    fn: 50
    In: 194
    Tn: 352
    Ns: 3000
    Nn: 2982
    Ist_In: 7.6000
    Tst_Tn: 2
    Tbr_Tn: 3
    pf: 86
    Vn: 400
    p: 1
    cosphi: 0.8600
    we: 314.1593
    Vin: 230.9401
    sn: 0.0060
    Ist: 1.4744e+003
    Tst: 704
    Tbr: 1056
    Pn: 1.0992e+005
```

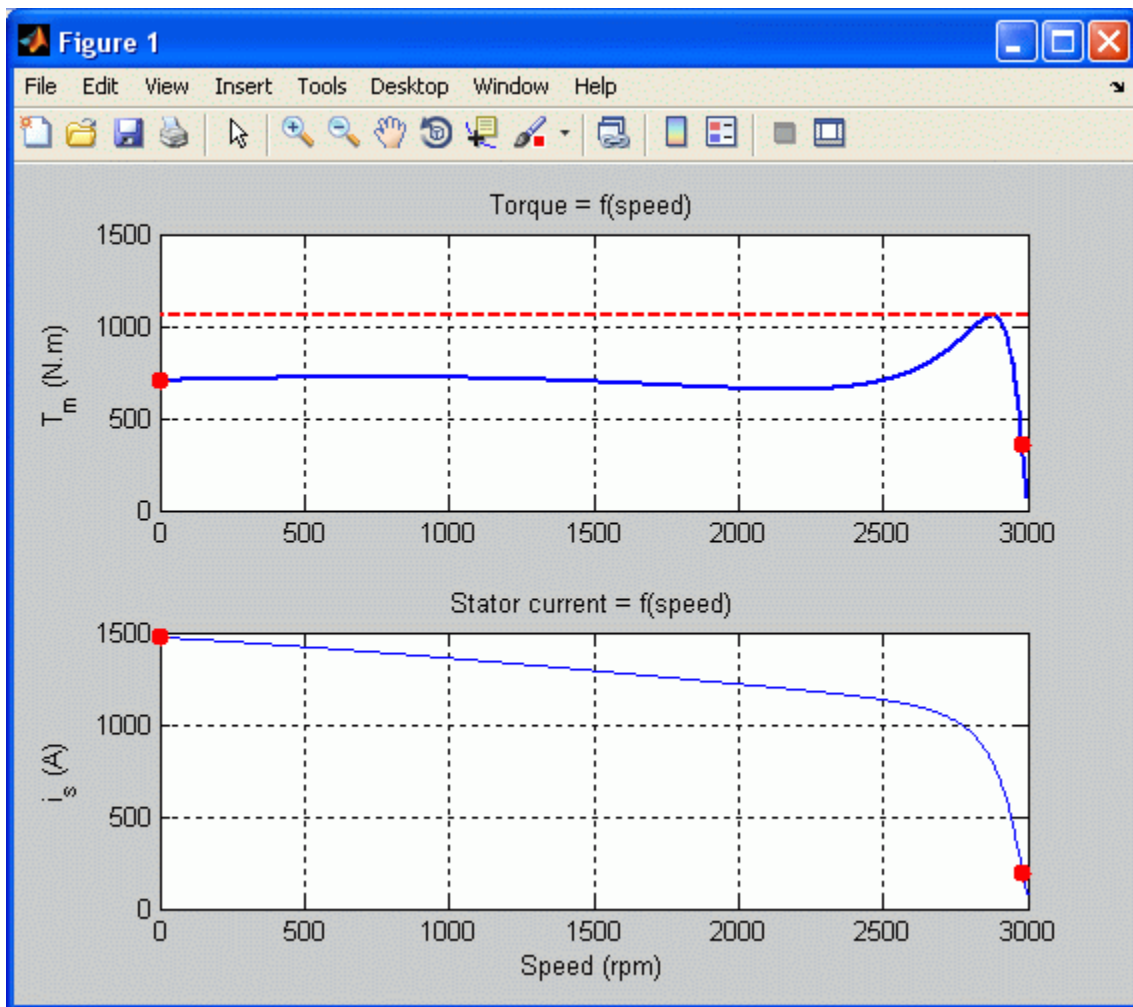
```
errors =
```

```
    In: -0.0048
    Tn: 0.0066
    pf: -0.0059
    Ist: -0.0035
    Ist_In: 0.0013
    Tst: -0.0284
    Tst_Tn: -0.0349
    Tbr: 2.1990e-004
    Tbr_Tn: -0.0063
    maxError: 0.0349
```

To draw graphs of torque versus speed and stator current versus speed, enter the following commands:

```
options.DrawGraphs = 1;
options.DisplayDetails = 0;
params = power_AsynchronousMachineParams(spec,options);
```

The following figure shows the resulting graph:



References

- [1] Pedra, Joaquin, "On the Determination of Induction Motor Parameters From Manufacturer Data for Electromagnetic Transient Programs." *IEEE Transactions on Power Systems*. Vol. 23, Number 4, 2008, pp. 1709-1718.

power_cableparam

Compute RLC parameters and frequency-dependent model parameters of a group of cables

Syntax

```
power_cableparam
DATA = power_cableparam('new')
DATA = power_cableparam(CableData)
```

Description

`power_cableparam` opens the Power Cable Parameters app with the default cable parameter values that the Simscape Electrical Specialized Power Systems software provides.

`DATA = power_cableparam('new')` returns a structure variable with default cable geometry parameters. You can use `DATA` as a template variable to configure a new cable geometry and to compute the cable parameters.

`DATA = power_cableparam(CableData)` computes the cable parameters for the geometric cable parameters given in the `CableData` structure. The function computes the `R`, `L`, `C`, and `G` fields of `DATA` structure as N -by- N matrices where N is the number of phases defined in the cable arrangement. The `WB` field of the `DATA` structure variable containing the frequency-dependent model parameters for the given frequency-range.

If you specify the `DATA` structure in the format of R2022a, the function assumes that a cable consists of an inner copper phase conductor with an outer screen conductor, using cross-linked polyethylene (XLPE) insulator material. The `R`, `L`, `C`, and `Z` parameters are then computed as in R2022a. The `G` and `WB` parameters are not computed.

Input Arguments

CableData — Cable geometry parameters characteristics

structure

Conductor and insulator characteristics for a set of cables, specified as a structure with these fields:

Field	Description
<code>configuration</code>	Specify the configuration of the cable arrangement: 'underground', 'aerial', 'pipe'.
<code>cables</code>	Structure specifying the cable arrangement geometry.
<code>types</code>	Structure specifying the geometrical and electrical parameters of the cable types.
<code>pipe</code>	Structure specifying the parameters of the pipe.
<code>crossbondTheSheaths</code>	Set to 1 to cross bond the sheaths. See the documentation of the Power Cable Parameters app for more information on how to use this parameter.

Field	Description
frequency	The frequency, in hertz, at which to evaluate the RLC cable arrangement parameters.
frequencyRange	Specify a vector of three elements [X1,X2,N] to define a frequency vector in the <code>logspace(X1, X2,N)</code> format, consisting of N logarithmically equally spaced points between 10^{X1} and 10^{X2} .
length	Length of the cable arrangement in meters.
groundResistivity	Ground resistivity in ohm-meters. Set to 0 to specify perfectly conducting ground.
comments	Comments related to the cable arrangement.

The `cables` field of `CableData` is a structure with the fields in this table. The length of the `cables` structure corresponds to the number of cables in your cable arrangement.

Field	Description
Vdist	When the <code>configuration</code> field is set to aerial or underground, specify the vertical position of the center of the cable in meters. When the <code>configuration</code> field is set to pipe specify the cable position relative to the center of the pipe in meters.
Hdist	When the <code>configuration</code> field is set to aerial or underground, specify the horizontal position of the center of the cable in meters. When the <code>configuration</code> field is set to pipe specify the position angle of the cable inside the pipe in meters.
Rout	Specify the external radius of the cable.
type	Specify the cable type as an integer between 1 and Number of Types.
phase	Associate each conductor of the cable to an electrical phase of the network.

The `types` field of `CableData` is a structure with the fields in this table. The length of the `types` structure corresponds to the number of cable types that are defined in your cable arrangement.

Field	Description
conductors	Geometric and electric parameters of the cable

The `conductors` field of `CableData.types` contains a structure with the fields in this table. The length of the conductor structure corresponds to the number of conductors of each cable type in your cable arrangement.

Field	Description
R_in	Inside radius of the conductor
R_out	Outside radius of the conductor
Rho	Conductor resistivity
MUE	Conductor relative permeability
MUE_IN	Surrounding insulation relative permeability

Field	Description
EPS_IN	Surrounding insulation relative permittivity
LFCT_IN	Surrounding insulation loss factor

pipe is a structure with the fields in this table.

Field	Description
R_in	Internal radius of the metallic pipe in meters
R_out	External radius of the metallic pipe
R_ext	External radius of the outer insulator
V_dpth	Depth of center of the pipe below the ground level
Rho	Pipe resistivity
MUE	Metallic pipe relative permeability
MUE_in	Inside insulation relative permeability
EPS_IN	Inside insulation relative permeability
LFCT_IN	Inside insulation loss factor
MUE_out	Outside insulation relative permeability
EPS_out	Outside insulation relative permeability
LFCT_out	Outside insulation loss factor
phase	Phase number connected to the metallic pipe

Output Arguments

DATA — Cable parameters

structure

Cable parameters, returned as a structure with these fields:

Field	Description
R	Resistance per unit length matrix of the cable arrangement
L	Inductance per unit length matrix of the cable arrangement
C	Capacitance per unit length matrix of the cable arrangement
G	Admittance per unit length matrix of the cable arrangement
WB	Frequency dependent parameters of the cable arrangement

See Also

Power Cable Parameters | `power_lineparam`

Topics

“Four Coupled 66-kV Cables”

power_customize

Prepare a Simscape Electrical Specialized Power Systems block for customization

Syntax

`power_customize`

Description

`power_customize` lets you customize Simscape Electrical Specialized Power Systems blocks to:

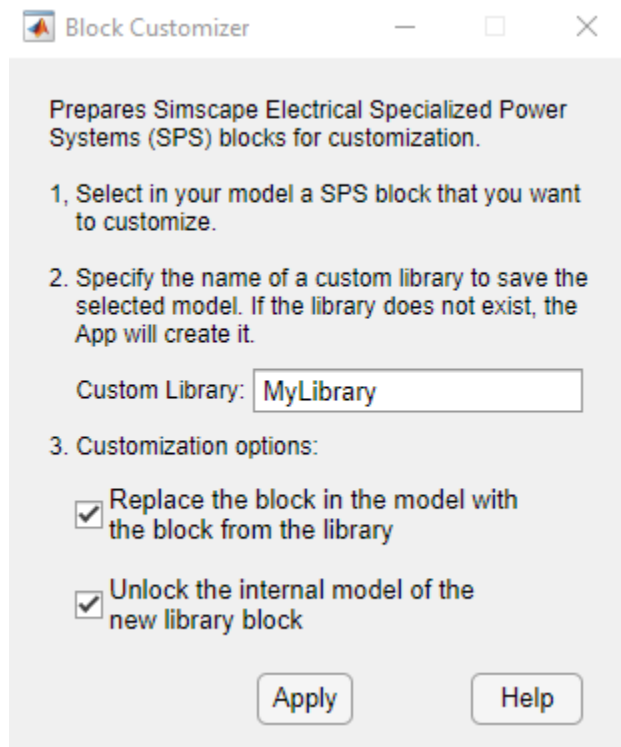
- Model behaviors that are not provided in the library blocks
- Build more advanced models
- Develop simplified models
- Provide additional user interfaces to existing blocks

Simscape Electrical Specialized Power Systems blocks you customize cannot have broken or inactive library links. In other words, you cannot disable or break the library link of a block and start customizing the block directly in your model. If you want to modify a block, you make a copy of the block in a custom library, break the block link to the product library, and start doing your modifications in the custom library. The function `power_customize` automates these steps and helps you to prepare a Simscape Electrical Specialized Power Systems block for customization.

You can call the function from the command line with `power_customize`, or you can click **Customize SPS blocks** in the **Tools** tab of the Powergui block to open the dialog box.

In the typical workflow, you open the `power_customize` dialog box, select the block you want to customize in a model, specify the name of a custom library to save the block in, and apply your changes. Then you customize the block in your library. The dialog box also has the option to replace the block in your model with a library link to the new custom block.

Dialog Box



Custom Library

Specify the name of a custom library to save the selected block to. If the specified library does not exist, `power_customize` creates it. The block is copied to the library and the name appended with the text (custom) when you apply the changes. The library opens, and you can then customize the block in your library.

The default value is `MyLibrary`.

Replace the block in the model with the block from the library

Replace the selected block in your model with a copy of the block (i.e., a library link) from the specified library. When you apply your changes, the function replaces the block in your model with the library link to the new custom block. When `power_customize` creates the library block and the link in the model, it appends the block name with the text (custom). If this parameter is cleared, the function does not replace the block in your model. In this situation, if you apply changes several times, the function creates several instances of the selected block in the specified library. You can customize these libraries with different changes.

This parameter is selected by default.

Unlock the internal model of the new library block

Simscape Electrical Specialized Power Systems blocks use internal functions that control the mask behavior and the internal Simulink model. For example, the function replaces the internal Simulink model of a Synchronous Machine block with a discrete model when the **Simulation type** parameter of the Powergui block is set to **Discrete** or to a phasor model when set to **Phasor**.

Select this parameter to enable customizing of the internal Simulink model of the selected block. The internal model of the block is then no longer controlled by the Powergui **Simulation type** parameter.

If you do not plan to customize the internal Simulink model but instead plan to rework other parts of the block, you can clear this parameter. The internal model of the block remains under control of the **Simulation type** parameter of the Powergui block.

This parameter is selected by default.

Examples

The “24-hour Simulation of a Vehicle-to-Grid (V2G) System” example model, which shows a 24-hour simulation of a vehicle-to-grid system, contains a customized Asynchronous Machine block (under the Load subsystem). The electrical internal model (a phasor model) was optimized to provide faster simulation in phasor mode. This custom block and the V2Glib custom library were prepared for customization using power_customize.

Version History

Introduced in R2015a

power_fftscope

Fourier analysis of simulation data signals

Syntax

```
FFTDATA = power_fftscope(ScopeData)
FFTDATA = power_fftscope(FFTDATA)
power_fftscope(ScopeData)
power_fftscope(FFTDATA)
```

Description

`FFTDATA = power_fftscope(ScopeData)` returns the FFT results for the signal saved in the `ScopeData` structure-with-time.

`FFTDATA = power_fftscope(FFTDATA)` uses the `FFTDATA` structure as a template variable to modify analysis settings and signal selection, and to perform FFT analysis. The `power_fftscope` function ignores any user-defined `FFTDATA` fields that are not listed in the `FFTDATA` structure.

`power_fftscope(ScopeData)` plots the FFT analysis results for the last simulation cycle of the signal saved in the `ScopeData` variable.

`power_fftscope(FFTDATA)` plots the FFT analysis results for the signal options specified in the `FFTDATA` structure.

Input Arguments

ScopeData — Signal you want to perform Fourier analysis on

structure with time (default)

Signal you want to perform the Fourier analysis on, specified as a structure with time.

FFTDATA — Signal options

structure (default)

Signal options, specified as a structure with these fields:

Field	Description
time	The time vector of the simulation data signal saved in the <code>ScopeData</code> variable.
signals	The signals saved in the <code>ScopeData</code> variable.
blockName	The name of the Scope block associated with the <code>ScopeData</code> variable.
input	The input signal of the selected simulation data variable.
signal	The index of the selected input signal specified by the <code>Input</code> field.

Field	Description
startTime	The start time of the FFT window.
cycles	The number of cycles of the FFT window.
fundamental	The fundamental frequency of the analyzed signal.
maxFrequency	The maximum frequency evaluated by the FFT analysis.
THDmaxFrequency	The maximum frequency for the total harmonic distortion (THD) calculation. Set the value to <code>inf</code> to calculate the THD at the Nyquist frequency.
FFTdata	The analyzed signal (FFT window data).
THDbase	The base used to compute the THD. Set to <code>fund</code> to normalize the THD with respect to fundamental value. Set the THDbase to <code>DC</code> to normalize the THD with respect to the DC component value.
freqAxis	The type of frequency axis of the FFT analysis plot. Set to <code>hertz</code> to display the frequency axis in hertz. Set to <code>harmonicorder</code> to display the frequency axis in harmonic orders.
mag	The computed magnitude of FFT.
phase	The computed phase of FFT.
freq	The frequency vector.
THD	The computed THD for the analyzed signal. The THD calculation includes all the inter-harmonics of the selected input signal.
samplingTime	The sampling time of the selected input signal.
samplePerCycle	The number of samples per cycle of the selected input signal.
DCcomponent	The DC component value of the selected input signal.
magFundamental	The fundamental component value of the selected input signal.

Output Arguments

FFTDATA — FFT results for signal

structure

FFT results for the signal saved in the `ScopeData` argument, returned as a structure with these fields:

Field	Description
time	The time vector of the simulation data signal saved in the <code>ScopeData</code> variable.

Field	Description
signals	The signals saved in the ScopeData variable.
blockName	The name of the Scope block associated with the ScopeData variable.
input	The input signal of the selected simulation data variable.
signal	The index of the selected input signal specified by the Input field.
startTime	The start time of the FFT window.
cycles	The number of cycles of the FFT window.
fundamental	The fundamental frequency of the analyzed signal.
maxFrequency	The maximum frequency evaluated by the FFT analysis.
THDmaxFrequency	The maximum frequency for the total harmonic distortion (THD) calculation. Set the value to <code>inf</code> to calculate the THD at the Nyquist frequency.
FFTdata	The analyzed signal (FFT window data).
THDbase	The base used to compute the THD. Set to <code>fund</code> to normalize the THD with respect to fundamental value. Set the THDbase to <code>DC</code> to normalize the THD with respect to the DC component value.
freqAxis	The type of frequency axis of the FFT analysis plot. Set to <code>hertz</code> to display the frequency axis in hertz. Set to <code>harmonicorder</code> to display the frequency axis in harmonic orders.
mag	The computed magnitude of FFT.
phase	The computed phase of FFT.
freq	The frequency vector.
THD	The computed THD for the analyzed signal. The THD calculation includes all the inter-harmonics of the selected input signal.
samplingTime	The sampling time of the selected input signal.
samplePerCycle	The number of samples per cycle of the selected input signal.
DCcomponent	The DC component value of the selected input signal.
magFundamental	The fundamental component value of the selected input signal.

Version History

Introduced before R2006a

See Also

FFT Analyzer

power_flicker

Perform statistical analysis of instantaneous flicker level signal

Syntax

power_flicker

Description

power_flicker opens a tool that computes the short-term flicker severity (Pst) and long-term flicker severity (Plt) and displays the cumulative probability function (CPF) of an instantaneous flicker level signal (S5 signal).

Note You must have a Control System Toolbox license to use the power_flicker function.

Dialog Box

Load

Load the S5 signal into the tool. Choose between **From file** and **From workspace**. The default value is **From workspace**.

10 minutes Short-term interval (IEC standard)

When this parameter is selected, the short-term interval observation period is set to 10 minutes (600 seconds). This setting corresponds to the time specified in the IEC 61000-4-15 standard. This parameter is selected by default.

Short-term interval Tshort (s)

Specify the short-term interval observation period, in seconds. This parameter is dimmed when the 10 minutes Short-term interval parameter (IEC standard) parameter is selected. The default value is 600.

Long-term interval (N x Tshort) N:

Specify the number of 10-minute intervals used by the tool to measure the long-term flicker severity (Plt). The default value is 1.

Start time (s)

Specify the time, in seconds, of the beginning of the observed period. The default value is 0.

Number of classes

Specify the number of classes, which are used by the tool to compute the cumulative probability function of the instantaneous flicker level. The default value is 256.

Compute

Compute the percentiles P0.1s, P1s, P3s, P10s, and P50s, which are the flicker levels exceeded for 0.1%, 1%, 3%, 10%, and 50% of the time during the observed period. Computes the short-term flicker severity (Pst) and long-term flicker severity (Plt) and displays the cumulative probability function (CPF) of the S5 signal.

Version History

Introduced in R2015b

See Also

Digital Flickermeter

power_hysteresis

View and edit hysteresis characteristic for saturable core of Saturable Transformer blocks

Syntax

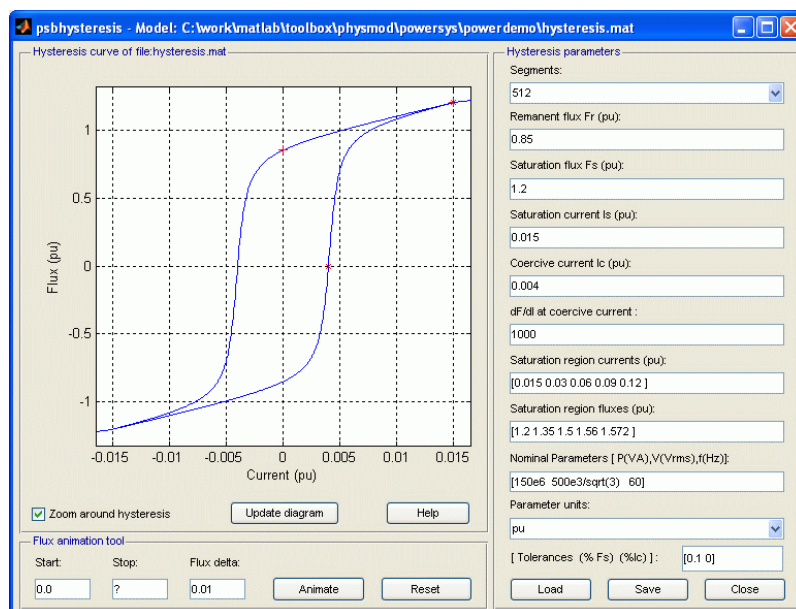
```
power_hysteresis
hparam = power_hysteresis(matfile)
```

Description

The `power_hysteresis` command opens a graphical user interface (GUI) that allows you to view and edit a hysteresis characteristic for the saturable core of the Multi-Winding Transformer, the Saturable Transformer, the Three-Phase Two Windings Transformer, and the Three-Phase Three Windings Transformer blocks. Hysteresis characteristic includes the saturation region located at the limits of the hysteresis loop. This GUI can also be activated from the Powergui block dialog box.

`hparam = power_hysteresis(matfile)` returns a structure variable `hparam` with hysteresis parameter values defining the hysteresis characteristic of the specified MAT file.

A default hysteresis characteristic is first displayed when you open the GUI, but you can build as many hysteresis characteristics as you want and save them in different MAT file names. You can use the same characteristic for all of your transformer blocks, or you can use different ones for each transformer block in the circuit. You need to select the **Simulate hysteresis** check box in the masks of the transformer blocks and specify a MAT file to be used by the model.



Hysteresis Parameters

Segments

In the pull-down menu, specify the number of linear segments used to define the right side of the hysteresis loop. The left side of the loop is the symmetric image of the right side.

Remanent flux F_r

Specify the remanent flux point of the hysteresis characteristic (flux at zero current). It is identified by a + mark in the plot.

Saturation flux F_s

Specify the saturation flux point where the hysteresis loop becomes a single-valued saturation curve. It is identified by a * mark in the plot.

Saturation current I_s

Specify the saturation current point where the hysteresis loop becomes a single-valued saturation curve. The saturation region is defined by the **Saturation region currents** parameter.

Coercive current I_c

Specify the coercive current point of the hysteresis characteristic.

 dF/dI at coercive current

Set the slope of the flux at the coercive current point (current at zero flux).

Saturation region currents

Specify the vector of current values that define the saturation characteristic. The number of specified points must be the same as for the **Saturation region fluxes** parameter. You only need to specify the positive part of the characteristic.

Saturation region fluxes

Specify the vector of flux values that define the saturation characteristic. The number of specified points must be the same as for the **Saturation region currents** parameter. You only need to specify the positive part of the characteristic.

Nominal Parameters

Specify the nominal parameters (nominal power in VA, nominal voltage of winding 1 in volts RMS, and nominal frequency in Hz) used in the conversion of the hysteresis parameters.

Parameter units

Convert the fluxes and currents that define the hysteresis characteristic from SI to pu or from pu to SI.

Load

Load an existing hysteresis characteristic from a MAT file.

Save

Save current hysteresis characteristic into a MAT file.

Close

Close the hysteresis GUI window.

Hysteresis Curve**Zoom around the hysteresis**

If selected, zoom the plot around the hysteresis curve. The default is selected.

Update diagram

When the parameters are entered, you can click **Display** to visualize the hysteresis characteristic.

Flux Animation Tool

The Flux Animation tool can be used to visualize how the simulation of the hysteresis is performed by Simscape Electrical Specialized Power Systems software. This is an optional tool that is not necessary

for the model parameterization. The initial trajectory will be calculated according to the defined hysteresis characteristic.

The model assumes that the last reversal point before starting flux is located on the major loop. The operating point will travel till the defined stop flux.

Start

Specify the starting flux.

Stop

Specify the flux at which the flux animation will stop.

Flux delta

Specify flux increment (in pu, or in SI) that is used to go from start flux to stop flux.

Animate

Start the Flux Animation tool.

Reset

Reset the Flux Animation tool.

Tolerances Tool

The Tolerances tool is an advanced tool mainly used to minimize the generation of superfluous very small internal loops or new trajectories because they have little effect and they consume computer memory space (the model can memorize at any time up to 50 embedded minor loops).

The TOL_F parameter is the tolerance value used to detect whether, after a flux reversal, the operating point remains on the same minor loop or a new embedded loop is created. The smaller the value, the lesser is the effect on the normal trajectory behavior. The bigger the value, the lesser is the generated number of embedded minor loops.

Finally, when the distance between the I coordinate of the actual point of reversal and the penultimate one is less than TOL_I, then evolution within these two points will follow a line segment instead of a loop.

See Also

Powergui

power_init

Set initial state values of model built with Simscape Electrical Specialized Power Systems software

Syntax

```
[x0,states] = power_init(sys)
power_init(sys,'states')
power_init(sys,'ForceToZero')
power_init(sys,'ResetToDefault')
power_init(sys,'x0')
power_init(sys,'look')
power_init(sys,'reset')
power_init(sys,'steady')
```

Description

`power_init` deals with two types of state variables: the Simulink State variables and the Electrical State variables. The Simulink States consists of the entire set of state variables of a model (including the Electrical states) seen by Simulink software. The Electrical States are the particular Simulink states that defines the capacitor initial voltages and inductor initial currents of the Simscape Electrical Specialized Power Systems blocks of your model.

`[x0,states] = power_init(sys)` returns the initial states values and state names of your model.

`power_init(sys,'states')` displays in a two-column format the initial state values and names of your model.

`power_init(sys,'ForceToZero')` sets to zero the initial states your model. It uses the **Data import/export initial state** option from the **Simulation > Model Configuration Parameters** menu in the **Simulink Editor** to specify the initial state values. Use the **ResetToDefault** option of `power_init` to reset the Simulink configuration parameters to its previous settings.

`power_init(sys,'ResetToDefault')` resets the initial state of Simulink configuration parameters to the settings prior to the use of the **ForceToZero** option of `power_init`. This option has no effect on system if the **ForceToZero** option is not used before.

`power_init(sys,'x0')` sets the initial state values equal to the vector specified by `x0`. The ordering of the states variables is given by the **state** option of `power_init`. This option uses the **Data import/export initial state** option from the **Simulation > Model Configuration Parameters** menu in the **Simulink Editor** to specify the initial state values.

`power_init(sys,'look')` displays in a two-column format the electrical initial states names and values of your model.

`power_init(sys,'reset')` resets the electrical initial states of `sys` to zero.

`power_init(sys,'steady')` sets the electrical initial states of `sys` to simulate the electrical circuit in the steady state.

See Also

`power_analyze` | Powergui

power_initstates

Compute and modify initial values of voltages and currents of model

Syntax

```
sps = power_initstates('sys')
```

Description

`sps = power_initstates('sys')` returns the initial values of the electrical state variables (inductor currents and capacitor voltages) of a model, and modifies the initial electrical states to start the simulation from any initial conditions. It returns a structure with these fields:

Field	Description
<code>circuit</code>	The name of the model
<code>States</code>	The names of the electrical state variables
<code>x0</code>	The current initial states vector used for the simulation of the model
<code>x0_steady</code>	The computed initial states that ensure a steady-state simulation
<code>x0_blocks</code>	The default initial states values defined in blocks
<code>DependentStates</code>	The names of the dependent states, if any
<code>x0DependentStates</code>	Initial state values of dependent states
<code>UseInitialStatesFrom</code>	The initial state for the simulation, expressed as a character vector ('block', 'steady', 'zero'). If set to 'block', the default initial states are used. If set to 'steady', the computed initial states that ensure a steady-state simulation are used. If set to 'zero', the initial states are forced to 0.

See Also

Measurements and States Analyzer

power_lineparam

Compute RLC parameters of overhead transmission line from its conductor characteristics and tower geometry

Syntax

```
DATA = power_lineparam('new')
DATA = power_lineparam(DATA)
DATA = power_lineparam('MYLINEDATA')
power_lineparam(DATA, 'BLK')
```

Description

`DATA = power_lineparam('new')` returns a structure variable with the default line geometry parameters.

`DATA = power_lineparam(DATA)` computes the RLC line parameters for the geometric line parameters in the `DATA` structure. The returned structure contains both geometric data and computed RLC line parameters.

`DATA = power_lineparam('MYLINEDATA')` computes the RLC line parameters for the geometric line parameters in the specified file. 'MYLINEDATA' is the name of a MAT-file. The MAT-file must contain a structure variable of the same format as the `DATA` variable.

`power_lineparam(DATA, 'BLK')` uploads the RLC line parameters in the specified PI Section Line, PI Section Cable, Distributed Parameters Line, or Three-Phase PI Section Line block. 'BLK' is a block path name. You can also specify a file name for `DATA`.

Examples

Simple Line Example With Two Conductors

The example uses a simple line consisting of two conductors spaced by 1 meter at an average height of 8 meters above a perfect ground (ground resistivity $\rho_g = 0$). The two conductors are solid aluminum conductors (resistivity $\rho_c = 28.3 \cdot 10^{-9} \Omega \cdot \text{m}$ at 20°C) that have a 15-mm diameter.

The DC resistance per km of each conductor is:

$$r = \frac{\rho_c \cdot l}{A} = \frac{28.3 \times 10^{-9} \times 1000}{\pi \cdot \left(\frac{15 \times 10^{-3}}{2}\right)^2} = 0.1601 \Omega/\text{km}$$

As ground is supposed to be perfect, the off-diagonal terms of the R matrix are zero and the diagonal terms represent the conductor resistances:

$$R = \begin{bmatrix} 0.1601 & 0 \\ 0 & 0.1601 \end{bmatrix} \Omega/\text{km}$$

For the solid conductors, the GMR is:

$$GMR = r \cdot e^{-\mu_r/4} = \frac{1.5}{2} e^{-1/4} = 0.5841 \text{ cm}$$

The self- and mutual-inductances are computed as follows. The ΔL correction terms are ignored because ground resistivity is zero.

$$L_{11} = L_{22} = \frac{\mu_0}{2\pi} \cdot \log \frac{2h_i}{GMR} = 2 \times 10^{-4} \log \frac{2 \times 8}{0.5841 \times 10^{-2}} = 1.583 \times 10^{-3} \text{ H/km}$$

$$L_{12} = L_{21} = \frac{\mu_0}{2\pi} \cdot \log \frac{D_{12}}{d_{12}} = 2 \times 10^{-4} \log \frac{\sqrt{16^2 + 1}}{1} = 0.5549 \times 10^{-3} \text{ H/km}$$

$$L = \begin{bmatrix} L_{11} & L_{12} \\ L_{21} & L_{22} \end{bmatrix} = \begin{bmatrix} 1.583 \times 10^{-3} & 0.5549 \times 10^{-3} \\ 0.5549 \times 10^{-3} & 1.583 \times 10^{-3} \end{bmatrix} \text{ H/km}$$

The self- and mutual-capacitances are computed as follows:

$$P_{11} = P_{22} = \frac{1}{2\pi\epsilon_0} \log \frac{2h}{r_i} = 1.7975 \times 10^7 \log \frac{2 \times 8}{0.75 \times 10^{-2}} = 1.3779 \times 10^8 \text{ km / F}$$

$$P_{12} = P_{21} = \frac{1}{2\pi\epsilon_0} \log \frac{D_{12}}{d_{12}} = 1.7975 \times 10^7 \log \frac{\sqrt{16^2 + 1}}{1} = 4.9872 \times 10^7 \text{ km / F}$$

$$C = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix}^{-1} = \begin{bmatrix} 8.352 \times 10^{-9} & -3.023 \times 10^{-9} \\ -3.023 \times 10^{-9} & 8.352 \times 10^{-9} \end{bmatrix} \text{ F/km}$$

You first create a template **DATA** structure to define the line geometry and the type of conductors of this line. At the MATLAB prompt, type

```
DATA = power_lineparam('new');
```

Then set the following fields with specific values:

```
DATA.frequency = 50;
DATA.groundResistivity = 0;
DATA.Geometry.NPhaseBundle = 2;
DATA.Geometry.NGroundBundle = 0;
DATA.Geometry.PhaseNumber = [1,2];
DATA.Geometry.X = [0,1];
DATA.Geometry.Ytower = [8,8];
DATA.Geometry.Ymin = [8,8];
DATA.Geometry.ConductorType = [1,1];
% define conductor Type 1:
DATA.Conductors.Diameter = [1.5];
DATA.Conductors.ThickRatio = [0.5];
DATA.Conductors.GMR = [0.5841];
DATA.Conductors.Res = [0.1601];
DATA.Conductors.Mur = [1];
DATA.Conductors.Nconductors = [1];
DATA.Conductors.BundleDiameter = [0];
DATA.Conductors.AngleConductor1 = [0];
DATA.Conductors.skinEffect = 'no';
```

Compute the RLC line parameters with the command:

```
DATA = power_lineparam(DATA);
disp(DATA.R),disp(DATA.L),disp(DATA.C)
```

You can also vary the ground resistivity and the frequency to observe their impact on the resistance and inductance of the conductor and of the ground return.

The following code varies the ground resistivity from zero to 10000 $\Omega\cdot\text{m}$:

```
Rground = [0,10,100,10000]
for k=1:4
    DATA.groundResistivity = Rground(k);
    DATA = power_lineparam(DATA);
    ConductorRS_RM(k) = DATA.R(1,1)-DATA.R(1,2);
    GroundRm(k) = DATA.R(1,2);
    ConductorLs_Lm(k) = DATA.L(1,1)-DATA.L(1,2);
    GroundLm(k) = DATA.L(1,2);
end
```

You should get the values listed in the following table. The expressions Rs-Rm and Ls-Lm represent respectively the resistance and the inductance of the conductor, whereas Rm and Lm are the resistance and the inductance of the ground return.

Impact of Ground Resistivity (Frequency = 50 Hz; Skin Effect Not Included)

Ground Resistivity ($\Omega\cdot\text{m}$)	Conductor Rs-Rm (Ω/km)	Ground Rm (Ω/km)	Conductor Ls-Lm (mH/km)	Ground Lm (mH/km)
0	0.1601	0	1.028	0.5549
10	0.1601	0.04666	1.029	1.147
100	0.1601	0.04845	1.029	1.370
10 000	0.1601	0.04925	1.029	1.828

When the ground resistivity varies in a normal range (between 10 $\Omega\cdot\text{m}$ for humid soil and 10 000 $\Omega\cdot\text{m}$ for dry, rocky ground), the ground resistance remains almost constant at 0.05 Ω/km , whereas its inductance increases from 1.15 mH/km to 1.83 mH/km.

Using the following code, repeat the computation with different frequencies ranging from 0.05 Hz to 50 kHz and with the conductor skin effect modeled for a ground resistivity of 100 Ohms-m.

```
DATA.groundResitvity = 100;
DATA.Conductors.skinEffect = 'yes';
Frequencies = [0.05,50,500,5000,50000]
for k=1:5
    DATA.frequency = Frequencies(k);
    DATA = power_lineparam(DATA);
    ConductorRS_RM(k) = DATA.R(1,1)-DATA.R(1,2);
    GroundRm(k) = DATA.R(1,2);
    ConductorLs_Lm(k) = DATA.L(1,1)-DATA.L(1,2);
    GroundLm(k) = DATA.L(1,2);
end
```

Impact of Frequency (Ground Resistivity = 100 $\Omega\cdot\text{m}$; with Conductor Skin Effect)

Frequency (Hz)	Conductor Rs-Rm (Ω/km)	Ground Rm (Ω/km)	Conductor Ls-Lm (mH/km)	Ground Lm (mH/km)
0.05	0.1601	4.93e-5	1.029	2.058
50	0.1606	0.04844	1.029	1.370
500	0.2012	0.4666	1.022	1.147
5000	0.5442	4.198	0.9944	0.9351
50 000	1.641	32.14	0.9836	0.7559

This table shows that frequency has a very large impact on ground resistance, but a much lower influence on ground inductance. Because of skin effect in the ground, when frequency increases, the ground current flows closer to the surface, reducing the equivalent section of the ground conductor and thereby increasing its resistance. As ground current travels at a lower depth at high frequencies, the loop inductance of the conductor plus ground return (or the self-inductance, L_s) decreases.

Because of conductor skin effect, frequency has a noticeable impact on conductor resistance from a few hundreds of hertz, but a negligible impact on conductor inductance. At nominal system frequency (50 Hz or 60 Hz), the increase of conductor resistance with respect to DC resistance (0.1601 Ω/km) is only 0.3%.

The second example corresponds to a 500 kV, three-phase, double-circuit line.

Power is transmitted over six phase conductors forming the two three-phase circuits. The line is protected against lightning by two ground wires. The phase conductors use bundles of three subconductors. Subconductors are located at the top of an equilateral triangle of 50-cm sides, corresponding to a 57.735 cm bundle diameter. This line configuration corresponds to the one shown in the figure “Configuration of a Three-Phase Double-Circuit Line” on page 2-41.

Phase numbering has been set to obtain the line parameters of the three-phase line equivalent to the two circuits connected in parallel. In the **Compute** section, click **RLC Line Parameters** to display the R, L, and C matrices and sequence parameters.

The positive-sequence and zero-sequence parameters of the transposed line are displayed in the **Computed Parameters** section in the R10 and L10 vectors:

$$R1 = 0.009009 \Omega/\text{km} \quad R0 = 0.2556 \Omega/\text{km} \quad (2-1)$$

$$L1 = 0.4408 \text{ mH}/\text{km} \quad L0 = 2.601 \text{ mH}/\text{km} \quad (2-2)$$

$$C1 = 25.87 \text{ nF}/\text{km} \quad C0 = 11.62 \text{ nF}/\text{km} \quad (2-3)$$

You can also get the parameters of the two individual circuits and have access to the six phase conductors. Change the phase numbers of conductors p4, p5, and p6 (circuit 2) to 6, 5, and 4, respectively. The positive-sequence, zero-sequence, and mutual zero-sequence parameters of the transposed line are:

$$R1 = 0.01840 \Omega/\text{km} \quad R0 = 0.2649 \Omega/\text{km} \quad R0m = 0.2462 \Omega/\text{km} \quad (2-4)$$

$$L1 = 0.9296 \text{ mH}/\text{km} \quad L0 = 3.202 \text{ mH}/\text{km} \quad L0m = 2.0 \text{ mH}/\text{km} \quad (2-5)$$

$$C1 = 12.57 \text{ nF/km} \quad C0 = 7.856 \text{ nF/km} \quad C0m = -2.044 \text{ nF/km} \quad (2-6)$$

As the line is symmetrical, the positive- and zero-sequence parameters for circuit 2 are identical to the parameters of circuit 1.

Input Arguments

'new' — Structure or file of RLC line parameters

string | structure

Structure containing RLC line parameters or file name for a file that contains a

Output Arguments

DATA — Parameters of overhead transmission line

structure

Structure containing RLC parameters.

If you generate a structure variable with the default line geometry parameters by using the argument 'new', the function returns the structure DATA with the fields in this table:

Field	Description
comments	Specify user comments related to the line parameters.
units	The units of the line parameters. Set to <code>metric</code> to specify the conductor diameter, GMR, and bundle diameter in centimeters and the conductor position in meters. Set to <code>English</code> to specify the conductor diameter, GMR, and bundle diameter in inches and the conductor position in feet. The default is <code>metric</code> .
frequency	The frequency, in hertz, to evaluate the RLC line parameters. The default is 60.
groundResistivity	The ground resistivity, in ohm-meters. Set to 0 to specify a perfectly conducting ground.
Geometry	Structure specifying the line geometry and conductor types.
Conductors	Structure specifying the conductor characteristics for each type.

Field	Description
evaluatedFrom	Specify how the internal conductor inductance is evaluated. Set to 'T/D ratio' to compute the inductance from DATA.Conductors.ThickRatio field. When this option is selected, the DATA.Conductors.GMR and DATA.Conductors.Xa fields are not considered by the function, you leave the values as-is or to empty values. Set to 'GMR' to compute the inductance from DATA.Conductors.GMR field. When this option is selected, the DATA.Conductors.Xa field is not considered by the function, you leave the Xa value as-is or to empty value. Set to 'Xa' to compute the inductance from DATA.Conductors.Xa field. When this option is selected, the DATA.Conductors.GMR field is not considered by the function, you leave the GMR value as-is or to empty value.

Geometry is a structure with the fields in this table.

Field	Description
NPhaseBundle	The number of phase conductors (single conductors or bundles of subconductors). The default is 3.
NGroundBundle	The number of ground wires (single conductors or bundles of subconductors). The default is 2.
PhaseNumber	1-by-NPhaseBundle + NGroundBundle vector. This setting represents the phase number to which the conductors belong. Several conductors may have the same phase number. The phase number of ground is 0. The default is [1 2 3 0 0].
X	1-by-NPhaseBundle + NGroundBundle vector. This setting represents the horizontal position of the conductors, in meters or feet. The location of the zero reference position is arbitrary. For a symmetrical line, you typically choose $X = 0$ at the center of the line. The default is [-12 0 12 -8 8].
Ytower	1-by-NPhaseBundle + NGroundBundle vector. This setting represents the vertical position of the conductors (at the tower) with respect to ground, in meters or feet. The default is [20 20 20 33 33].
Ymin	1-by-NPhaseBundle + NGroundBundle vector. This setting represents the vertical position of the conductors with respect to ground at mid-span, in meters or feet. The default is [20 20 20 33 33].
ConductorType	1-by-NPhaseBundle + NGroundBundle vector. This setting represents the type of conductor. The numbers correspond to the elements of the vector defined in the Conductors structure. The default is [1 1 1 2 2].

Conductors is a structure with the fields in this table. The length of the Conductors structure corresponds to the number of conductor types in your line parameters:

Field	Description
Diameter	The conductor outside diameter, in cm. The default is [3.5500 1.2700].
ThickRatio	The T/D ratio of the hollow conductor. T is the thickness of the conducting material, and D is the outside diameter. This parameter can vary between 0 and 0.5. A T/D value of 0.5 indicates a solid conductor. For Aluminum Cable Steel Reinforced (ACSR) conductors, you can ignore the steel core and consider a hollow aluminum conductor (typical T/D ratios are between 0.3 and 0.4). The default is [0.3700 0.5000].
GMR	The Geometric Mean Radius.
Xa	The Reactance Xa, in ohms/km. The default is 0.
Res	The conductor DC resistance, in ohms/km. The default is [0.0430 3.1060].
Mur	The conductor relative permeability. The default is [1 1].
NConductors	The number of conductors per bundle. The default is [4 1].
BundleDiameter	The bundle diameter, in cm or inches. The default is [65 0].
AngleConductor1	The angle, in degrees, that determines the position of the first conductor in the bundle with respect to a horizontal line parallel to ground. This angle determines the bundle orientation. The default is [45 0].
skinEffect	Set to yes to include the impact of the frequency on conductor AC resistance and inductance (skin effect). If set to no, the resistance is kept constant at the value specified by the Res field, and the inductance is kept constant at the value computed in DC, using the Diameter field, and the ThickRatio field.

If you input an existing structure into the power_lineparam function, the returned DATA structure is augmented with the RLC line parameters fields in this table:

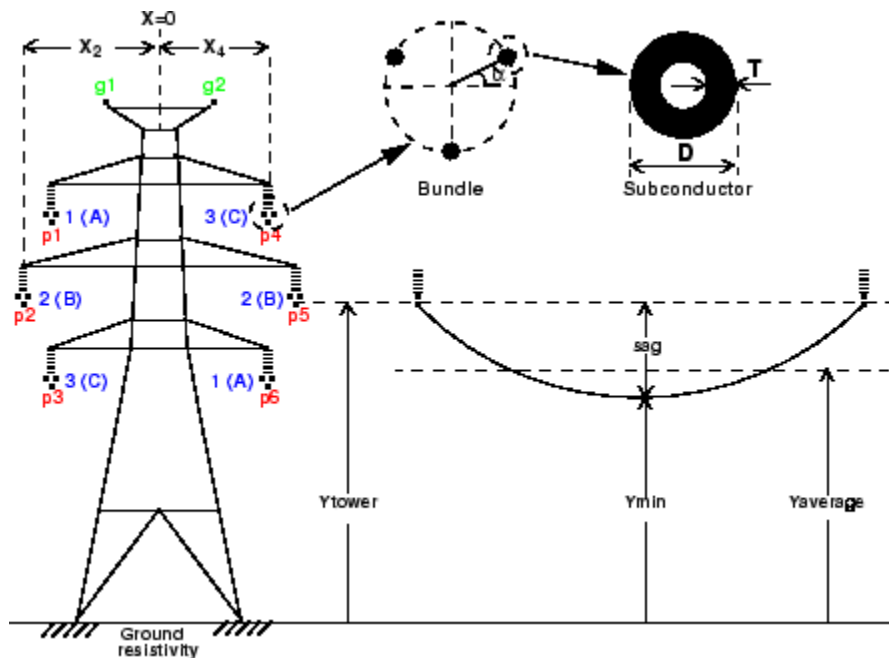
Field	Description
R	The N -by- N resistance matrix of the line, in ohms/km.
L	The N -by- N inductance matrix of the line, in Henries/km.
C	The N -by- N capacitance matrix of the line, in Farads/km.
R10	For a three-phase line, this setting represents the positive-sequence and zero-sequence resistances, in ohms/km. For a six-phase line (two coupled three-phase circuits), this setting represents the positive-sequence and zero-sequence resistances of circuit 1, the mutual zero-sequence resistance between circuit 1 and circuit 2, and the positive-sequence and zero-sequence resistances of circuit 2.

Field	Description
L10	For a three-phase line, the positive-sequence and zero-sequence inductances, in Henries/km. For a six-phase line (two coupled three-phase circuits), this setting represents the positive-sequence and zero-sequence inductances of circuit 1, the mutual zero-sequence inductance between circuit 1 and circuit 2, and the positive-sequence and zero-sequence inductances of circuit 2.
C10	For a three-phase line, this setting represents the positive-sequence and zero-sequence capacitances, in Farads/km. For a six-phase line (two coupled three-phase circuits), this setting represents the positive-sequence and zero-sequence capacitances of circuit 1, the mutual zero-sequence capacitance between circuit 1 and circuit 2, and the positive-sequence and zero-sequence capacitances of circuit 2.

Algorithms

The `power_lineparam` function computes the resistance, inductance, and capacitance matrices of an arbitrary arrangement of conductors of an overhead transmission line. For a three-phase line, the symmetrical component RLC values are also computed.

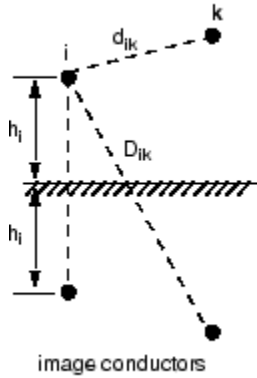
The following figure shows a typical conductor arrangement for a three-phase double-circuit line. This line configuration illustrates the various line parameters you specify in the DATA structure.



Configuration of a Three-Phase Double-Circuit Line

For a set of N conductors, `power_lineparam` computes three N -by- N matrices: the series resistance and inductance matrices $[R]$ and $[\bar{L}]$ and the shunt capacitance matrix $[C]$. These matrices are

required by the Distributed Parameters Line block for modeling N -phase asymmetrical lines and by the single-phase PI Section Line block. `power_lineparam` also computes the symmetrical component RLC parameters required by the Three-Phase PI Section Line block. For two coupled conductors, i and k , the self and mutual terms of the R, L, and C matrices are computed using the concept of image conductors [1].



The self and mutual resistance terms are:

$$R_{ii} = R_{int} + \Delta R_{ii} \text{ } \Omega/\text{km}$$

$$R_{ik} = \Delta R_{ik} \text{ } \Omega/\text{km}$$

The self and mutual inductance terms are:

$$L_{ii} = L_{int} + \frac{\mu_0}{2\pi} \cdot \log \frac{2h_i}{r_i} + \Delta L_{ii} \text{ } H/\text{km}$$

$$L_{ik} = \frac{\mu_0}{2\pi} \cdot \log \frac{D_{ik}}{d_{ik}} + \Delta L_{ik} \text{ } H/\text{km}$$

The self and mutual potential coefficients terms are:

$$P_{ii} = \frac{1}{2\pi\epsilon_0} \cdot \log \frac{2h_i}{r_i} \text{ } \text{km}/\text{F}$$

$$P_{ik} = \frac{1}{2\pi\epsilon_0} \cdot \log \frac{D_{ik}}{d_{ik}} \text{ } \text{km}/\text{F}$$

$$[C] = [P]^{-1} \text{ } F/\text{km}$$

where:

- μ_0 is the permeability of free space = $4\pi \cdot 10^{-4} \text{ H/km}$
- ϵ_0 is the permittivity of free space = $8.8542 \cdot 10^{-9} \text{ F/km}$
- r_i is the radius of conductor i in meters
- d_{ik} is the distance between conductors i and k , in meters

- D_{ik} is the distance between conductor i and image of k , in meters
- h_i is the average height of conductor i above ground, in meters
- R_{int} , L_{int} are the internal resistance and inductance of conductor
- ΔR_{ii} , ΔR_{ik} are the Carson R correction terms due to ground resistivity
- ΔL_{ii} , ΔL_{ik} are the Carson L correction terms due to ground resistivity

The conductor self-inductance is computed from the magnetic flux circulating inside and outside the conductor, and produced by the current flowing in the conductor itself. The part of flux circulating inside the conducting material contributes to the internal inductance L_{int} , which is dependent on the conductor geometry. Assuming a hollow or solid conductor, the internal inductance is computed from the T/D ratio, where D is the conductor diameter and T is the thickness of the conducting material. See the figure "Configuration of a Three-Phase Double-Circuit Line" on page 2-41. The conductor self-inductance is computed by means of modified Bessel functions from the conductor diameter, T/D ratio, resistivity, and relative permeability of conducting material and specified frequency [1].

The conductor self-inductance can also be computed from parameters that are usually found in tables provided by conductor manufacturers: the Geometric Mean Radius (GMR) or the "Reactance at one-foot spacing."

The GMR is the radius of the equivalent hollow conductor with zero thickness, producing no internal flux, giving the same self-inductance. The conductor self-inductance is then derived from the GMR using the following equation:

$$L_{ii} = \frac{\mu_0}{2\pi} \cdot \log \frac{2h_i}{GMR} + \Delta L_{ii} \text{ H/km}$$

For a solid conductor ($T/D=0.5$), the GMR is

$$GMR = r \cdot e^{-\mu_r/4}$$

r = radius of conductor

μ_r = relative permeability of conducting material

The GMR obtained from this equation assumes a uniform current density in the conductor. This assumption is strictly valid in DC. In AC, the GMR is slightly higher. For example, for a 3 cm diameter solid aluminum conductor ($R_{dc} = 0.040 \text{ } \Omega/\text{km}$), the GMR increases from 1.1682 cm in DC to 1.1784 cm at 60 Hz. Manufacturers usually give the GMR at the system nominal frequency (50 Hz or 60 Hz).

The reactance X_a at 1-foot spacing (or 1-meter spacing, if metric units are used) is the positive-sequence reactance of a three-phase line having one foot (or one meter) spacing between the three phases and infinite conductor heights. The reactance at 1-foot spacing (or 1-meter spacing) at frequency f is related to the GMR by the following equation:

$$X_a = \omega \cdot \frac{\mu_0}{2\pi} \cdot \log \frac{1}{GMR} \text{ } \Omega/\text{km}$$

GMR = Geometric Mean Radius in feet or meters

$\omega = 2\pi \cdot f$ in rad/s

f = frequency in hertz

The conductor resistance matrix at a particular frequency depends on the DC resistance of the conductor corrected for skin effect and ground resistivity. Both the resistance matrix and the

inductance matrix are dependent on the ground resistivity and frequency. Correction terms for the R and L terms as developed by J.R. Carson in 1926 [2] are implemented in `power_lineparam`.

References

- [1] Dommel, H., et al., *Electromagnetic Transients Program Reference Manual* (EMTP Theory Book), 1986.
- [2] Carson, J. R., "Wave Propagation in Overhead Wires with Ground Return," *Bell Systems Technical Journal*, Vol. 5, pp 539-554, 1926.

See Also

Powergui | Distributed Parameters Line | PI Section Line | Three-Phase PI Section Line | Power Line Parameters

power_loadflow

Perform positive-sequence load flow or unbalanced load flow and initialize models containing load flow blocks

Syntax

```
LF = power_loadflow(sys, 'solve')
LF = power_loadflow(sys, 'noupdate')
power_loadflow(sys, 'solve', 'report')
power_loadflow(sys, 'solve', 'report', fname)
power_loadflow(sys, 'solve', 'ExcelReport', fname)
power_loadflow(sys, 'AddBuses')
```

Description

`LF = power_loadflow(sys, 'solve')` computes the load flow of `sys`. The model is initialized with the load flow solution.

Note Before release R2021a, this function provided access to the machine tool initialization. Although this is still possible, use this function in release R2021a and later to initialize models containing electric machines. Although this function continues to support the `-v2` option, this option is no longer required to perform load flows. You can remove this option from your scripts and M functions.

`LF = power_loadflow(sys, 'noupdate')` computes the load flow but does not initialize the model with the load flow solution.

`power_loadflow(sys, 'solve', 'report')` computes the load flow and opens the editor to save the load flow report.

`power_loadflow(sys, 'solve', 'report', fname)` computes the load flow and saves detailed information in the `fname` file.

`power_loadflow(sys, 'solve', 'ExcelReport', fname)` computes the load flow and saves detailed information in the `fname` Excel® file. This capability is available only for the positive-sequence load flow.

`power_loadflow(sys, 'AddBuses')` adds Load Flow Bus blocks to the model `sys`. The added blocks are single-connector Load Flow Bus blocks, which specify bus parameters for a positive-sequence load flow.

Examples

Performing Load Flow Analysis at the Command Line

As an alternative to using the Load Flow Analyzer to perform a load flow, you can use this function at the command line. For example, to perform the positive-sequence load flow on the “Initializing a 5-Bus Network with the Load Flow Tool of Powergui” model, enter:

```
LF = power_loadflow('-v2','power_LFnetwork_5bus','solve')
```

```
LF =  
    model: 'power_LFnetwork_5bus'  
  frequency: 60  
 basePower: 100000000  
 tolerance: 0.0001  
   Ybus1: [5x5 double]  
     bus: [1x7 struct]  
      sm: [1x1 struct]  
     asm: [1x1 struct]  
    vsrc: [1x1 struct]  
   pqload: [1x1 struct]  
  rlcload: [1x2 struct]  
 Networks: [1x1 struct]  
   status: 1  
 iterations: 2  
    error: ''  
 LoadFlowSolver: 'PositiveSequence'
```

The `power_loadflow` function returns the solution in the LF structure, and the model is initialized to start in steady state. You can obtain a detailed load flow report by entering:

```
LF = power_loadflow('-v2','power_LFnetwork_5bus','solve','report');
```

The function prompts you to save the report in a file that is displayed in the MATLAB Editor.

You can use the same command to perform an unbalanced load flow on the “IEEE 13 Node Test Feeder” model:

```
LF = power_loadflow('-v2','power_13NodeTestFeeder','solve')
```

```
LF =  
    model: 'power_13NodeTestFeeder'  
  frequency: 60  
 basePower: 100000  
 tolerance: 0.0001  
   Ybus: [29x29 double]  
     bus: [1x29 struct]  
      sm: [1x1 struct]  
     asm: [1x1 struct]  
    vsrc: [1x1 struct]  
   pqload: [1x1 struct]  
  rlcload: [1x1 struct]  
 Networks: [1x1 struct]  
   status: 1  
 iterations: 3  
    error: ''
```

```
LoadFlowSolver: 'Unbalanced'
```

Input Arguments

sys — Model for computing and initialization of loadflow

model

Model of which you want to compute the load flow and initialize the load flow solution.

fname — File name

character vector

File name on which you want to save the detailed information of the load flow solution, specified as a character vector.

Output Arguments

LF — Load flow parameters and solutions

structure

Load flow parameters and solutions, returned as a structure with these fields:

Field	Description
model	The name of the model.
frequency	The load flow frequency, in hertz. This value corresponds to the Load flow frequency parameter of the powergui block.
basePower	The base power used by the Load Flow Analyzer. This value corresponds to the Base power Phase parameter of the powergui block.
tolerance	The tolerance used by the Load Flow Analyzer. This value corresponds to the PQ Tolerance parameter of the powergui block.
bus	[1 × Nbus] structure with fields defining the bus parameters. For a positive-sequence load flow, Nbus is the number of three-phase buses in the model. For an unbalanced load flow, Nbus is the total number of single-phase buses in the model.
sm	[1 × Nsm] structure with fields defining the load flow parameters of the Synchronous Machine blocks. Nsm is the number of Synchronous Machine blocks in the model.
asm	[1 × Nasm] structure with fields defining the load flow parameters of the Asynchronous Machine blocks. Nasm is the number of Asynchronous Machine blocks in the model.

Field	Description
<code>vsrc</code>	[1 × Nsrc] structure with fields defining the load flow parameters of the Three-Phase Source, Three-Phase Programmable Voltage Source, and AC Voltage Source blocks. Nsrc is the number of single-phase and three-phase voltage source blocks in the model.
<code>pqload</code>	[1 × Npq] structure with fields defining the load flow parameters of the Three-Phase Dynamic Load blocks. Npq is the number of Three-Phase Dynamic Load blocks in the model.
<code>rlcload</code>	[1 × Nr lc] structure with fields defining the load flow parameters of the Three-Phase Parallel RLC Load, Three-Phase Series RLC Load, Parallel RLC Load, and Series RLC Load blocks. Nr lc is the number of three-phase and single-phase blocks in the model.
<code>Ybus1</code>	[Nbus × Nbus] positive-sequence complex admittance matrix in pu/Phase used for positive-sequence load flow solution.
<code>Ybus</code>	[Nbus × Nbus] complex admittance matrix in pu/Phase used for unbalanced load flow solution.
<code>Networks</code>	The bus numbers of each independent network.
<code>status</code>	Returns 1 when a solution is found, and -1 when no solution is found.
<code>iterations</code>	The number of iterations that the solver took to solve the load flow.
<code>error</code>	Displays an error message when no solution is found.
<code>LoadFlowSolver</code>	The type of load flow solver: 'PositiveSequence' or 'Unbalanced'

Algorithms

The `power_loadflow` function computes the positive-sequence load flow or the unbalanced load flow of a Simscape Electrical Specialized Power Systems model and initializes the load flow blocks to start simulation in steady-state. Load flow blocks are Simscape Electrical Specialized Power Systems blocks in which you can specify active power (P) and reactive power (Q) to solve the positive-sequence load flow.

The `power_loadflow` function uses the Newton-Raphson method to provide a robust and fast convergence solution and offers most of the functionality of other load flow software available in the power utility industry.

The `power_loadflow` function applies the load flow solution to the model by initializing these block parameters:

- The **Phase-to-phase voltage** and **Phase angle of phase A** parameters of Three-Phase Source blocks

- The **Positive-sequence: [Amplitude Phase]** parameter of Three-Phase Programmable Voltage Source blocks
- The **Nominal phase-to-phase voltage** parameter of Three-Phase Parallel RLC Load and Three-Phase Series RLC Load blocks
- The **Active and reactive power at initial voltage [Po Qo]** parameter and **Initial positive-sequence voltage Vo [Mag(pu) Phase (deg.)]** parameter of Three-Phase Dynamic Load blocks
- The **Initial Conditions [slip, th, ia,ib,ic, pha,phb,phc]** parameter of Asynchronous Machine blocks
- The **Initial conditions [dw th ia,ib,ic pha,phb,phc Vf]** parameter of Synchronous Machine and Simplified Synchronous Machine blocks

The function also initializes the following parameters of the blocks that are connected to the **Pm**, **Vf**, and **Tm** inputs of the Synchronous Machine and Asynchronous Machine blocks:

- The **Constant value** parameter of Constant blocks
- The **Initial Value** parameter of Step blocks
- The **Initial mechanical power** parameter of Hydraulic Turbine and Governor blocks
- The **Initial power and generator rotor angle** of the Steam Turbine and Governor blocks

The type of load flow solution (positive-sequence or unbalanced) is determined by the Load Flow Bus blocks present in the model. For more information on how to use the Load Flow Bus blocks in your model, see Load Flow Bus.

Version History

Introduced before R2006a

See Also

Load Flow Analyzer | Load Flow Bus

Topics

“Initializing a 5-Bus Network with the Load Flow Tool of Powergui”
 “IEEE 13 Node Test Feeder”

power_ltiview

Create state-space object of model for use with the Linear System Analyzer tool

Syntax

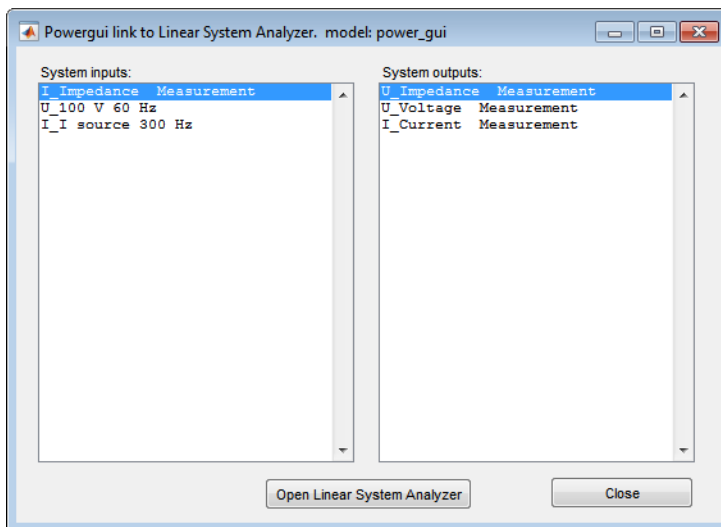
Linear System Analyzer

```
power_ltiview('sys')
```

Description

`power_ltiview('sys')` creates a state-space object (SS) of the linear part of the electrical circuit (ABCD state-space matrices representing the RLC devices in the model). The nonlinear elements, such as the distributed parameter line blocks, are modeled as inputs and outputs to the SS object (because they are modeled as current source in feedback to the state-space system). You can analyze the time and frequency responses of this SS object with the Linear System Analyzer. The command opens the Linear System Analyzer dialog box. This tool can also be activated from the Powergui block dialog box by selecting **Use Linear System Analyzer**. The Linear System Analyzer functionality is provided with Control System Toolbox software.

Dialog Box



System inputs

Lists the inputs of the state-space model of your circuit. Select the inputs to be used by the Linear System Analyzer.

System outputs

Lists the outputs of the state-space model of your circuit. Select the outputs to be used by the Linear System Analyzer.

Open Linear System Analyzer

Generates the state-space model of the circuit and opens the Linear System Analyzer for the selected system inputs and outputs.

power_new

Create new Simscape Electrical Specialized Power Systems model populated by required and commonly used blocks

Syntax

`power_new`

Description

`power_new` creates a new Simscape Electrical Specialized Power Systems model, with the Powergui block already on the model canvas and the Simulink solver set to `VariableStepAuto`.

The function also opens the `power_new_palette` library that contains commonly used Simscape Electrical, Simulink, and Simscape blocks.

power_PMSynchronousMachineParams

Convert manufacturer specifications to mask parameters of Permanent Magnet Synchronous Machine block

Syntax

```
power_PMSynchronousMachineParams
params = power_PMSynchronousMachineParams(spec)
```

Description

`power_PMSynchronousMachineParams` opens a graphical user interface to compute the parameters of the Permanent Magnet Synchronous Machine block from standard manufacturer specifications.

`params = power_PMSynchronousMachineParams(spec)` computes the block parameters from the input structure `spec` which contains the manufacturer specifications. The solution is returned in the output structure `params`.

This table shows the electrical and mechanical parameters of the Permanent Magnet Synchronous Machine block along with their units.

Parameter	Unit
Stator phase resistance, R_s	ohm
Stator phase inductance, L_s (trapezoidal back EMF waveform model)	H
Armature inductance, L (sinusoidal back EMF round rotor type)	H
Inductances, [L_d , L_q] (sinusoidal back EMF salient-pole rotor type)	H
Flux linkage established by magnets	V.s
Voltage constant, k_e (peak line-to-line voltage per 1000 rpm)	V _{peak} /krpm
Torque constant, k_t (torque per ampere constant)	N.m/A _{peak}
Inertia, J (combined machine and load inertia)	kg.m ²
Viscous damping, F (combined machine and load viscous damping)	N.m.s

Manufacturers provide machine parameters in several units, usually different from the units required by the Permanent Magnet Synchronous Machine block. This table provides the parameters and units commonly found in manufacturer specifications.

Manufacturer Specification	Units
Line-to-line resistance, R	ohm (line-line)
Line-to-line inductance, Lab	mH (line-line)
d- and q-axis phase inductances, Ld and Lq	mH
Voltage Constant ke	Vpeak/krpm Vrms/krpm Vpeak/rad/s Vrms/rad/s
Torque Constant kt	N.m/Apeak N.m/Arms N.cm/Apeak oz.in/Arms lb.in/Apeak lb.ft/Arms
Inertia J	kg.m ² kg.cm ² g.cm ² lb.in ² kg.cm.s ² lb.in.s ² oz.in.s ²
Viscous damping F	N.m.s N.m/rpm oz.in/rpm oz.in/krpm

The `power_PMSynchronousMachineParams` function converts the given manufacturer parameters and units to the parameters required by the Permanent Magnet Synchronous Machine block. The function computes the machine parameters using:

- Stator phase resistance

$$R_s = \frac{R}{2}$$

- Stator phase inductance

$$L_s = \frac{L_{ab}}{2}$$

- Armature inductance

$$L = \frac{L_{ab}}{2}$$

- Voltage constant

$$1 \frac{V_{peak}}{krpm} = \frac{3}{100 \cdot \pi} \frac{V_{peak}}{(rad/s)} = 1 \frac{\sqrt{2} \cdot V_{rms}}{krpm} = \frac{3}{100 \cdot \pi} \frac{\sqrt{2} \cdot V_{rms}}{(rad/s)}$$

- Torque constant

$$1 \frac{N \cdot m}{A_{peak}} = \frac{1}{\sqrt{2}} \frac{N \cdot m}{A_{rms}} = 100.13 \frac{oz \cdot in}{A_{rms}} = 8.85 \frac{lb \cdot in}{A_{peak}} = 0.52 \frac{lb \cdot ft}{A_{rms}}$$

- Inertia

$$1 \text{ kg} \cdot \text{m}^2 = 3417.17 \text{ lb} \cdot \text{in}^2 = 10.197 \text{ kg} \cdot \text{cm} \cdot \text{s}^2 = 141.612 \text{ oz} \cdot \text{in} \cdot \text{s}^2$$

- Viscous damping

$$1 \text{ N} \cdot \text{m} \cdot \text{s} = 0.1047 \frac{\text{N} \cdot \text{m}}{\text{rpm}} = 14.829 \frac{\text{oz} \cdot \text{in}}{\text{rpm}} = 14.829 \times 10^3 \frac{\text{oz} \cdot \text{in}}{\text{krpm}}$$

Input Arguments

spec

The manufacturer data is specified in a structure with the following fields.

Field	Description
backEMF	Specifies the shape of back EMF waveform {'sinusoidal','trapezoidal'}
rotorType	Specifies the type of rotor {'round','salient'}
R	Line-to-line resistance (ohm)
Lab	Line-to-line inductance (mH). Specify this field only for a round, rotor machine.
Ld	<i>d</i> -axis phase inductance (mH). Specify this field only for a salient-pole rotor machine.
Lq	<i>q</i> -axis phase inductance (mH). Specify this field only for a salient-pole rotor machine.
suppliedConstant	Specifies the type of constant (k) provided to the function {'voltage','torque'}
k	Value of the voltage or torque constant
kUnitsNum	Units of the voltage or torque constant numerator {'Vpeak','Vrms','N.m','N.cm','oz.in','lb.in','lb.ft'}
kUnitsDenom	Units of the voltage or torque constant denominator {'krpm','rad/s','Apeak','Arms'}
J	Moment of inertia
inertiaUnits	Units for moment of inertia {'kg.m^2','kg.cm^2','g.cm^2','lb.in.s^2','oz.in.s^2'}
F	Viscous damping
frictionUnits	Units for viscous damping {'N.m.s','N.m/rpm','oz.in/rpm','oz.in/krpm'}
p	Number of pole pairs

params contains additional options to the function in a structure with the following fields.

Output Arguments

params

The Permanent Magnet Synchronous Machine block parameters are returned in a structure with these fields.

Field	Description
λ	Flux induced by magnets (V.s)
k_e	Voltage constant (V _{peak} /krpm)
k_t	Torque constant (N.m/A _{peak})
J	Machine inertia (kg.m ²)
F	Machine viscous damping (N.m.s)
R_s	Stator phase resistance (ohm)
L_s	Stator phase inductance (H). Computed for the round rotor machine
L_d	D-axis phase inductance (H). Computed for a sinusoidal back EMF with salient-pole rotor machine.
L_q	Q-axis phase inductance (H). Computed for a sinusoidal back EMF and salient-pole rotor machine.
p	Number of pole pairs

Interface

You can also obtain the mask parameters of the Permanent Magnet Synchronous Machine block through the user interface of the function.

Specifications

Enter the manufacturer specifications. Depending on the option that you select in the **Back EMF waveform**, **Rotor type**, and **Specify** parameters, some specifications are not available.

Unavailable parameters are not taken into account in block parameter computation.

Compute Block Parameters

Compute the electrical parameters and display the solution in the **Block Parameters** section. When you obtain the block parameters, the button is unavailable.

When you enter new values in the **Specifications** section, the button is enabled to indicate that the displayed block parameters no longer correspond to the displayed specifications.

Block Parameters

Displays the parameters corresponding to the given manufacturer specifications. The first time that you open the interface, these fields are empty. You cannot edit these block parameters.

When the **Rotor type** is set to **Salient-pole**, the **Armature inductance** parameter is unavailable, indicating that its value is not computed using the **Compute Block Parameters** button.

When the Rotor type is set to Round, the **Inductances** parameter is unavailable, indicating that its value is not computed using the **Compute Block Parameters** button.

Apply to Selected Block

Open your model, select the machine you want to parametrize, then click the **Apply to selected block** button to load the displayed block parameters in the block. When the **Block Parameters** fields are empty, the button is unavailable. The button is enabled when a valid solution is displayed in the **Block Parameters** section.

The button has no effect on a selected block that is not a Permanent Magnet Synchronous Machine block.

Help

Opens the help page of the power_PMSynchronousMachineParams function.

Close

Closes the interface.

Examples

Consider a 4 kW, permanent magnet synchronous machine (round rotor and sinusoidal back emf) with this manufacturer specification.

Data	Values
R	0.36 ohm (line-line)
Lab	1.67 mH (line-line)
kt	60.70 oz.in/Apeak
J	5.5E-03 lb.in.sec^2
B	4.5 oz.in/krpm
p	4

Based on this specification, to obtain the required mask parameters for the Permanent Magnet Synchronous Machine block, in the Command Window, enter these commands:

```
spec.backEMF = 'sinusoidal';
spec.rotorType = 'round' ;
spec.R = 0.36;
spec.Lab = 1.67;
spec.suppliedConstant = 'torque';
spec.k = 60.70;
spec.kUnitsNum = 'oz.in';
```

```
spec.kUnitsDenom = 'Apeak';  
spec.J = 5.5e-3;  
spec.inertiaUnits = 'lb.in.s^2' ;  
spec.F = 4.5;  
spec.frictionUnits = 'oz.in/krpm';  
spec.p = 4;  
  
params = power_PMSynchronousMachineParams(spec)
```

You obtain these results:

```
params =  
  
    lambda: 0.0714  
    ke: 51.8307  
    kt: 0.4286  
    J: 6.2142e-004  
    F: 3.0345e-004  
    Rs: 0.1800  
    Ls: 8.3500e-004  
    Ld: []  
    Lq: []  
    p: 4
```

See Also

`power_AsynchronousMachineParams`

power_report

Generate report of steady state variables, initial states, and machine load flow for model

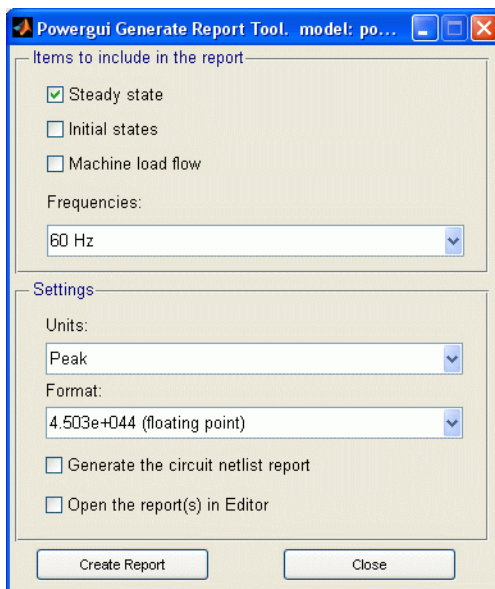
Syntax

```
power_report('sys')
```

Description

`power_report('sys')` opens the Generate Report Tool dialog box, which lets you generate a report of steady state variables, initial states, and machine load flow for a model. This tool can also be activated from the Powergui block dialog box by selecting **Generate Report**.

Dialog Box



Items to include in the report

In the check boxes, select any combination of measurements to include in the generated report, **Steady state**, **Initial states**, and **Machine load flow**. The default is unselected for all three.

Frequencies

Select the frequency or frequencies to include in the generated report, **60 Hz** or **All**. The default is **60 Hz**.

Units

Set the **Units** parameter to **Peak** values to display the peak values of the selected values. Set the **Units** parameter to **RMS** to display the root-mean-square (RMS) values of the selected values.

Format

In the pull-down menu, choose the format in which you want your measurements displayed. The **floating point** option is displayed in mantissa-exponent form with five significant figures. The

`best` of option is displayed with four significant figures and uses mantissa-exponent form only for numbers larger than 9999. The final option is displayed in plain numbers with two figures to the right of the decimal point. The default is `floating point`.

Generate the circuit netlist report

Generates a netlist stored in a file, `sys.net`. The file contains the node numbers automatically generated by `power_analyze`, as well as parameter values of all linear elements. See the formats described in the `power_statespace` reference page.

Open the report(s) in Editor

Lets you edit the report.

Create Report

Generate a report and save it to a file.

See Also

`power_analyze` | `power_statespace`

power_statespace

Compute state-space model of linear electrical circuit

Syntax

```
[A,B,C,D,states,x0,x0sw,rlsw,u,x,y,freq,Asw,Bsw,Csw,Dsw,Hlin] =  
power_statespace(rlc,switches,source,line_dist,yout,y_type,unit )  
[A,B,C,D,states,x0,x0sw,rlsw,u,x,y,freq,Asw,Bsw,Csw,Dsw,Hlin] =  
power_statespace(rlc,switches,source,line_dist,yout,y_type,unit ,  
net_arg1,net_arg2,net_arg3,...,netsim_flag,fid_outfile,  
freq_sys,ref_node,vary_name,vary_val)
```

Description

[A,B,C,D,states,x0,x0sw,rlsw,u,x,y,freq,Asw,Bsw,Csw,Dsw,Hlin] = power_statespace(rlc,switches,source,line_dist,yout,y_type,unit) computes the state-space model of a linear electrical circuit expressed as

$$\dot{x} = Ax + Bu$$

$$y = Cx + Du$$

where x is the vector of state-space variables (inductor currents and capacitor voltages), u is the vector of voltage and current inputs, and y is the vector of voltage and current outputs.

When you build a circuit from Simscape Electrical Specialized Power Systems blocks, `power_statespace` is automatically called by the `power_analyze` command. `power_statespace` is also available as a stand-alone command for expert users. This allows you to generate state-space models without using the Simscape Electrical Specialized Power Systems block modeling interface and to access options that are not available through the Simscape Electrical Specialized Power Systems library. For example, using `power_statespace`, you can model transformers and mutual inductances with more than three windings.

You must call `power_statespace` with a minimum of seven input arguments.

The linear circuit can contain any combination of voltage and current sources, RLC branches, multiwinding transformers, mutually coupled inductances, and switches. The state variables are inductor currents and capacitor voltages.

The state-space representation (matrices A,B,C,D, and vector x_0) computed by `power_statespace` can then be used in a Simulink system, via a State-Space block, to perform simulation of the electrical circuit (see “Examples” on page 2-69). Nonlinear elements (mechanical or power electronic switches, transformer saturation, machines, distributed parameter lines, etc.) can be connected to the linear circuit.

These Simulink models are interfaced with the linear circuit through voltage outputs and current inputs of the state-space model. You can find the models of the nonlinear elements provided with Simscape Electrical software in the Specialized Power Systems library.

```
[A,B,C,D,states,x0,x0sw,rlsw,u,x,y,freq,Asw,Bsw,Csw,Dsw,Hlin] =  
power_statespace(rlc,switches,source,line_dist,yout,y_type,unit ,
```

`net_arg1, net_arg2, net_arg3, ..., netsim_flag, fid_outfile, freq_sys, ref_node, vary_name, vary_val`) computes the same as above but you can also specify optional arguments. To use these optional arguments, the number of input arguments must be 12, 13, 14 or 16.

Input Arguments

The number of input arguments must be 7, 12, 13, 14, or 16. Arguments 8 to 16 are optional. The first seven arguments that must be specified are

- `rlc`: Branch matrix specifying the network topology as well as the resistance R, inductance L, and capacitance C values. See format below.
- `switches`: Switch matrix. Specify an empty variable if no switches are used. See format below.
- `source`: Source matrix specifying the parameters of the electrical voltage and current sources. Specify an empty variable if no sources are used. See format below.
- `line_dist`: Distributed parameter line matrix. Specify an empty variable if no distributed lines are used. See format below.
- `yout`: Matrix of output, expressed as character vectors. See format below.
- `y_type`: Integer vector indicating output types (0 for voltage output, 1 for current output).
- `unit`: Character vector specifying the units to be used for R, L, and C values in the `rlc` matrix. If `unit = 'OHM'`, R L C values are specified in ohms Ω at the fundamental frequency specified by `freq_sys` (default value is 60 Hz). If `unit = 'OMU'`, R L C values are specified in ohms (Ω), millihenries (mH), and microfarads (μF).

The last nine arguments are optional. The first three are used to pass arguments from the `power_analyze` command. Hereafter, only the arguments to be specified when `power_statespace` is used as a stand-alone command are described:

- `net_arg1, net_arg2, net_arg3`: Used to pass arguments from `power_analyze`. Specify an empty variable `[]` for each of these arguments.
- `netsim_flag`: Integer controlling the messages displayed during the execution of `power_statespace`. Default value is 0.

If `netsim_flag = 0`, the version number, number of states, inputs, outputs, and modes are displayed. Output values are displayed in polar form for each source frequency.

If `netsim_flag = 1`, only version number, number of states, inputs, and outputs are displayed.

If `netsim_flag = 2`, no message is displayed during execution.

- `fid_outfile`: File identifier of the `power_statespace` output file containing parameter values, node numbers, steady-state outputs, and special messages. Default value is 0.
- `freq_sys`: Fundamental frequency (Hz) considered for specification of X_L and X_C reactances if `unit` is set to `'OHM'`. Default value is 60 Hz.
- `ref_node`: Reference node number used for ground of PI transmission lines. If `-1` is specified, the user is prompted to specify a node number.
- `vary_name`: Matrix containing the symbolic variable names used in output, expressed as character vectors. These variables must be defined in your MATLAB workspace.
- `vary_val`: Vector containing the values of the variable names specified in `vary_name`.

Output Arguments

- A, B, C, D : state-space matrices of the linear circuit with all switches open.

$A(nstates, nstates)$, $B(nstates, ninput)$,
 $C(noutput, nstates)$, $D(noutput, ninput)$,

where $nstates$ is the number of state variables, $ninput$ is the number of inputs, and $noutput$ is the number of outputs.

- **states**: Matrix containing the names of the state variables. Each name has the following format:

Inductor currents: $Il\_bxx\_nzz1\_zz2$

Capacitor voltages: $Uc\_bxx\_nzz1\_zz2$

where

xx = branch number
 $zz1$ = first node number of the branch
 $zz2$ = second node number of the branch

The last lines of the **states** matrix, which are followed by an asterisk, indicate inductor currents and capacitor voltages that are not considered as state variables. This situation arises when inductor currents or capacitor voltages are not independent (inductors forming a cut set – for example, inductors connected in series – or capacitors forming a loop). The currents and voltages followed by asterisks can be expressed as a linear combination of the other state variables:

- $x0$: Column vector of initial values of state variables considering the open or closed status of switches.
- $x0sw$: Vector of initial values of switch currents.
- $rlsw$: Matrix ($nswitch, 2$) containing the R and L values of series switch impedances in ohms (Ω) and henries (H). $nswitch$ is the number of switches in the circuit.
- u, x, y : Matrices $u(ninput, nfreq)$, $x(nstates, nfreq)$, and $y(noutput, nfreq)$ containing the steady-state complex values of inputs, states, and outputs. $nfreq$ is the length of the **freq** vector. Each column corresponds to a different source frequency, as specified by the next argument, **freq**.
- **freq**: Column vector containing the source frequencies ordered by increasing frequency.
- Asw, Bsw, Csw, Dsw : State-space matrices of the circuit including the closed switches. Each closed switch with an internal inductance adds one extra state to the circuit.
- **Hlin**: Three-dimensional array ($nfreq, noutput, ninput$) of the $nfreq$ complex transfer impedance matrices of the linear system corresponding to each frequency of the **freq** vector.

Format of the RLC Input Matrix

Two formats are allowed:

- Six columns: Implicit branch numbering. Branch numbers correspond to the RLC line numbers.
- Seven columns: Explicit branch numbering. Branch number **Nobr** is assigned by the user.

Each line of the RLC matrix must be specified according to the following format.

[node1, node2, type, R, L, C, Nobr] for RLC branch or line branch

[node1, node2, type, R, L, C, Nobr] for transformer magnetizing branch

[node1, node2, type, R, L, U, Nobr] for transformer winding

[node1, node2, type, R, L, U, Nobr] for mutual inductances

- **node1**: First node number of the branch. The node number must be positive or zero. Decimal node numbers are allowed.
- **node2**: Second node number of the branch. The node number must be positive or zero. Decimal node numbers are allowed.
- **type**: Integer indicating the type of connection of RLC elements, or, if negative, the transmission line length:

type = 0: Series RLC element

type = 1: Parallel RLC element

type = 2: Transformer winding

type = 3: Coupled (mutual) winding

If **type** is negative, the transmission line is modeled by a PI section of length $|\text{type}|$. See details below.

For a mutual inductor or a transformer having N windings, N+1 consecutive lines must be specified in RLC matrix:

- 1 N lines with **type** = 2 or **type** = 3; (one line per winding). Each line specifies R/L/U or R/Xl/Xc where [R/L, R/Xl] = winding resistance and leakage reactance for a transformers or winding resistance and self reactance for mutually coupled windings. U is the nominal voltage of transformer winding (specify 0 if **type** = 3).
- 2 One extra line with **type** = 1 for the magnetizing branch of a transformer (parallel Rm/Lm or Rm/Xm) or one line with **type** = 0 for a mutual impedance (series Rm/Lm or Rm/Xm).

For a transformer magnetizing branch or a mutual impedance, the first node number is an internal node located behind the leakage reactance of the first winding. The second node number must be the same as the second node number of the first winding.

To model a saturable transformer, you must use a nonlinear inductance instead of the linear inductance simulating the reactive losses. Set the Lm/Xm value to 0 (no linear inductance) and use the Saturable Transformer block, set with proper flux-current characteristics.

This block can be found in the Fundamental Blocks/Elements library. It must be connected to the linear part of the system (State-Space block or S-function) between a voltage output (voltage across the magnetizing branch) and a current input (current source injected into the transformer internal node). See the "Examples" on page 2-69.

If **type** is negative, its absolute value specifies the length (km) of a transmission line simulated by a PI section. For a transmission line, the R/L/C or R/Xl/Xc values must be specified in Ω/km , mH/km, and $\mu\text{F}/\text{km}$, or in Ω/km .

Parameter	Description
R	Branch resistance (Ω)

Parameter	Description
Xl	Branch inductive reactance (Ω at <code>freq_sys</code>) or transformer winding leakage reactance (Ω at <code>freq_sys</code>)
L	Branch inductance (mH)
Xc	Branch capacitive reactance (Ω at <code>freq_sys</code>). The negative sign of Xc is optional.
C	Capacitance (μF)
U	Nominal voltage of transformer winding. The same units (volts or kV) must be used for each winding. For a mutual inductance (<code>type=3</code>), this value must be set to zero.
Zero value for R, L or Xl, C or Xc in a series or parallel branch indicates that the corresponding element does not exist.	

The following restrictions apply for transformer winding R-L values. Null values are not allowed for secondary impedances if some transformer secondaries form loops (as in a three-phase delta connection). Specify a very low value for R or L or both (e.g., $1\text{e-}6$ pu based on rated voltage and power) to simulate a quasi-ideal transformer. The resistive and inductive parts of the magnetizing branch can be set to infinite (no losses; specify $X_m = R_m = \text{inf}$).

Format of the Source Input Matrix

Three formats are allowed:

- Five columns: All sources are generating the same frequency specified by `freq_sys`.
- Six columns: The frequency of each source is specified in column 6.
- Seven columns: The seventh column is used to specify the type of nonlinear element modeled by the current source.

Each line of the source matrix must be specified according to the following format:

[`node1`, `node2`, `type`, `amp`, `phase`, `freq`, `model`]

- `node1`, `node2`: Node numbers corresponding to the source terminals. These are the polarity conventions:
 - Voltage source: `node1` is the positive terminal.
 - Current source: Positive current flowing from `node1` to `node2` inside the source.
- `type`: Integer indicating the type of source: 0 for voltage source, 1 for current source.
- `amp`: Amplitude of the AC or DC voltage or current (V or A).
- `phase`: Phase of the AC voltage or current (degree).
- `freq`: Frequency (Hz) of the generated voltage or current. Default value is 60 Hz. For a DC voltage or current source, specify `phase = 0` and `freq = 0`. `amp` can be set to a negative value. The generated signals are

$\text{amp} * \sin(2\pi * \text{freq} * t + \text{phase})$ for AC, `amp` for DC.

- `model`: Integer specifying the type of nonlinear element modeled by the current source (saturable inductance, thyristor, switch,...). Used by `power_analyze` only.

Order in Which Sources Must Be Specified

The commands that compute the state-space representation of a system expect the sources in a certain order. You must respect this order in order to obtain correct results. You must be particularly careful if the system contains any switches. This is the proper ordering of sources:

- 1 The currents from all switches that have a null inductance ($L_{on} = 0$), if any.
- 2 The currents from all nonlinear models that have a finite inductance (switches with $L_{on} > 0$, the magnetizing inductance in saturable transformers, etc.), if any.
- 3 All other voltage and current sources in any order, if any.

Refer to the Example section below for an example illustrating proper ordering of sources for a system containing nonlinear elements.

Format of the Switches Input Matrix

Switches are nonlinear elements simulating mechanical or electronic devices such as circuit breakers, diodes, or thyristors. Like other nonlinear elements, they are simulated by current sources driven by the voltage appearing across their terminals. Therefore, they cannot have a null impedance. They are simulated as ideal switches in series with a series R-L circuit. Various models of switches (circuit breaker, ideal switch, and power electronic devices) are available in the Simscape Electrical Specialized Power Systems library. They must be interconnected to the linear part of the system through appropriate voltage outputs and current inputs.

The switch parameters must be specified in a line of the switches matrix in seven different columns, according to the following format.

```
[ node1, node2, status, R, L/Xl, no_I , no_U ]
```

Parameter	Description
node1, node2	Node numbers corresponding to the switch terminals
status	Code indicating the initial status of the switch at $t = 0$: 0 = open; 1 = closed
R	Resistance of the switch when closed (Ω)
L/Xl	Inductance of the switch when closed (mH) or inductive reactance (Ω at freq_sys)

For these last two fields, you must use the same units as those specified for the RLC matrix. Either field can be set to 0, but not both.

The next two fields specify the current input number and the voltage output number to be used for interconnecting the switch model to the State-Space block. The output number corresponding to the voltage across a particular switch must be the same as the input number corresponding to the current from the same switch (see Example section below):

- no_I: Current input number coming from the output of the switch model
- no_U: Voltage output number driving the input of the switch model

Format of the Line_Dist Matrix

The distributed parameter line model contains two parts:

- 1 A linear part containing current sources and resistances that are connected at the line sending and receiving buses together with the linear circuit.
- 2 A nonlinear part available in the Distributed Parameters Line block of the Elements library. This block performs the phase-to-mode transformations of voltage and currents and simulates the transmission delays for each mode. The *distributed_param_line* block must be connected to appropriate voltage outputs and current inputs of the linear part of the system. The line parameters have to be specified in the `line_dist` matrix and also in the Distributed Parameters Line block.

Each row of the `line_dist` matrix is used to specify a distributed parameter transmission line. The number of columns of `line_dist` depends on the number of phases of the transmission line.

For an `nphase` line, the first $(4 + 3 * nphase + nphase^2)$ columns are used. For example, for a three-phase line, 22 columns are used.

[`nphase`, `no_I`, `no_U`, `length`, `L/Xl`, `Zc`, `Rm`, `speed`, `Ti`]

Parameter	Description
<code>nphase</code>	Number of phases of the transmission line
<code>no_I</code>	Input number in the source matrix corresponding to the first current source <code>Is_1</code> of the line model. Each line model uses $2*nphase$ current sources specified in the source matrix as follows: <code>Is_1</code> , <code>Is_2</code> ,..., <code>Is_nphase</code> for the sending end followed by <code>Ir_1</code> , <code>Ir_2</code> ,..., <code>Ir_nphase</code> for the receiving end.
<code>no_U</code>	Output number of the state-space corresponding to the first voltage output <code>Vs_1</code> feeding the line model. Each line model uses $2*nphase$ voltage outputs in the source matrix as follows: <code>Vs_1</code> , <code>Vs_2</code> ,..., <code>Vs_nphase</code> for the sending end followed by <code>Vr_1</code> , <code>Vr_2</code> ,..., <code>Vr_nphase</code> for the receiving end.
<code>length</code>	Length of the line (km)
<code>Zc</code>	Vector of the <code>nphase</code> modal characteristic impedances (Ω)
<code>Rm</code>	Vector of the <code>nphase</code> modal series resistances (Ω/km)
<code>speed</code>	Vector of the <code>nphase</code> modal propagation speeds (km/s)
<code>Ti</code>	Transformation matrix from mode to phase currents such that $I_{\text{phase}} = T_i * I_{\text{mod}}$. The $nphase * nphase$ matrix must be given in vector format, [<code>col_1</code> , <code>col_2</code> ,..., <code>col_nphase</code>].

Format of the Yout Matrix

The desired outputs are specified by the matrix `yout`. Each line of the `yout` matrix must be an algebraic expression containing a linear combination of states and state derivatives, specified according to the following format.

Parameter	Description
<code>Uc_bn</code>	Capacitor voltage of branch <i>n</i>
<code>Il_bn</code>	Inductor current of branch <i>n</i>
<code>dUc_bn</code>	Derivative of <code>Uc_bn</code> or <code>Il_bn</code>

Parameter	Description
U_n, I_n	Source voltage or current specified by line n of the source matrix
U_{nx1_x2}	Voltage between nodes $x1$ and $x2 = U_{x1} - U_{x2}$
I_{bn}	Current in branch n flowing from node1 to node2 (See format of RLC matrix). For a parallel RLC branch, I_{bn} corresponds to the total current $I_R + I_L + I_C$.
I_{bn_nx}	Current flowing into node x of a PI transmission line specified by line n of the RLC matrix. This current includes the series inductive branch current and the capacitive shunt current.

Each output expression is built from voltage and current variable names defined above, their derivatives, constants, other variable names, parentheses and operators (+ - * / ^), in order to form a valid MATLAB expression. For example

```
yout =
char(['R1*I_b1+Uc_b3-L2*dIl_b2', 'U_n10_20', 'I2+3*I_b5']);
```

If variable names are used (R1 and L2 in the above example), their names and values must be specified by the two input arguments `vary_name` and `vary_val`.

Sign Conventions for Voltages and Currents

Parameter	Sign Convention
I_{bn}, I_{l_bn}, I_n	Branch current, inductor current of branch n , or current of source # n is oriented from node1 to node2
I_{bn_nx}	Current at one end (node x) of a PI transmission line. If $x = \text{node1}$, the current is entering the line. If $x = \text{node2}$, the current is leaving the line.
U_{c_bn}, U_n	Voltage across capacitor or source voltage ($U_{\text{node1}} - U_{\text{node2}}$)
U_{nx1_x2}	Voltage between nodes $x1$ and $x2 = U_{x1} - U_{x2}$. Voltage of node $x1$ with respect to node $x2$.

Order in Which Outputs Must Be Specified

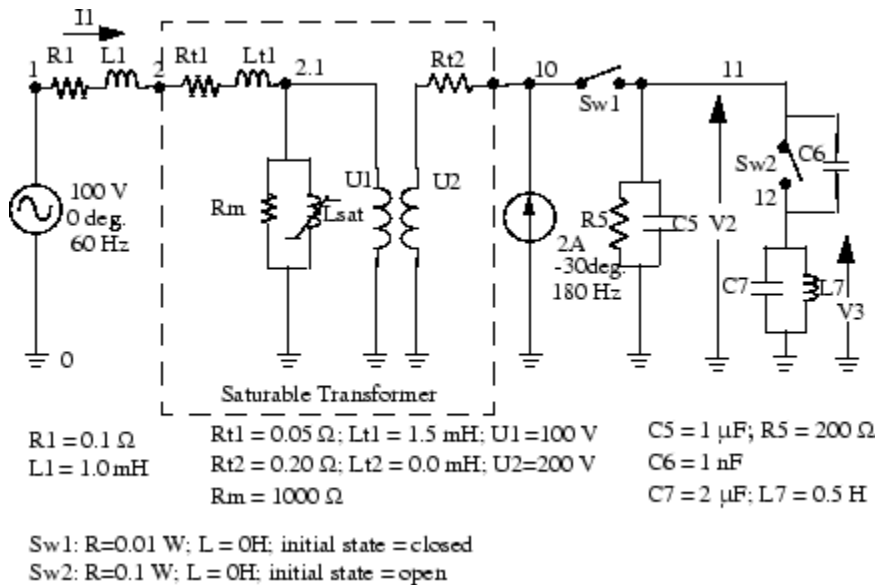
The commands that compute the state-space representation of a system expect the outputs to be in a certain order. You must respect this order in order to obtain correct results. You must be particularly careful if the system contains any switches. The following list gives the proper ordering of outputs:

- 1 The voltages across all switches that have a null inductance ($L_{on} = 0$), if any
- 2 The currents of all switches that have a null inductance ($L_{on} = 0$), if any, in the same order as the voltages above
- 3 The voltages across all nonlinear models that have a finite inductance (switches with $L_{on} > 0$, the magnetizing inductance in saturable transformers, etc.)
- 4 All other voltage and current measurements that you request, in any order

Refer to the Example section below for an example illustrating proper ordering of outputs for a system containing nonlinear elements.

Examples

The following circuit consists of two sources (one voltage source and one current source), two series RLC branches (R1-L1 and C6), two parallel RLC branches (R5-C5 and L7-C7), one saturable transformer, and two switches (Sw1 and Sw2). Sw1 is initially closed whereas Sw2 is initially open. Three measurement outputs are specified (I1, V2, and V3). This circuit has seven nodes numbered 0, 1, 2, 2.1, 10, 11, and 12. Node 0 is used for the ground. Node 2.1 is the internal node of the transformer where the magnetization branch is connected.



Linear State-Space

You can use the `power_statespace` command to find the state-space model of the linear part of the circuit. The nonlinear elements Sw1, Sw2, and Lsat must be modeled separately by means of current sources driven by the voltages appearing across their terminals. Therefore you must provide three additional current sources and three additional voltage outputs for interfacing the nonlinear elements to the linear circuit.

You can find the state-space model of the circuit by entering the following commands in a MATLAB script file. The example is available in the `power_circ2ss` file. Notice that an output text file named `power_circ2ss.net` containing information on the system is requested in the call to `power_statespace`.

```

unit='OMU'; % Units = ohms, mH, and uF

rlc=[
%N1  N2    type    R    L    C(uF)/U(V)
1    2    0      0.1    1    0      %R1 L1
2    0    2      0.05   1.5   100    %transfo Wind.#1
10   0    2      0.20   0    200    %transfo Wind.#2
2.1  0    1     1000    0    0      %transfo mag. branch
11   0    1      200    0    1      %R5 C5
11   12   0      0      0    1e-3    %C6
12   0    1      0      500   2      %L7 C7
];

source=[
%N1  N2    type    U/I    phase    freq
10   11    1      0      0      %Sw1

```

```

11    12    1    0    0    0    %Sw2
2.1    0    1    0    0    0    %Saturation
1    0    0    100    0    60    %Voltage source
0    10    1    2    -30    180    %Current source
];

switches=[
%N1    N2    status    R(ohm)    L(mH)    I#    U#    #
10    11    1    0.01    0    1    1    %Sw1
11    12    0    0.1    0    2    2    %Sw2
];

%outputs
%
% Both switches have Lon=0, so their voltages must be the first outputs,
% immediately followed by their currents (in the same order as the voltages).
% The voltage across all nonlinear models that don't have L=0 follow
% (in this case the saturable transformer's magnetizing inductor).
% The measurements that you request follow, in any order.
%
y_u1='U_n10_11';          %U_Sw1= Voltage across Sw1
y_u2='U_n11_12';          %U_Sw2= Voltage across Sw2
y_i3='I1';                %I1= Switch current Sw1
y_i4='I2';                %I2= Switch current Sw2
y_u5='U_n2.1_0';          %U_sat= Voltage across saturable reactor
y_i6='I_b1';              %I1 measurement
y_u7='U_n11_0';           %V2 measurement
y_u8='U_n12_0';           %V3 measurement

yout=char(y_u1,y_u2,y_i3,y_i4,y_u5,y_i6,y_u7,y_u8);
y_type=[0,0,1,1,0,1,0,0]; %output types; 0=voltage 1=current

% Open file that contains power_statespace output information
fid=fopen('power_circ2ss.net','w');

[A,B,C,D,states,x0,x0sw,rlsw,u,x,y,freq,Asw,Bsw,Csw,Dsw,Hlin]=
power_statespace(rlc,switches,source,[],yout,y_type,unit,[],[],
[],0,fid);

```

Command Line Messages

While `power_statespace` is executing, the following messages are displayed.

```

Computing state space representation of linear electrical circuit
(V2.0)...
(4 states ; 5 inputs ; 7 outputs)

```

```

Oscillatory modes and damping factors:
F=159.115Hz zeta=4.80381e-08

```

```

Steady state outputs @ F=0 Hz :
y_u1= 0Volts
y_u2= 0Volts
y_i3= 0Amperes
y_i4= 0Amperes
y_u5= 0Volts
y_i6= 0Amperes
y_u7= 0Volts
y_u8= 0Volts
Steady state outputs @ F=60 Hz :
y_u1 = 0.009999 Volts < 3.168 deg.
y_u2 = 199.4 Volts < -1.148 deg.
y_i3 = 0.9999 Amperes < 3.168 deg.
y_i4 = 0 Amperes < 0 deg.
y_u5 = 99.81 Volts < -1.144 deg.
y_i6 = 2.099 Amperes < 2.963 deg.
y_u7 = 199.4 Volts < -1.148 deg.
y_u8 = 0.01652 Volts < 178.9 deg.

```

```

Steady state outputs @ F=180 Hz :
y_u1 = 0.00117 Volts < 65.23 deg.
y_u2 = 22.78 Volts < 52.47 deg.
y_i3 = 0.117 Amperes < 65.23 deg.
y_i4 = 0 Amperes < 0 deg.
y_u5 = 11.4 Volts < 53.48 deg.

```

```

y_i6 = 4.027 Amperes < 146.5 deg.
y_u7 = 22.83 Volts < 52.47 deg.
y_u8 = 0.0522 Volts < 52.47 deg.

```

State-Space Output

The names of the state variables are returned in the `states` matrix.

```

states
states =
Il_b2_n2_2.1
Uc_b5_n11_0
Uc_b6_n11_12
Il_b7_n12_0
Il_b1_n1_2*
Uc_b7_n12_0*

```

Although this circuit contains a total of six inductors and capacitors, there are only four state variables. The names of the state variables are given by the first four lines of the `states` matrix. The last two lines are followed by an asterisk indicating that these two variables are a linear combination of the state variables. The dependencies can be viewed in the output file `power_circ2ss.net`.

```

The following capacitor voltages are dependent:
Uc_b7_n12_0 = + Uc_b5_n11_0 - Uc_b6_n11_12
The following inductor currents are dependent:
Il_b1_n1_2 = + Il_b2_n2_0

```

The A,B,C,D matrices contain the state-space model of the circuit without nonlinear elements (all switches open). The `x0` vector contains the initial state values considering the switch Sw1 closed. The `Asw`, `Bsw`, `Csw`, and `Dsw` matrices contain the state-space model of the circuit considering the closed switch Sw1. The `x0sw` vector contains the initial current in the closed switch.

```

A
A =
-4.0006e+05    0    0    0
           0   -4995    0  -499.25
           0  -4992.5    0   4.9925e+05
           0    2    -2    0

```

```

Asw
Asw =
-80.999   -199.99    0    0
4.9947e+05  -5244.7    0   -499.25
4.9922e+05  -5242.1    0   4.9925e+05
           0    2    -2    0

```

The system source frequencies are returned in the `freq` vector.

```

freq
freq =
    0    60   180

```

The corresponding steady-state complex outputs are returned in the (6-by-3) `y` matrix where each column corresponds to a different source frequency.

For example, you can obtain the magnitude of the six voltage and current outputs at 60 Hz as follows.

```

abs(y(:,2))
ans =

```

```
0.0099987
199.42
0.99987
0
99.808
2.0993
199.41
0.016519
```

The initial values of the four state variables are returned in the `x0` vector. You must use this vector in the State-Space block to start the simulation in steady state.

```
x0
x0 =
    2.3302
   14.111
   14.07
  3.1391e-05
```

The initial values of switch currents are returned in `x0sw`. To start the simulation in steady state, you must use these values as initial currents for the nonlinear model simulating the switches.

```
x0sw
x0sw =
    0.16155
    0
```

The Simulink diagram of the circuit is available in the `power_circ2ss_slk` model. If no resistive switches had been used, the linear part of the circuit could have been simulated with the State-Space block of the Simulink/Continuous library. However, as resistive switches are used, the `sfun_psbcontc` S-function is used instead of the State-Space block. This S-function reevaluates the state-space matrices during simulation when the circuit topology is changing (after a switch is opened or closed). Appropriate inputs and outputs are used to connect the switch and saturable reactance models to the linear system. Notice that the status of each switch is fed back from the breaker to the S-function, after the inputs mentioned earlier. You can find the Breaker and Saturable Transformer blocks in the Fundamental Blocks/Elements library containing all the nonlinear continuous models used by Simscape Electrical Specialized Power Systems software. As the breaker model is vectorized, a single block is used to simulate the two switches Sw1 and Sw2.

If you use the Simscape Electrical Specialized Power Systems library to build your circuit, the same Simulink system is generated automatically by the `power_analyze` command. The Simscape Electrical Specialized Power Systems version of this system is also available in the `power_circ2ss_sps` model.

See Also

Distributed Parameters Line | `power_analyze`

power_steadystate

Compute steady-state values of voltages and currents of model

Syntax

```
sps = power_steadystate('sys')
```

Description

`sps = power_steadystate('sys')` returns the steady-state values of the inductor currents and capacitor voltages (electrical state variables), steady-state voltages and currents of measurement blocks and sources, and steady-state voltages and currents of nonlinear blocks of a model.

If the model contains power electronic switches (diodes, thyristors, IGBTs, and so on), the steady-state values are computed considering them as open-circuit. If the model contains breakers (the Breaker, Three-Phase Breaker, or Three-Phase Fault blocks) the steady-state values are computed considering their initial status (open-circuit or short-circuit).

This function returns a structure with these fields:

Field	Description
circuit	The name of the model
Frequencies	Vector listing the source frequencies in the model
States	The names of the electrical states of the model
Xss	The steady-state values of electrical states of the model
DependentStates	The names of the dependent states, if any
Measurements	The names of the measurement blocks in the model
Yss_Measurements	The steady-state values of the measurement blocks in the model
Sources	The names of the electrical source blocks in the model
Uss_Sources	The steady-state values of the electrical sources in the model
NonlinearOutputs	The names of the outputs that correspond to the nonlinear blocks (current-source injection based models). This value consist of the measured terminal voltages used by the nonlinear models to compute the corresponding current sources
Yss_NonlinearOutputs	The steady-state values of the outputs used by the nonlinear blocks
NonlinearInputs	The names of the inputs corresponding to the nonlinear blocks in the model

Field	Description
Uss_NonlinearInputs	The steady-state values of the current inputs defined by the nonlinear blocks (current-source injection based models)

Note When the model contains more than one source frequency, the Xss, Yss_Measurements, Uss_Sources, Yss_NonlinearOutputs, and Uss_NonlinearInputs fields return a matrix of steady-state values. For every source frequency listed in the Frequencies field, there is a corresponding column of steady-states values in the matrix.

See Also

Measurements and States Analyzer

power_zmeter

Compute impedance versus frequency of Impedance Measurement blocks of model

Syntax

```
power_zmeter('sys')
Zdata = power_zmeter('sys',freq)
```

Description

`power_zmeter('sys')` opens the Impedance Measurement Tool app to display the impedance versus frequency defined by the Impedance Measurement blocks. This app can also be activated from the powergui block by selecting **Impedance Measurement**.

`Zdata = power_zmeter('sys',freq)` returns a structure that contains a list of Impedance Measurement blocks and computed impedances versus frequency for the specified frequency range. For example:

```
Zdata = power_zmeter('power_gui',logspace(0,3,500))
Zdata =
```

```
    Blocks: {[1x23 char] 'power_gui/Impedance B1'}
           Z: [500x2 double]
           Freq: [500x1 double]
```


Apps

Cable Parameter Calculator

Compute RLC matrices and frequency-dependent parameters of cable arrangement

Description

The Cable Parameter Calculator app computes the RLGC matrices and the WB frequency-dependent cable parameters of a group of aerial cables, underground cables, or cables in a metal pipe. The cables are single-core and can be made up of any number of concentric conductors and insulators.

The parameters computed by the app can be used in the PI Section Line block, Distributed Parameter Line block, and Distributed Parameter Line Frequency Dependent Line block to model a single-phase or multi-phase cable arrangement.

General Parameters

4-cable configuration used by power_cable example.

-Assuming a flat configuration with distance D between cables

-Geometric Mean Distance between cables (GMD) is: $GMD = D \cdot 12^{1/6} = 1.513 \cdot D$
 $GMD = 0.247 \text{ m} \rightarrow D = 0.161 \text{ m}$

Ground resistivity (ohm.m): 100

Number of cables: 4

Number of cable types: 1

Frequency-Dependent Model Parameters

Frequency range $\text{logspace}(X1, X2, N)$: [-4 5 400]

Cable length (km): 6

RLGC Cable Parameters

Frequency (Hz): 16.7

Cable Configuration: Underground ☐ Cross-bond the sheaths

	V (m)	H (m)	R_out (m)	Cable Type	Phase Number
1	-2	0	0.0389	1	[1 2]
2	-2	0.163	0.0389	1	[3 4]
3	-2	0.326	0.0389	1	[5 6]
4	-2	0.489	0.0389	1	[7 8]

Cable Types: Show cable type: 1 Number of conductors: 2

	R_in (m)	R_out (m)	Rho (ohm.m)	MUE	MUE_IN	EPS_IN	LFCT_IN
1	0	0.01265	2.69e-08	1	1	2.3	0
2	0.0329	0.0349	4.49e-08	1	1	2.25	0

Pipe

Select Cables in a pipe configuration to define pipe parameters

R_in (m): 0.24 MUE: 1 Rho (ohm.m): 1.8e-08

R_out (m): 0.25 MUE_in: 1 MUE_out: 1

R_ext (m): 0.27 EPS_in: 1 EPS_out: 2

V_dpth (m): -2.107 LFCT_in: 0.001 LFCT_out: 0.001

Phase number: 0

Open the Cable Parameter Calculator App

- powergui Block Parameters dialog box: On the **Tools** tab, click **Cable Parameter Calculator**.
- MATLAB command prompt: Enter powerCableParameters.

Parameters

Menu

Parameters|Load typical — Load typical parameters

Load typical cable parameters provided with Simscape Electrical Specialized Power Systems software. This command opens a browser window where you can select typical underground, aerial, or pipe cable configurations. By default, the app is configured with the parameters of the Param_4cablesUnderground2cond.mat file.

Parameters|Load user defined — Load user parameters

Opens a browser window where you can select your own cable data. Select the desired MAT file.

Parameters|Save — Save cable parameters

Saves your cable data by generating a MAT file that contains the app information and cable data.

Configuration|Underground cables — Underground cables configuration

The **Cable Configuration** table assumes that the cables are buried below the ground level. Any previous edited parameters in the table are lost. The position of the cables is specified by **V (m)** and **H (m)** parameters. Aerial cables cannot be defined in this configuration. The title of the table is Cable Configuration: Underground. Default configuration of the app.

Note If you set this configuration, you must possess a license for Optimization Toolbox™ to simulate your model.

Configuration|Aerial cables — Aerial cables configuration

The **Cable Configuration** table assumes that the cables are aerial. Any previously edited parameters in the table are lost. The position of the cables is specified by **V (m)** and **H (m)** parameters. Underground cables cannot be defined in this configuration. The title of the table is Cable Configuration: Aerial.

Configuration|Cables in a pipe — Cables in pipe configuration

The **Cable Configuration** table assumes that the cables are inside a metallic pipe. Any previously edited parameters in the table are lost. The position of the cables is specified by **X (m)** and **Angle (deg)** parameters. Aerial and underground cables cannot be defined in this configuration. The title of the table is Cable Configuration: Inside a pipe.

Help — App documentation

Open the documentation page of the app.

Geometry Tab

General Parameters

User comments — User comments

Free space to write comments related to the cable arrangement modeled by the app.

Ground resistivity (ohm.m) — Ground resistivity

100 (default) | positive scalar

Ground resistivity in ohm-meters.

Number of cables — Number of cables

4 (default) | integer

Specify how many cables are in the cable configuration. A cable is an arrangement of concentric conductors separated with insulators.

If you increase the value of the **Number of Cables** parameter, additional lines are added to the **Cable Configuration** table. The App automatically duplicate the parameter values of the last cable in the table and increment the **Phase Number** values accordingly. Depending on the cable configuration, the **H (m)** parameter value or the **Angle (deg)** parameter value is computed to not overlap position of other cables in the table.

Number of cable types — Number of cable types

1 (default) | integer

Number of cable types. Specify how many different types of cables you have in your cable arrangement.

Frequency-Dependent Model Parameters**Frequency range logspace(X1,X2,N)** — Frequency range logspace(X1,X2,N)

[-4,5,400] (default) | 3-element vector

Specify a vector of three elements to define a frequency vector in the logspace(X1, X2,N) format, consisting of N logarithmically equally spaced points between 10^{X1} and 10^{X2} .

Cable length (km) — Cable length

6 (default) | positive scalar

Specify the length of the cable arrangement in kilometers. All the cables contained in the arrangement have the same length.

Compute — Compute frequency-dependent model parameters button

button

Computes the WB structure variable that contains the frequency-dependent parameters for the cable model. RLGC matrices are also computed at the frequency specified in the **Frequency (Hz)** parameter. After computing the parameters, the app displays the results in the **Computed Parameters** tab.

RLGC Cable Parameters**Frequency (Hz)** — Cable frequency

60 (default) | positive scalar

Frequency in hertz that is used to evaluate RLC parameters.

Compute — Compute RLGC matrices

button

Computes the R, L, G and C matrices for a given cable. After computing the parameters, the app displays the results in the **Computed Parameters** tab.

Cable Configuration Table

Cross-bond the sheaths — Cross-bond sheaths

cleared (default) | ☐

When selected the elements of the RLGC matrices are averaged to reflect cross-bonding the sheaths. The sheath conductors correspond to the conductor no.2 of the cables. The cable can have more than two conductors; however, to use the option all the additional conductors (3,4, ...) of each cables need to be grounded, i.e., associated to phase number 0 in the **Phase Number** parameter vector.

The electric power losses in a cable system dependent on the induced currents flowing in the metallic sheaths. These induced currents depend on the coupling between phase and sheath conductors. For example, in a three-phase cable, if sheaths are not cross-bonded a flat formation will produce higher induced sheath voltages than a trefoil formation. This will in turn result in higher sheath circulating currents and higher losses if the sheaths are bonded (grounded) at both ends. To decrease sheath induced voltage and currents, cross-bonding of sheaths is performed on minor cable sections, like transposition of three-phase aerial transmission lines.

Select this option as an alternative to model all minor sections of cables with physical transposition of sheaths. This will result in approximately same induced voltage and currents in the sheaths while simulating much faster because the model size is reduced, and a longer cable length allows using a larger sample time.

Dependencies

To enable this parameter,

- All the cables in the Cable Configuration table must be set to the same Cable Type value.
- The Cable Type value must correspond to a cable type having two or more conductors.
- All conductors outside the sheath must be grounded (Phase Number value set to 0).

V (m) — Cable vertical position

-2 (default) | scalar

Vertical position of the center of the cable.

When you select **Aerial cables** configuration in the **Configuration** menu, all the V values in the table must be finite positive values. When you select **Underground cables** configuration in the **Configuration** menu, all the V values in the table must be finite negative values.

The app does not allow a mix of aerial and underground cables.

Dependencies

To enable this parameter, select **Aerial cables** or **Underground cables** options in the **Configuration** menu.

H (m) — Cable horizontal position

0 (default) | scalar

Represent the horizontal position of the cable.

H value can be negative, 0, or positive. You can define the horizontal mark position 0 anywhere in your cable arrangement and define the position of the cables according to this mark. For example, the position 0 can be defined in the center of the cable arrangement, so the cables placed on the left will have a negative H (m) value and those on the right will have a positive H (m) value.

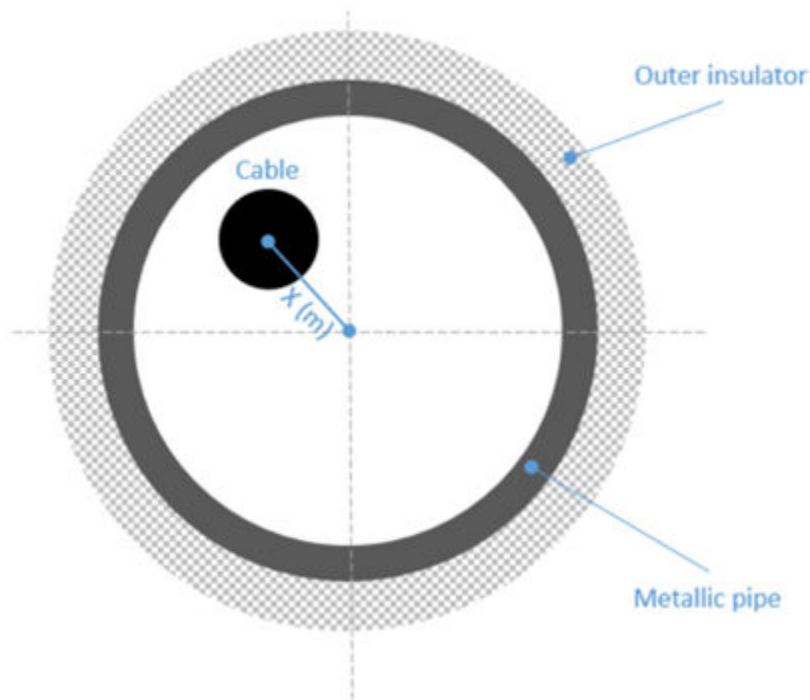
Dependencies

To enable this parameter, select **Aerial cables** or **Underground cables** options in the **Configuration** menu.

X (m) — Cable position inside pipe
0.00331 (default) | positive scalar

Cable position relative to the center of the pipe, in meters.

Specify the position of the center of the cable relative to the center of the pipe.



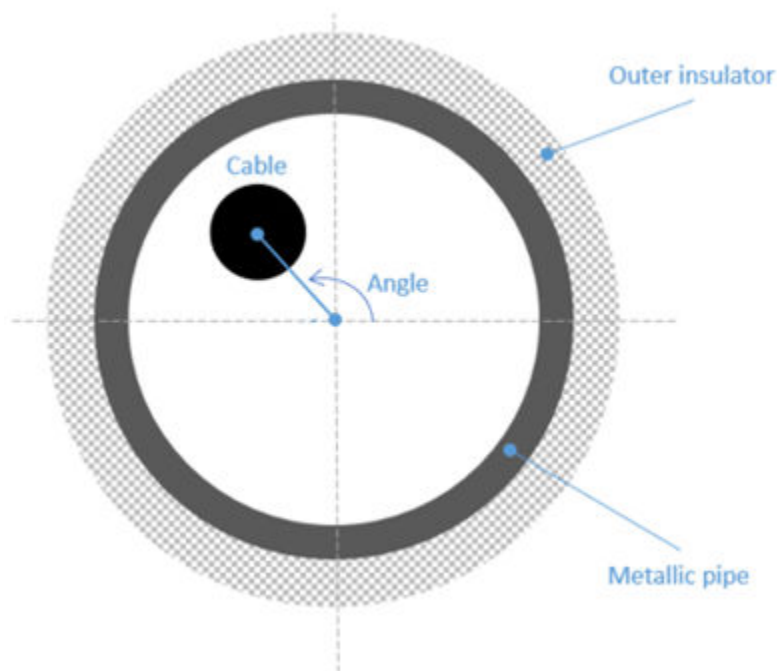
Dependencies

To enable this parameter, select **Cables in a pipe** in the **Configuration** menu.

Angle (deg) — Cable position inside pipe
60 (default) | positive scalar

Position angle of the cable inside the pipe, in degrees.

Specify the angle of the imaginary line that represent the distance between the center of the pipe and the center of the cable.



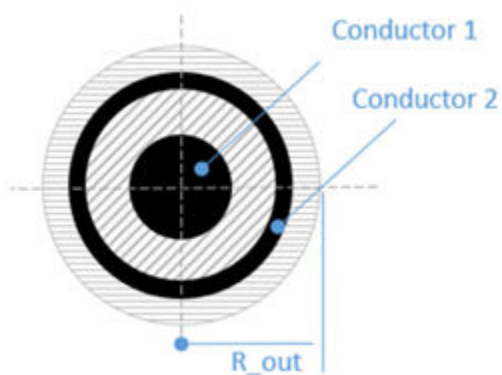
Dependencies

To enable this parameter, select **Cables in a pipe** in the **Configuration** menu.

R_out (m) — Cable external radius
0.0389 (default) | positive scalar

External radius of the cable.

The radius of the of the outer insulation of the cable.



Cable type — Type of cable
1 (default) | positive integer

Value corresponding to the cable type number defined in the **Cable types** table.

Phase Number — Conductor phase numbers
[1,2] (default) | vector of positive integer

Associate each conductor of the cable to the electrical phases of the network. All the conductors for all the cables in the cable configuration need to be associated to a unique phase number. The only exception is phase 0 (meaning that the conductor is grounded) that can be specified for several conductors in several cables.

Phase numbers start with 1 (or 0 if you have grounded conductors) and increase consecutively. The phase numbers do not need to be specified in the numerical order in the table. For example, the following configuration is correct: we have a total of 7 conductors; two are grounded, and all the other conductors are uniquely attributed to a phase number 1,2,3,4, and 5 :

Cable Configuration: Underground ☐ Cross-bond the sheaths

	V (m)	H (m)	R_out (m)	Cable Type	Phase Number
1	-2	0	0.0389	2	3
2	-2	0.161	0.0389	2	1
3	-2	0.322	0.0389	2	2
4	-2	0.483	0.0389	1	[5 0]
5	-2	0.6	0.0389	1	[4 0]

Cable Types Table

Show cable type — Number of cable types

1 (default) | list

Select the cable type number and the **Cable Types** table will show its parameters. The drop-down menu lists all the cable type numbers, from 1 to **Number of cable types**.

Number of conductors— — Number of conductors of selected cable type

2 (default) | positive integer

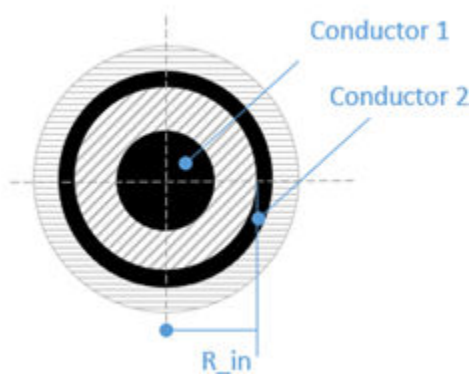
Specify how many concentric conductors exist in the selected cable. When the number of conductors parameter is increased, the app automatically adds a new line in the table with similar parameters as of the previous conductor, and it resizes the **R_in** and **R_out** dimensions proportionally.

R_in (m) — Inside conductor radius

0 (default) | positive scalar

Inside radius of the conductor.

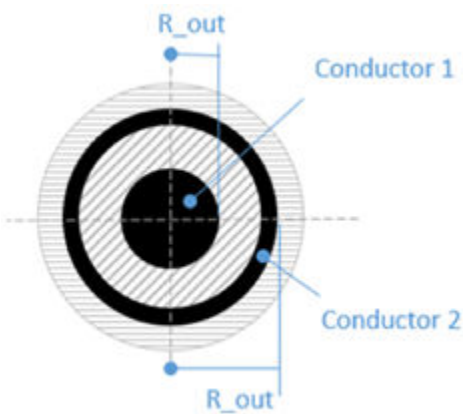
Specify the distance from the center of the cable to the inner part of the conductor. For the first conductor, this value is usually set to 0.



R_{out} (m) — Outside conductor radius
0.01265 (default) | positive scalar

Outside radius of the conductor.

Specify the distance from the center of the cable to the outer part of the conductor.



ρ (Ohm.m) — Conductor resistivity
2.69e-08 (default) | positive scalar

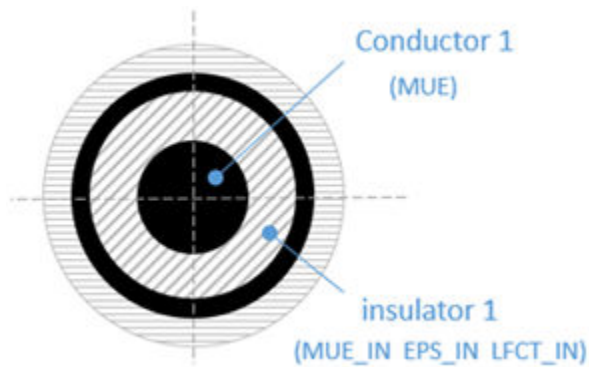
Conductor resistivity.

The resistivity of the conductor in ohm.m

μ — Conductor relative permeability
1 (default) | positive scalar

Conductor relative permeability.

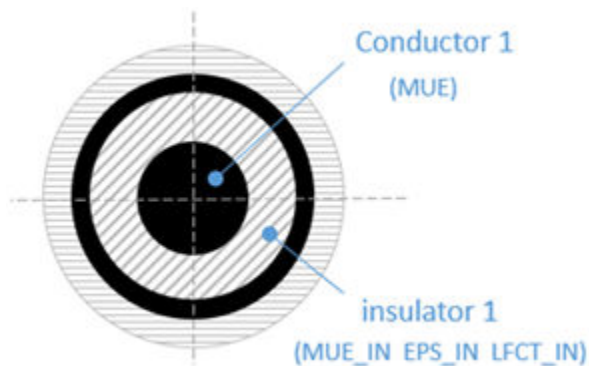
The relative permeability of the conductor.



MUE_IN — Insulation relative permeability
1 (default) | positive scalar

Surrounding insulation relative permeability.

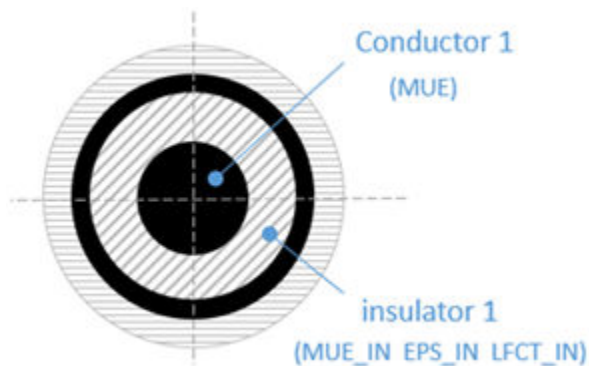
The relative permeability of the insulation that is surrounding the conductor.



EPS_IN — Insulator relative permittivity
2.3 (default) | positive scalar

Surrounding insulation relative permittivity.

The relative permittivity of the insulation that surrounds the conductor.

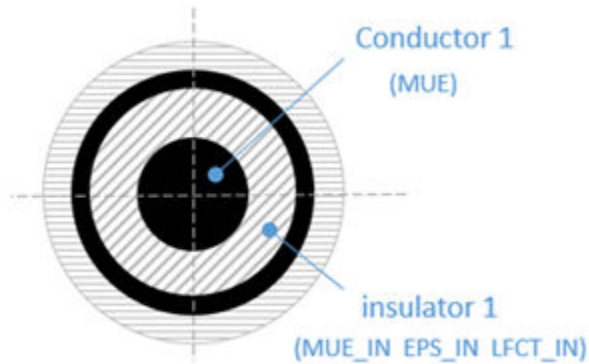


LFCT_IN — Insulator loss factor
0 (default) | positive scalar

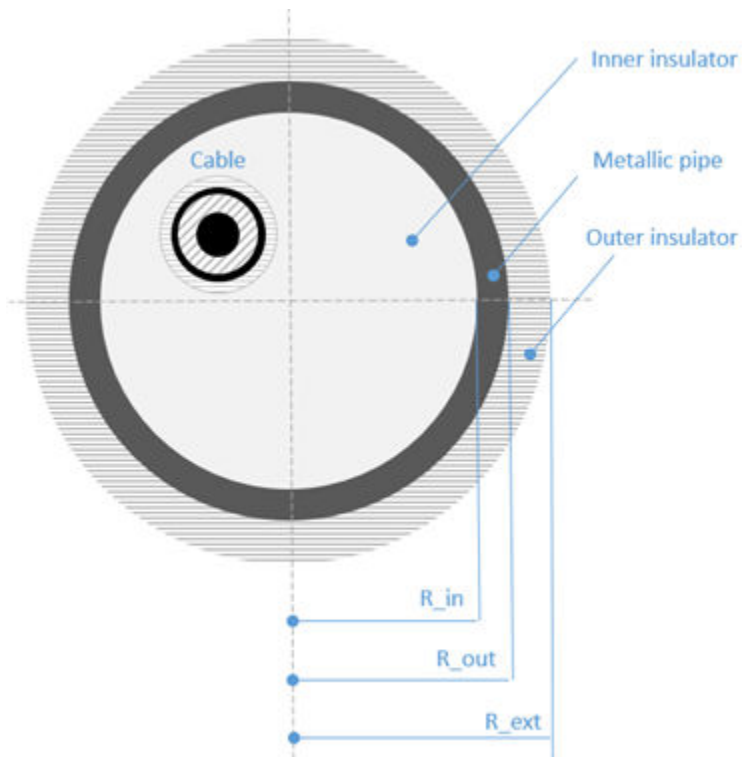
Surrounding insulation loss factor.

The loss factor of the insulation that is surrounding the conductor.

A loss factor of 0.001 means that the active power losses (in watts) inside the dielectric is the 0.1% of the reactive power (in vars) associated with the Conductor1-Conductor 2 capacitance.



Pipe Parameters



R_{in} (m) — Internal radius
0.24 (default) | positive scalar

Internal radius of the metallic pipe in meters.

R_{out} (m) — External metallic pipe radius
0.25 (default) | positive scalar

External radius of the metallic pipe in meters.

R_ext — Pipe external radius
0.27 (default) | positive scalar

External radius of the outer insulator.

V_depth (m) — Pipe depth
-2.107 (default) | negative scalar

Depth of center of the pipe below the ground level.

MUE — Metallic pipe relative permeability
1 (default) | positive scalar

Metallic pipe relative permeability.

MUE_in — Inner insulation relative permeability
1 (default) | positive scalar

Inside insulation relative permeability.

EPS_in — Inner insulation relative permittivity
1 (default) | positive scalar

Inside insulation relative permittivity.

LFCT_in — Inner insulation loss factor
0.001 (default) | positive scalar

Inside insulation loss factor.

Rho (ohm.m) — Metallic pipe resistivity
1.8e-08 (default) | positive scalar

Pipe resistivity.

The resistivity of the metallic pipe.

MUE_out — Outer insulation relative permeability
1 (default) | positive scalar

Outside insulation relative permeability.

EPS_out — Outer insulation relative permittivity
2 (default) | positive scalar

Outside insulation relative permittivity.

LFCT_out — Outer insulation loss factor
0.001 (default) | positive scalar

Outside insulation loss factor.

Phase number — Pipe phase number
7 (default) | integer

Phase number connected to the metallic pipe.

Specify a value of 0 to connect the metallic pipe to the electrical ground. Specify a positive integer value between 1 and N+1, where N is the maximum phase number value that is specified in the **Phase Number** column of the **Cable Configuration** table.

Computer Parameters Tab

Display the R, L, G, and C matrices resulting of the **Compute** buttons.

Send to workspace — Send RLGC, and WB to workspace

button

Sends the R, L, G, C, and WB parameters to the MATLAB workspace. The app creates the variables R, L, G, C, and WB in your workspace.

Block — Block selector

button

Confirms the block selection. The name of the selected block appears in the Send to block edit parameter.

Block Name — Selected block name

char string

Edit box to either manually specify the name of a PI Section Line block, Distributed Parameters Line, or Distributed Parameters Line Frequency Dependent block where the RLGC or WB parameters will be sent when the user click the **Send** button. By default, it displays: - No block specified, and the **Send** button is grayed out and unavailable.

Send to block — Send RLGC matrices to the selected block

button

Sends the RLC parameters to the Distributed Parameter Line or Pi Section Line blocks specified in the **Send to block** parameter. Send the WB parameters via a mat file to the Distributed Parameter Line Frequency Dependent block specified in the **Send to block** parameter.

Version History

Introduced in R2022b

See Also

Apps

Power Line Parameters

Functions

power_cableparam | power_lineparam

Blocks

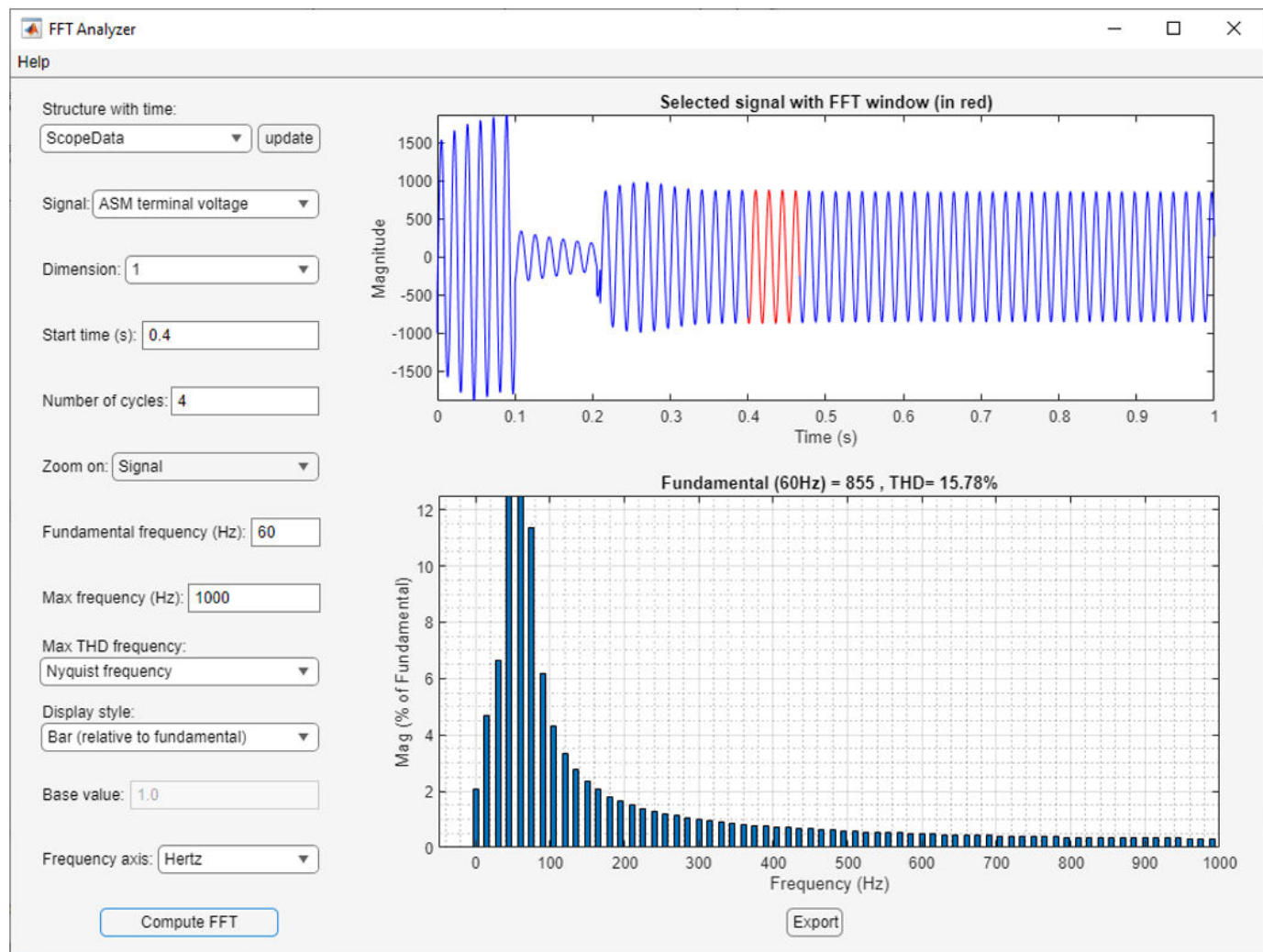
PI Section Line | Distributed Parameters Line | Distributed Parameters Line Frequency-Dependent

FFT Analyzer

Perform Fourier analysis of simulation data signals

Description

The FFT Analyzer app allows you to perform Fourier analysis of simulation data and provides access to all the simulation data that are defined as structure-with-time variables in your workspace. The app displays the spectrum as a bar graph or as a list in percentages relative to a base value or to the DC component of the signal.



Open the FFT Analyzer App

- powergui Block Parameters dialog box: On the **Apps** tab, click **FFT Analyzer**.
- MATLAB command prompt: Enter powerFFT

Parameters

Structure with time (click to update) — Refresh simulation data variables
button

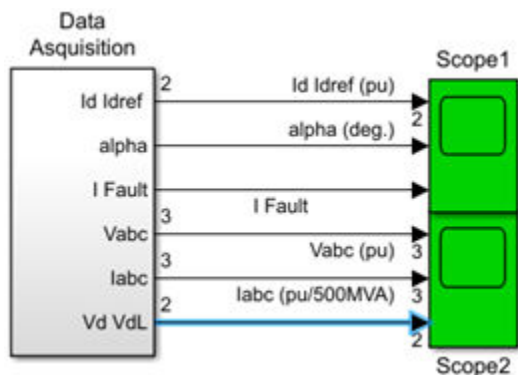
Refreshes the list of simulation data variables in the **Structure with time** drop-down menu. Refreshing allows you to import additional simulation signals from the workspace without the need to close and reopen the tool. If the workspace contains no structure-with-time variables, the drop-down menu shows Empty list.

Structure with time — List of simulation data variables
Empty list (default)

Lists all the simulation data variables that are present in your workspace. Use the drop-down list to select the structure-with-time variable that you want to analyze. If the workspace contains no structure-with-time variables, the list shows Empty list. Structure-with-time variables are generated by the Scope and To Workspace blocks in your model.

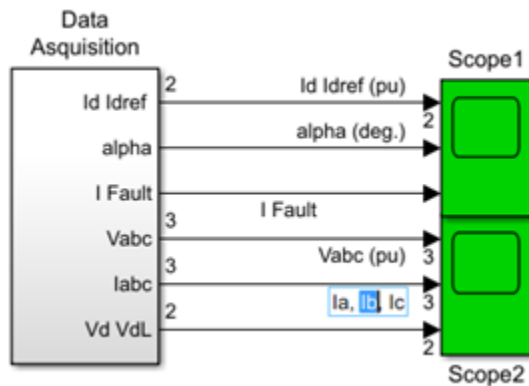
Signal — Selected signal
no default

Lists the signal of the structure-with-time variable specified in the **Structure with time** drop-down menu. Structure-with-time variables that have multiple signals can be generated by a Scope block that has multiple input ports. The list shows the name of the signals as they are labeled in your models. If a signal is not labeled, such as the third signal of Scope2 in this figure, the list shows unnamed (input 3).



Dimension — Index of selected signal
no default

Indicates the index of the signal specified by the **Signal** parameter. For example, the **Dimension** parameter allows you to select the Ib measurement of a three-phase signal connected to the second input of Scope2 block.



Zoom on — Upper plot zoom

Signal (default) | FFT window

Select **Signal** to display the complete view of the signal selected by the **Structure with time**, **Signal**, and **Dimension** parameters in blue in the upper plot. The FFT window that is specified by the **Start time**, **Number of cycles**, and **Fundamental frequency** parameters is displayed in red.

Select **FFT window** to display only the portion of the selected signal where the FFT analysis is performed in the upper plot.

Start time (s) — Start time of the FFT analysis

0 (default) | positive scalar

Specifies the start times for the FFT analysis. The FFT analysis is performed on the portion of the signal starting at the specified time and for the specified number of cycles.

Number of cycles —

1 (default) | positive integer

Specifies the number of cycles of the selected signal to analyze for the FFT analysis.

Fundamental frequency (Hz) —

60 (default) | positive scalar

Specifies the fundamental frequency, in Hz, as a reference frequency for the FFT analysis.

Max frequency (Hz) —

1000 (default) | positive scalar

Specifies the maximum frequency, in Hz, for the FFT analysis.

Max frequency for THD computation —

Nyquist frequency (default) | Same as Max frequency

Select **Nyquist frequency** to compute the THD to the maximum frequency equal to the Nyquist frequency. The Nyquist frequency is half the sampling frequency of the selected signal.

Select **Same as Max frequency** to compute the THD to the maximum frequency equal to the frequency specified in the **Max frequency (Hz)** parameter.

The THD calculation includes all the inter harmonics of the selected input signal.

Display style — Display style for spectrum

Bar (relative to fundamental) (default) | Bar (relative to specified base) | Bar (relative to DC component) | List (relative to fundamental) | List (relative to specified base) | List (relative to DC component)

Select **Bar (relative to fundamental)** to display the spectrum as a bar graph relative to the fundamental frequency. Select **Bar (relative to specified base)** to display the spectrum as a bar graph relative to the base defined by the **Base value** parameter. Select **Bar (relative to DC component)** to display the spectrum as a bar graph relative to the DC component of the signal.

Select **List (relative to fundamental)** to display the spectrum as a list in percentages relative to the fundamental or DC component. Select **List (relative to specified base)** to display the spectrum as a list in percentages relative to the base value defined by the **Base value** parameter. Select **List (relative to DC component)** to display the spectrum as a list relative to the DC component of the signal.

Base value — Harmonics base value

1 (default) | positive scalar

Enter a base value for the display of harmonics.

Frequency axis — Frequency axis of FFT spectrum

Hertz (default) | Harmonic order

Select **Hertz** to display the spectrum frequency axis in hertz. Select **Harmonic order** to display the spectrum frequency axis in harmonic order relative to the fundamental frequency.

Compute FFT —

button

Displays the FFT analysis results for the selected simulation data signal in the **FFT** graph.

Version History

Introduced in R2021a

See Also

Functions

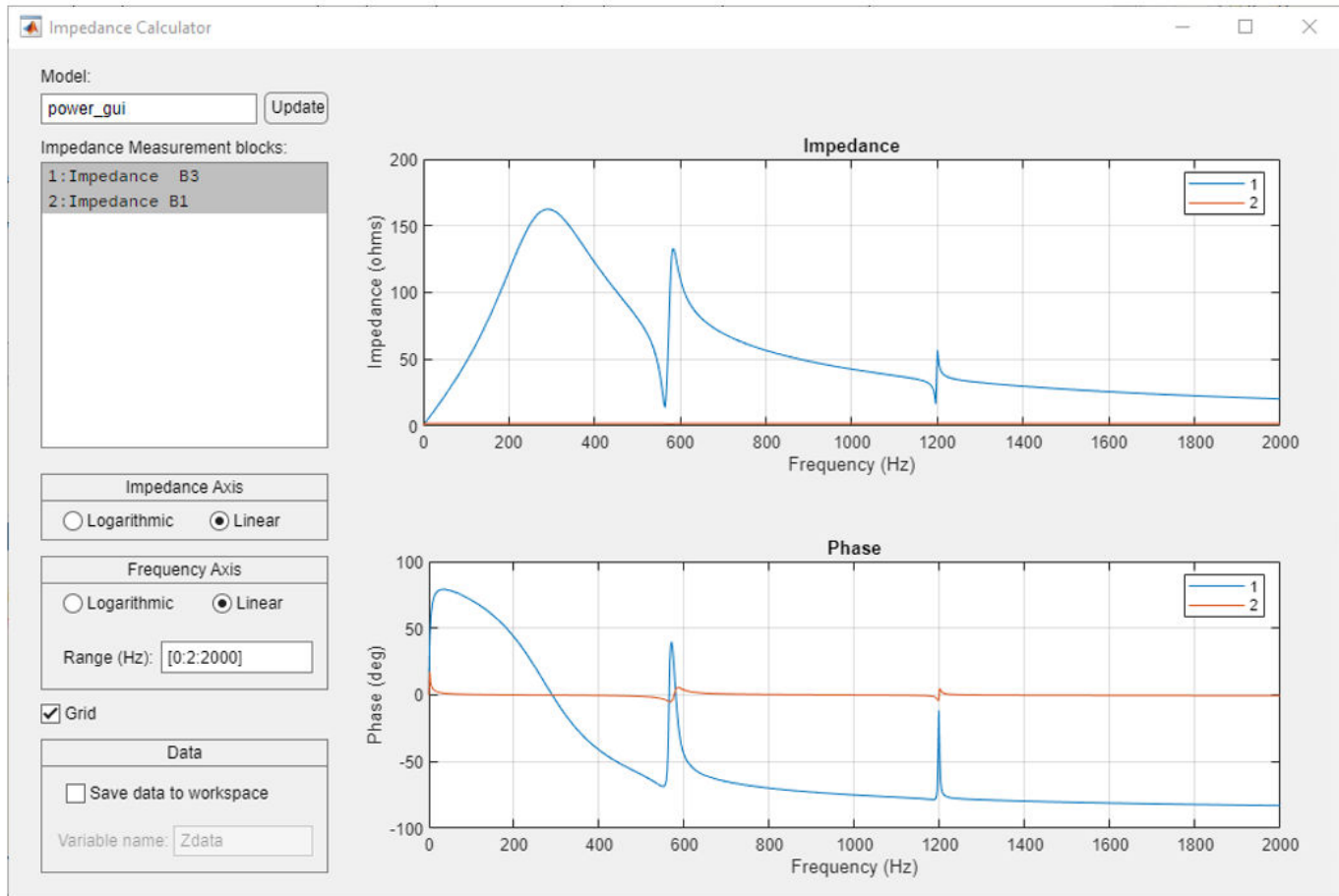
power_fftscope

Impedance Calculator

Compute impedance versus frequency of Impedance Measurement blocks in model

Description

The Impedance Calculator app uses the `power_zmeter` function to compute the impedance versus frequency of the Impedance Measurement blocks in your model.



Open the Impedance Calculator App

- MATLAB command prompt: Enter `powerImpedance(sys)`, where `sys` is a string scalar or character vector that is the name of a model.
- powergui block: Under **Tools**, click **Impedance Calculator**.

Parameters

Model — Name of model
no default

Name of the model under analysis. Enter the name of a model to analyze, then click **Update**.

Update — Display or update
button

Display or update the impedance versus frequency measurement for the model specified in the **Model** parameter. Click this button when you want the app to take into account any change made to the model. Any previously selected block and displayed frequency response are cleared from the app.

Impedance Measurement blocks — Lists Impedance Measurement blocks
no default

Lists the Impedance Measurement blocks in the model. Select the blocks for which you want to obtain the frequency response. Press **Ctrl** to select several impedances to display on the same plot.

Impedance Axis — Choose logarithmic or linear scale
Logarithmic (default) | Linear

Choose logarithmic or linear scale for the vertical impedance scale.

Frequency Axis — Choose logarithmic or linear scale
Logarithmic (default) | Linear

Choose logarithmic or linear scale for the horizontal frequency scales.

Range — Frequency vector
[1:2:500] (default) | vector

Frequency vector, in hertz. You can specify any valid MATLAB expression that defines a vector of frequencies; for example, `0:2:1000` or `linspace(0,1000,500)`.

Save selected Impedance Measurement block data to workspace — Save data in variable
off (default) | on

Select to save the data in a variable in the workspace. The name of the variable is defined by the **Variable name** parameter. The data is saved in the workspace every time you select the Impedance Measurement block in the list.

Variable name — Name of variable
Zdata (default) | string

Name of the variable to save in the workspace.

Dependencies

To enable this parameter, select **Save selected Impedance Measurement block data to workspace**.

Version History

Introduced in R2021b

See Also

Blocks

Impedance Measurement

Load Flow Analyzer

Perform positive-sequence load flow or unbalanced load flow and initialize models containing load flow blocks

Description

The Load Flow Analyzer app uses the `power_loadflow` function and relies on the Newton-Raphson method to provide a robust and fast convergence solution.

The Load Flow Analyzer app allows you to perform two types of load flows:

- Positive-sequence load flow applied to a three-phase system. Positive-sequence voltages and active power (P) and reactive power (Q) flows are computed at each three-phase bus.
- Unbalanced load flow applied to a mix of three-phase, two-phase, and single-phase systems. Individual phase voltage and PQ flow are computed for each phase.

To solve a load flow, you need to determine these four quantities at each bus:

- The net active power P and reactive power Q injected into the bus
- The voltage magnitude V and angle Vangle of bus positive-sequence voltage (positive-sequence voltage or phase voltage)

Before solving the load flow, two of the above quantities are known at every bus and the other two are to be determined. Therefore, the following bus types are used:

- PV bus — For this type of bus, specify P and V. This is a generation bus where a generator, such as a voltage source or three-phase synchronous machine, is connected. The active power P is generated and the generator terminal voltage V is imposed. The load flow solution returns the machine reactive power Q that is required to maintain the reference voltage magnitude V, and the reference voltage angle Vangle.
- PQ bus — At this bus, the specified active power P and reactive power Q are either injected into the bus (generation PQ bus) or absorbed by a load connected at that bus. The load flow solution returns the bus voltage magnitude V and angle Vangle.
- Swing bus — This bus imposes the voltage magnitude V and angle Vangle. The load flow solution returns the active power P and reactive power Q that is generated or absorbed at that bus in order to balance generated power, loads, and losses. At least one bus in the model must be defined as a swing bus, but usually a single swing bus is required unless you have isolated networks. For a positive-sequence load flow, you typically select one synchronous machine or voltage source as a swing bus. For an unbalanced load flow, you can select the three phases of a Three-Phase Voltage Source block or single-phase AC Voltage Source blocks as swing buses.

Use the Load Flow Bus block to define the buses in your model.

If you perform a positive-sequence load flow, you connect a Load Flow Bus block with the **Connectors** parameter specified as `single` to any phase (A, B, or C) of every load flow block in the model. When several load flow blocks are connected together at the same nodes, only one Load Flow Bus block is required to identify the bus.

If you perform an unbalanced load flow, you connect a Load Flow Bus block to all phases of every load flow block in the model. Depending on the number of phases, you need to specify the **Connectors**

parameter by selecting either three connectors (ABC), two connectors (AB, AC, or BC) or a single connector (A, B, or C). When several load flow blocks are connected together at the same nodes, only one Load Flow Bus block is required to identify the bus. In the load flow report, each bus is identified by its **Bus identification** parameter followed by `_a`, `_b`, or `_c`.

Load Flow Blocks for Positive-Sequence Load Flow

Load flow blocks are Simscape Electrical Specialized Power Systems blocks in which you can specify active power (P) and reactive power (Q) to solve the positive-sequence load flow. They are:

- Asynchronous Machine
- Simplified Synchronous Machine
- Synchronous Machine
- Three-Phase Dynamic Load
- Three-Phase Parallel RLC Load
- Three-Phase Series RLC Load
- Three-Phase Programmable Voltage Source
- Three-Phase Source

You specify P and Q in the **Load Flow** tab of the block dialog boxes.

Load Flow Parameters of Three-Phase Sources and Synchronous Machines

The Three-Phase Source and Synchronous Machine blocks allow you to control the generated or absorbed powers P and Q and the positive-sequence terminal voltage. You can set **Generator type** to swing, PV, or PQ.

Load Flow Parameters of Asynchronous Machine Blocks

The Asynchronous Machine blocks require you to specify the mechanical power P_{mec} at the machine shaft.

Load Flow Parameters of the RLC Load Blocks

For the Three Phase RLC Load block, you can specify **Load type** as constant Z (impedance), constant PQ (power), or constant I (current).

Load Flow Parameters of Dynamic Load Blocks

The Three-Phase Dynamic Load block dialog box does not have a **Load Flow** tab. The load is always considered as a constant PQ load. P and Q are the initial active and reactive power P_o , Q_o that you specify by using the **Active and reactive power at initial voltage [Po(W) Qo(var)]** parameter. The **Initial positive-sequence voltage Vo [Mag(pu) Phase (deg.)]** parameter (Mag and Phase) updates according to the load flow solution.

Load Flow Blocks for Unbalanced Load Flow

Load flow blocks are Simscape Electrical Specialized Power Systems blocks in which you can specify active power (P) and reactive power (Q) to solve the load flow at each phase of every bus. They are:

- AC Voltage Source
- Asynchronous Machine
- Parallel RLC Load

- Series RLC Load
- Synchronous Machine
- Three-Phase Dynamic Load
- Three-Phase Parallel RLC Load
- Three-Phase Series RLC Load
- Three-Phase Source

You specify P and Q in the **Load Flow** tab of the block dialog boxes.

Load Flow Parameters of Single-Phase and Three-Phase Sources

The single-phase AC Voltage Source block allows you to control the generated or absorbed powers P and Q and the terminal voltage. The Three-Phase Source block allows you to control the generated or absorbed powers P and Q and the terminal voltages for each phase (A, B, and C). For these two blocks, you can set **Generator type** to swing, PV, or PQ.

Load Flow Parameters of Synchronous Machine

The Three-Phase Synchronous Machine block allows you to control generated or absorbed powers P and Q (total of phases A, B, and C) and its positive-sequence terminal voltage. You can set **Generator type** to PV or PQ.

Load Flow Parameters of Asynchronous Machine Blocks

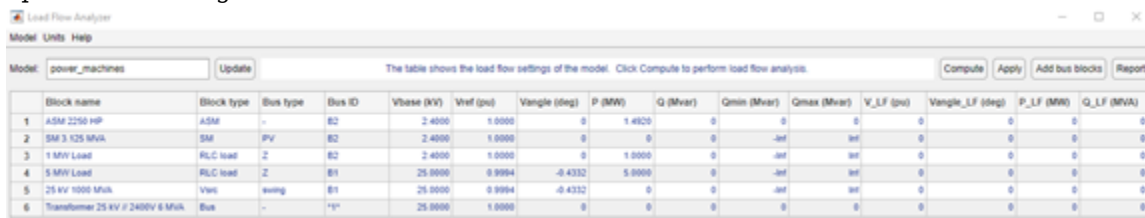
The Asynchronous Machine blocks require you to specify the mechanical power P_{mec} developed in positive-sequence at the machine shaft.

Load Flow Parameters of the RLC Load Blocks

You can specify the **Load type** parameter for the single-phase and three-phase RLC Load blocks as constant Z (impedance), constant PQ (power), or constant I (current). You can connect single-phase loads phase-to-ground or phase-to-phase. You can connect three-phase loads connected in Wye (grounded or floating) or delta.

Load Flow Parameters of Dynamic Load Blocks

The Three-Phase Dynamic Load block dialog box does not have a **Load Flow** tab. The load is always considered as a constant PQ load. P and Q are the initial active and reactive power P_o , Q_o that you specify by using the **Active and reactive power at initial voltage [Po(W) Qo(var)]** parameter. The **Initial positive-sequence voltage V_o [Mag(pu) Phase (deg.)]** parameter (Mag and Phase) updates according to the load flow solution.



The screenshot shows the 'Load Flow Analyzer' application window. At the top, there's a menu bar with 'Model', 'Units', and 'Help'. Below it, a toolbar contains buttons for 'Update', 'Compute', 'Apply', 'Add bus blocks', and 'Report'. A status bar indicates 'The table shows the load flow settings of the model. Click Compute to perform load flow analysis.' The main area displays a table with 14 columns: Block name, Block type, Bus type, Bus ID, Vbase (KV), Vref (pu), Vangle (deg), P (MW), Q (Mvar), Qmin (Mvar), Qmax (Mvar), V_LF (pu), Vangle_LF (deg), P_LF (MW), and Q_LF (MVA). The table lists six blocks: ASD 2250 HP, SM 3 125 MVA, 1 MW Load, 5 MW Load, 25 KV 1000 MVA, and Transformer 25 KV # 2400V 6 MVA.

Block name	Block type	Bus type	Bus ID	Vbase (KV)	Vref (pu)	Vangle (deg)	P (MW)	Q (Mvar)	Qmin (Mvar)	Qmax (Mvar)	V_LF (pu)	Vangle_LF (deg)	P_LF (MW)	Q_LF (MVA)
1 ASD 2250 HP	ASD	-	B2	2.4000	1.0000	0	1.4620	0	0	0	0	0	0	0
2 SM 3 125 MVA	SM	PV	B2	2.4000	1.0000	0	0	0	-inf	inf	0	0	0	0
3 1 MW Load	RLC load	Z	B2	2.4000	1.0000	0	1.0000	0	-inf	inf	0	0	0	0
4 5 MW Load	RLC load	Z	B1	25.0000	0.9984	-0.4332	5.0000	0	-inf	inf	0	0	0	0
5 25 KV 1000 MVA	Var	swing	B1	25.0000	0.9984	-0.4332	0	0	-inf	inf	0	0	0	0
6 Transformer 25 KV # 2400V 6 MVA	Bus	-	*1*	25.0000	1.0000	0	0	0	0	0	0	0	0	0

Open the Load Flow Analyzer App

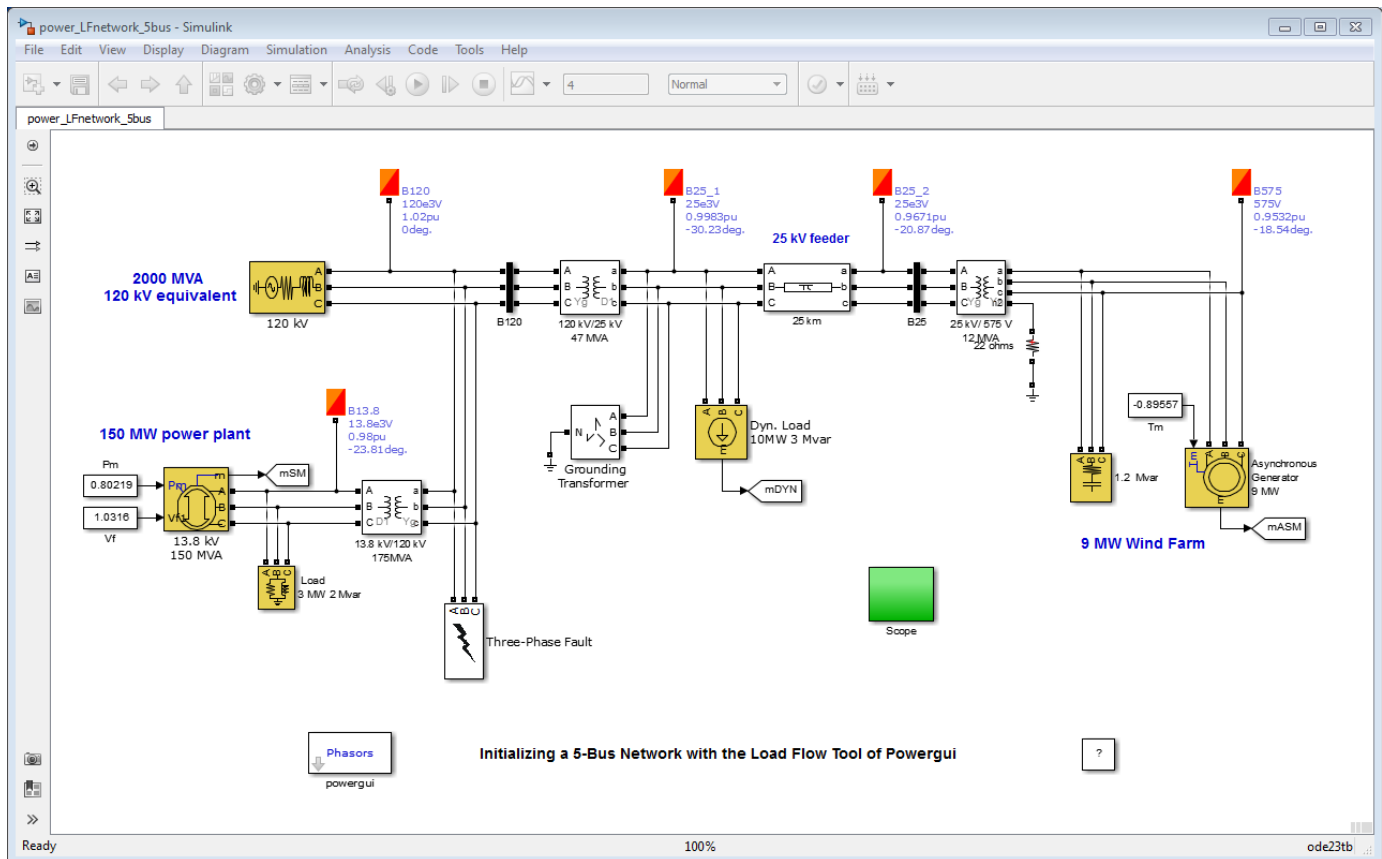
- MATLAB command prompt: Enter powerLoadFlow

- powergui Block Parameters dialog box: On the **Tools** tab, click **Load Flow Analyzer**.
- To perform a load flow analysis and initialize your model so that it starts in steady state:
 - 1 Define the model buses using Load Flow Bus blocks.
 - 2 Specify the load flow parameters of all blocks that have load flow parameters. These blocks are referred to as load flow blocks.
 - 3 Solve the load flow and interactively modify the load flow parameters until a satisfactory solution is obtained.
 - 4 Save the load flow parameters and machine initial conditions in the model.

Examples

Positive-Sequence Load Flow

Open the “Initializing a 5-Bus Network with the Load Flow Tool of Powergui” example to open a model containing five Load Flow Bus blocks and six load flow blocks.



The Load Flow Bus blocks are shown in orange and the load flow blocks are shown in yellow.

The Load Flow Bus blocks specify the bus base voltages (nominal phase-to-phase rms voltage). They also specify the voltage at PV buses or the voltage and angle of the swing buses. Once the load flow is solved, the Load Flow Bus block displays the bus positive-sequence voltage magnitude and phase angle as block annotations.

The bus type (PV, PQ, or swing) is determined by the load flow blocks connected to the bus. If you have several load flow blocks with different types (specified in the **Generator type** parameter or in the **Load type** parameter) connected to the same bus, the Load Flow tool determines the resulting bus type (swing, PQ, or PV).

In the “Initializing a 5-Bus Network with the Load Flow Tool of Powergui” example, the bus types are determined as follows:

Bus	Load Flow Blocks	Resulting Bus Type
B120	120 kV Three-Phase Source - Generator type = swing	swing V=1.02 p.u. 0 deg. Specify the voltage and angle in the B120 Load Flow Bus block.
B13.8	13.8 kV 150 MVA Synchronous Machine - Generator type = PV 3 MW 2 Mvar RLC Load - Load type = constant PQ	PV P = 117 MW V = 0.98 pu Specify the voltage in the B13.8 Load Flow Bus block.
B25_1	10 MW, 3 Mvar Dynamic Load - Implicit load type = constant PQ	PQ P = -10 MW Q = -3 Mvar
B25_2	No load flow blocks	PQ P = 0 MW Q = 0 Mvar
B575	Asynchronous generator 9 MW 1.2 Mvar RLC Load - Load type = constant Z	PQ P = 0 MW Q = 0 Mvar Constant Z load is included in the Ybus admittance matrix.

Some restrictions apply when you connect several source blocks and synchronous machines at the same bus:

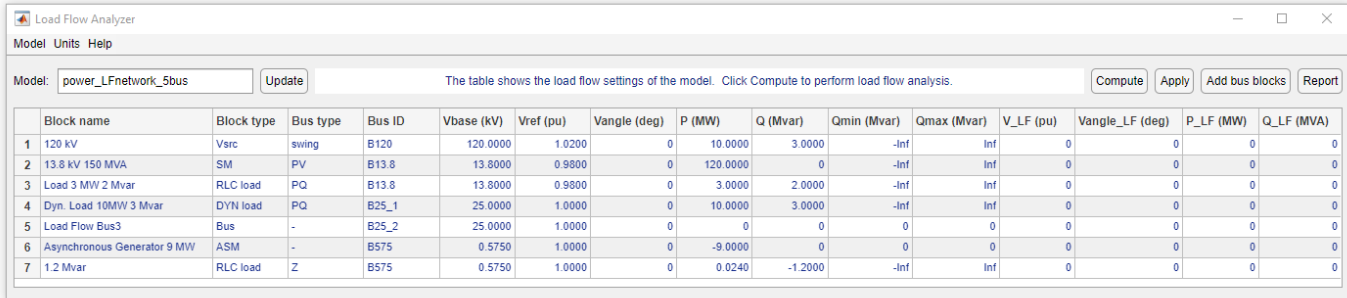
- Two swing generators cannot be connected in parallel.
- A swing generator cannot be connected in parallel with a PV ideal voltage source.
- When a swing voltage source with RL impedance is connected to a PV generator, the swing bus is automatically moved to the ideal voltage source connection node, behind the RL source impedance.
- Only one PV generator with finite Q limits can be connected at a generation bus. However, you may have other PQ generators and loads connected on the same bus.

For more information on how to use the Load Flow Bus block in your model, see the Load Flow Bus page.

Using the Load Flow Tool to Perform Load Flow Analysis

Once you have entered the load flow parameters in the Load Flow Bus blocks and in the various load flow blocks, open the Load Flow Analyzer by clicking the **Load Flow Analyzer** button of the

powergui block. The tool displays a summary of the load flow data of the model. The table below shows the data found in the “Initializing a 5-Bus Network with the Load Flow Tool of Powergui” example.



The screenshot shows the 'Load Flow Analyzer' window. At the top, there's a menu bar with 'Model', 'Units', and 'Help'. Below it, a 'Model' dropdown is set to 'power_LFnetwork_5bus'. To the right of the dropdown is an 'Update' button. Further right is a text box containing the message: 'The table shows the load flow settings of the model. Click Compute to perform load flow analysis.' To the right of this text box are four buttons: 'Compute', 'Apply', 'Add bus blocks', and 'Report'. Below these elements is a table with 16 columns and 7 rows of data.

	Block name	Block type	Bus type	Bus ID	Vbase (kV)	Vref (pu)	Vangle (deg)	P (MW)	Q (Mvar)	Qmin (Mvar)	Qmax (Mvar)	V_LF (pu)	Vangle_LF (deg)	P_LF (MW)	Q_LF (MVA)
1	120 kV	Vsrc	swing	B120	120.0000	1.0200	0	10.0000	3.0000	-Inf	Inf	0	0	0	0
2	13.8 kV 150 MVA	SM	PV	B13.8	13.8000	0.9800	0	120.0000	0	-Inf	Inf	0	0	0	0
3	Load 3 MW 2 Mvar	RLC load	PQ	B13.8	13.8000	0.9800	0	3.0000	2.0000	-Inf	Inf	0	0	0	0
4	Dyn. Load 10MW 3 Mvar	DYN load	PQ	B25_1	25.0000	1.0000	0	10.0000	3.0000	-Inf	Inf	0	0	0	0
5	Load Flow Bus3	Bus	-	B25_2	25.0000	1.0000	0	0	0	0	0	0	0	0	0
6	Asynchronous Generator 9 MW	ASM	-	B575	0.5750	1.0000	0	-9.0000	0	0	0	0	0	0	0
7	1.2 Mvar	RLC load	Z	B575	0.5750	1.0000	0	0.0240	-1.2000	-Inf	Inf	0	0	0	0

Note that the table contains seven lines, but there are only six load flow blocks in the model. This is because the bus B25_2 is not connected to any load flow block. Line 5 is added in the table for that particular bus, so that you can see all buses listed together with their bus voltages. This bus will be considered in the load flow analysis as a PQ bus with zero P and Q.

The **Block name** column identifies the block type. The **Block type** column displays the bus type of the load flow blocks. The next four columns give the bus identification label, the bus base voltage, the reference voltage (in pu of base voltage), and the voltage angle of the load flow bus where the block is connected. The following columns are the P and Q values specified in the **Load Flow** tab of the blocks.

The last four columns display the current load flow solution. Because the load flow has not been performed, the columns display zero values.

The load flow parameters in the **Preferences** tab of the powergui block are used to build the Ybus network admittance matrix and to solve the load flow. The base power is used to specify units of the normalized Ybus matrix in pu/Pbase and bus base voltages. The “Initializing a 5-Bus Network with the Load Flow Tool of Powergui” model contains five buses; consequently, the Ybus matrix will be a 5-by-5 complex matrix evaluated at the frequency specified by the **Frequency (Hz)** parameter.

The load flow algorithm uses an iterative solution based on the Newton-Raphson method. The **Max iterations** parameter defines the maximum number of iterations. The load flow algorithm will iterate until the P and Q mismatch at each bus is lower than the **PQ tolerance** parameter (in pu/Pbase). The power mismatch is defined as the difference between the net power injected into the bus by generators and PQ loads and the power transmitted on all links leaving that bus.

To avoid a badly conditioned Ybus matrix, you should select a **Base power** parameter value in the range of nominal powers and loads connected to the network. For a transmission network with voltages ranging from 120 kV to 765 kV, a 100 MVA base is usually selected. For a distribution network or for a small plant consisting of generators, motors, and loads that have a nominal power in the range of hundreds of kilowatts, a 1 MVA power base is better adapted.

To solve the load flow, click the **Compute** button. The load flow solution is displayed in the last five columns of the table.

V_LF (pu)	Vangle_LF (deg)	P_LF (MW)	Q_LF (MVA)
1.0200	0	-114.3673	62.7471
0.9800	-23.8133	120.0000	-37.2482
0.9800	-23.8133	3.0000	2.0000
0.9983	-30.2257	10.0000	3.0000
0.9671	-20.8696	0	0
0.9532	-18.5436	-8.9012	4.3805
0.9532	-18.5436	0.0218	-1.0903

To display the load flow report showing power flowing at each bus, click the **Report** button. Save this report in a file by specifying the file name at the prompt.

The report displays the summary of active and reactive powers, including the total PQ sharing between generators (SM- and Vsrc-type blocks), PQ loads (PQ-type RLC loads and DYN loads), shunt constant Z loads (Z-type RLC loads and magnetizing branches of transformers), and asynchronous machine loads (ASM):

The Load Flow converged in 2 iterations !

SUMMARY for subnetwork No 1

```

Total generation :    P=  5.61 MW    Q= 25.51 Mvar
Total PQ load   :    P= 13.00 MW    Q=  5.00 Mvar
Total Zshunt load :  P=  0.68 MW    Q= -0.51 Mvar
Total ASM load  :    P= -8.90 MW    Q=  4.38 Mvar
Total losses   :    P=  0.83 MW    Q= 16.64 Mvar

```

The Total losses line represents the difference between generation and loads (PQ type + Z type +ASM) and represents series losses. After this summary, a voltage and power report is presented for each bus:

```

1 : B120  V= 1.020 pu/120kV 0.00 deg ; Swing bus
    Generation : P= -114.39 MW Q=  62.76 Mvar
    PQ_load    : P=   0.00 MW Q=   0.00 Mvar
    Z_shunt     : P=   0.25 MW Q=   0.23 Mvar
--> B13.8     : P= -116.47 MW Q=  53.89 Mvar
--> B25_1     : P=   1.84 MW Q=   8.63 Mvar

2 : B13.8  V= 0.980 pu/13.8kV -23.81 deg
    Generation : P=  120.00 MW Q= -37.25 Mvar
    PQ_load    : P=   3.00 MW Q=   2.00 Mvar
    Z_shunt     : P=   0.17 MW Q=   0.17 Mvar
--> B120      : P=  116.83 MW Q= -39.42 Mvar

3 : B25_1  V= 0.998 pu/25kV -30.22 deg
    Generation : P=   0.00 MW Q=   0.00 Mvar
    PQ_load    : P=  10.00 MW Q=   3.00 Mvar
    Z_shunt     : P=   0.25 MW Q=   0.21 Mvar
--> B120      : P=  -1.83 MW Q=  -8.44 Mvar
--> B25_2     : P=  -8.41 MW Q=   5.23 Mvar

4 : B25_2  V= 0.967 pu/25kV -20.85 deg
    Generation : P=   0.00 MW Q=   0.00 Mvar
    PQ_load    : P=  -0.00 MW Q=  -0.00 Mvar
    Z_shunt     : P=   0.01 MW Q=  -0.03 Mvar

```

```

--> B25_1      : P=    8.87 MW Q=   -3.67 Mvar
--> B575       : P=   -8.88 MW Q=    3.70 Mvar

5 : B575  V= 0.953 pu/0.575kV -18.51 deg
    Generation : P=    0.00 MW Q=    0.00 Mvar
    PQ_load    : P=   -0.00 MW Q=   -0.00 Mvar
    Z_shunt     : P=    0.01 MW Q=   -1.09 Mvar
--> ASM        : P=   -8.90 MW Q=    4.38 Mvar
--> B25_2      : P=    8.89 MW Q=   -3.29 Mvar

```

For every bus, the bus voltage and angle are listed on the first line. The next three lines give the PQ generated at the bus (all SM and voltage sources), the PQ absorbed by the PQ type loads, and the PQ absorbed by the Z-type loads.

The last lines, preceded by an arrow (-->), list the PQ transmitted to neighbor buses connected through lines, series impedances, and transformers, as well power absorbed by ASM.

Apply the Load Flow Solution to Your Model

When performing a load flow analysis, you may need to iterate on P, Q, and V values until you find satisfactory voltages at all buses. This may require, for example, changing generated power, load powers, or reactive shunt compensation.

To change the load flow setup, you need to edit the parameters of the load flow blocks and the Load Flow Bus blocks. Then click the **Update** button to refresh the load flow data displayed by the table in the Load Flow Analyzer. The previous load flow solution is then deleted from the table. Click the **Compute** button to obtain a new load flow solution that corresponds to the changes you made.

Once you have obtained a satisfactory load flow, update the model initial conditions according to the load flow solution. Click the **Apply to Model** button to initialize the machine blocks of the model, and as the initial conditions of regulators connected to the machines.

Open the Three-Phase Parallel RLC Load block connected at the B13.8 bus. Because the **Load type** specified in the **Load Flow** tab is constant PQ, the nominal voltage of this block has been changed to the corresponding bus voltage of 0.98 pu. The **Nominal phase-to-phase voltage Vn (Vrms)** parameter is set to $(13800) * 0.98$.

Open the Three-Phase Dynamic Load block connected at the B25_1 bus. The **Initial positive-sequence voltage Vo [Mag(pu) Phase (deg.)]** is set to $[0.998241 \ -30.2228]$.

Note that the voltage magnitudes and angles obtained at each bus are written as block annotations under the Load Flow Bus blocks.

Open the Scope block and start the simulation.

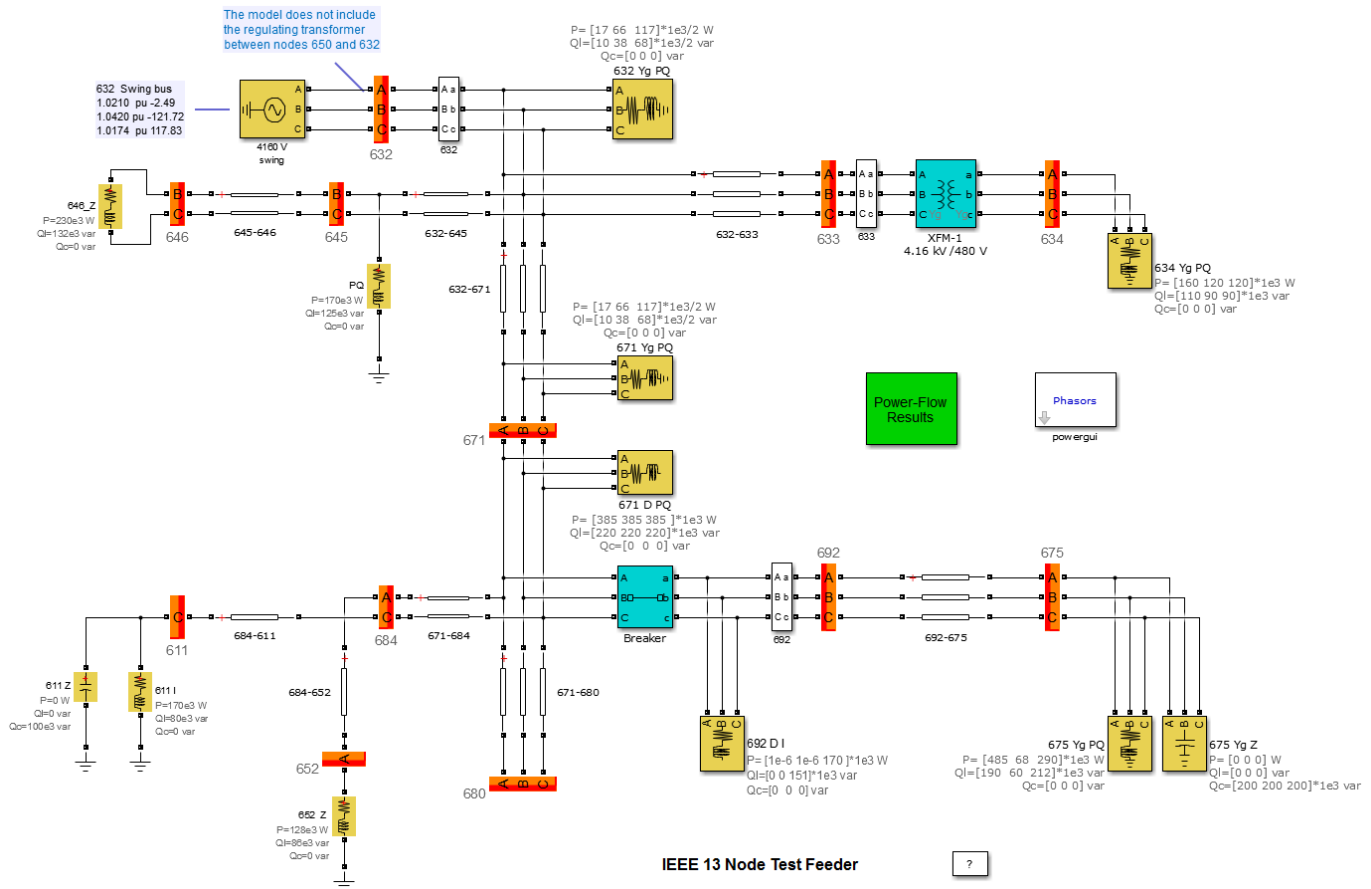
The Three-Phase Fault block applies a six-cycle fault at the B120 bus.

Observe the waveforms of the SM active power, SM and ASM speeds, and PQ of DYN load, and notice that simulation starts in steady state.

Example of Unbalanced Load Flow

Open the “IEEE 13 Node Test Feeder” to open a model containing 12 Load Flow Bus blocks and 13 load flow blocks. This model is a benchmark network taken from the Radial Distribution Test Feeder in the Distribution System Analysis Subcommittee Report by the Power Engineering Society in pages 908-912 written in 2001.

The original benchmark system contains 13 nodes. However, because the “IEEE 13 Node Test Feeder” model does not include the regulating transformer, it contains only 12 nodes.



The Load Flow Bus blocks are shown in orange and the Load Flow blocks are shown in yellow.

The Load Flow Bus blocks specify the bus base voltages (nominal phase-to-ground rms voltage). They specify the voltage at PV buses or the voltage and angle of the swing buses. Once the load flow is solved, the Load Flow Bus block displays the bus voltage magnitude and phase angle as block annotations.

Note By default, the block annotations are set in the **Block Annotation** tab of the Load Flow Bus block properties to display the phase A magnitude (<VLF> parameter) and the phase A angle (<angleLF> parameter). To display phase B magnitude and angle, specify <VLFb> and <angleLFB>, respectively. To display phase C magnitude and angle, specify <VLFc> and <angleLFC>, respectively.

You can also delete some block annotations. In the “IEEE 13 Node Test Feeder” example, only the bus identification is displayed (<ID> parameter).

The bus type (PV, PQ, or swing) is determined by the load flow blocks connected to the bus. If you have several load flow blocks with different types (specified in the **Generator type** parameter or in the **Load type** parameter) connected to the same bus, the Load Flow tool determines the resulting

bus type (swing, PQ, or PV). The table shows how the bus types are determined for some of the model buses of the “IEEE 13 Node Test Feeder” example.

Bus	Load Flow Blocks	Resulting Bus Type
632	4160 V swing - Generator type = swing	632_a=swing V=1.0210 pu -2.49 deg. 632_b=swing V=1.042 pu -121.72 deg. 632_c=swing V=1.074 pu -121.72 deg. Voltages and angles are specified in the ‘632’ Load Flow Bus block.
633	No load flow block	PQ 633_a -> P= 0 kW; Q = 0 kvar 633_b -> P= 0 kW; Q = 0 kvar 633_c -> P= 0 kW; Q = 0 kvar
634	634 Yg PQ load block - Load type = constant PQ	PQ 634_a -> P= 160 kW; Q = 110 kvar 634_b -> P= 120 kW; Q = 90 kvar 634_c -> P= 120 kW; Q = 90 kvar
646	646_Z load block - Load type = constant Z - Load connection ‘bc’	PQ 646_bc -> P= 0 MW Q = 0 Mvar Constant Z loads are included in the Ybus admittance matrix.
675	675 Yg PQ load - Load type = constant PQ 675 Yg Z load - Load type = constant Z	PQ 675_a -> P= 485 kW; Q = 190 kvar 675_b -> P= 68 kW; Q = 60 kvar 675_c -> P= 290 kW; Q = 212 kvar Constant Z loads are included in the Ybus admittance matrix.

Some restrictions apply when you have several source blocks and synchronous machines connected to the same load flow bus:

- You cannot connect two swing generators in parallel.
- You cannot connect a swing generator in parallel with a PV ideal voltage source
- You can connect only one PV generator with finite Q limits at a generation bus. However, you can have other PQ generators and loads connected on the same bus.

For more information on how to use the Load Flow Bus block in your model, see Load Flow Bus block.

Open the Load Flow Tool to Perform Load Flow Analysis

Open the Load Flow Analyzer by clicking the **Load Flow Analyzer** button in the powergui block. The tool displays a list of the individual single-phase buses (one bus per phase) found in the “IEEE 13 Node Test Feeder” model. In the Load Flow Analyzer, the load flow has not yet been performed, so the **V_LF (pu)** and **Vangle_LF (deg)** columns display zero values.

Bus ID	V_LF (pu)	Vangle_LF (deg)
1 611_s	0	0
2 632_s	0	0
3 632_s	0	0
4 632_s	0	0
5 633_s	0	0
6 633_s	0	0
7 633_s	0	0
8 634_s	0	0
9 634_s	0	0
10 634_s	0	0
11 646_s	0	0
12 646_s	0	0
13 646_s	0	0
14 646_s	0	0
15 671_s	0	0
16 671_s	0	0
17 671_s	0	0
18 671_s	0	0
19 671_s	0	0
20 675_s	0	0
21 675_s	0	0
22 680_s	0	0
23 680_s	0	0
24 680_s	0	0
25 684_s	0	0
26 684_s	0	0
27 682_s	0	0
28 682_s	0	0
29 682_s	0	0

The load flow parameters in the **Preferences** tab of the powergui block are used to build the Ybus network admittance matrix and to solve the load flow. The base power is used to specify units of the normalized Ybus matrix in pu/Pbase and bus base voltages. The “IEEE 13 Node Test Feeder” model contains 29 single phase buses; consequently, the Ybus matrix is a 29-by-29 complex matrix evaluated at the frequency specified by the **Frequency (Hz)** parameter.

The load flow algorithm uses an iterative solution based on the Newton-Raphson method. The **Max iterations** parameter defines the maximum number of iterations. The load flow algorithm iterates until the P and Q mismatch at each bus is lower than the **PQ tolerance** parameter (in pu/Pbase). The power mismatch is defined as the difference between the net power injected into the bus by generators and PQ loads and the power transmitted on all links leaving that bus.

To avoid a badly conditioned Ybus matrix, select the **Base power** parameter value in the range of nominal powers and loads connected to the network. For a transmission network with voltages ranging from 120 kV to 765 kV, a 100 MVA base is usually selected. For a distribution network with loads that have a nominal power in the range of tens to hundreds of kVA, a 100 kVA to 1 MVA power base is better adapted.

To solve the load flow, click **Compute**. The bus voltages and angles appear in the **V_LF (pu)** and **Vangle_LF (deg)** columns of the table.

	Bus ID	V_LF (pu)	Vangle_LF (...)
1	611_c	0.9737	115.7789
2	632_a	1.0210	-2.4900
3	632_b	1.0420	-121.7200
4	632_c	1.0174	117.8300
5	633_a	1.0180	-2.5546
6	633_b	1.0401	-121.7654
7	633_c	1.0148	117.8259
8	634_a	0.9939	-3.2318
9	634_b	1.0217	-122.2226
10	634_c	0.9959	117.3452
11	645_b	1.0328	-121.8997
12	645_c	1.0154	117.8572
13	646_b	1.0311	-121.9753
14	646_c	1.0134	117.9025
15	652_a	0.9825	-5.2462
16	671_a	0.9900	-5.2977
17	671_b	1.0529	-122.3402
18	671_c	0.9777	116.0260
19	675_a	0.9836	-5.5440
20	675_b	1.0553	-122.5164
21	675_c	0.9756	116.0476
22	680_a	0.9900	-5.2977
23	680_b	1.0529	-122.3402
24	680_c	0.9777	116.0260
25	684_a	0.9881	-5.3207
26	684_c	0.9757	115.9247
27	692_a	0.9900	-5.2976
28	692_b	1.0529	-122.3404
29	692_c	0.9777	116.0260

To display a load flow report that shows the power flow at each bus, click **Report**. Save this report in a file by specifying the file name at the prompt.

The report displays the summary of active and reactive powers, including the total PQ sharing between generators (SM- and Vsrc-type blocks), PQ loads (PQ-type RLC loads, dynamic loads, and asynchronous machine loads), and shunt constant Z loads (Z-type RLC loads and magnetizing branches of transformers):

SUMMARY for subnetwork No 1

Total generation	:	P=	3518.74 kW	Q=	1540.14 kvar
Total PQ load	:	P=	3101.90 kW	Q=	1880.42 kvar
Total Zshunt load	:	P=	363.47 kW	Q=	-479.42 kvar
Total losses	:	P=	53.36 kW	Q=	139.14 kvar

The **Total losses** line represents the difference between generation and loads (PQ type + Zshunt type) and represents series losses. After this summary, a voltage and power report appears for each bus. For each phase of every bus, the bus voltage and angle are listed on the first line. The next three lines give the PQ generated at the bus (all SM and voltage sources), the PQ absorbed by the PQ type loads, and the PQ absorbed by the Z-type loads. The last lines, preceded by an arrow (->), list the PQ power transmitted on all links leaving that bus.

The last column gives the positive-sequence bus voltage V1 (magnitude and angle, for three-phase buses only) and the sum of PQ powers for all phases (PQ generated by sources, PQ absorbed by loads, and PQ transmitted through transformers, lines, and series impedances). For example, you can verify that the total PQ load absorbed at bus 634 (P = 400 kW Q = 290 kvar) corresponds to the sum of active and reactive powers specified for phases A, B, and C in the load block.

Apply the Load Flow Solution to Your Model

When performing a load flow analysis, you might need to try different P, Q, and V values until you find satisfactory voltages at all buses. This can require, for example, changing generated power, load powers, or reactive shunt compensation.

To change the load flow setup, edit the parameters of the load flow blocks and of the Load Flow Bus blocks. Then click **Update** to refresh the load flow data displayed by the table. Click **Compute** to get a new load flow solution that corresponds to the changes you made.

Once you have a satisfactory load flow, update the model initial conditions according to the load flow solution. Click **Apply to Model** to initialize the PQ-type load blocks, the source block internal voltages, the machine blocks, and the initial conditions of associated regulators.

Open the Three-Phase Series RLC Load block connected at bus 632. Because the **Load type** specified in the **Load Flow** tab is constant PQ, the vector of **Nominal phase-to-neutral voltages [Va Vb Vc] (Vrms)** of this block has been changed to the corresponding bus voltages [1.021 1.042 1.0174]*2401.78 Vrms. Open the Three-Phase Source block connected at bus 632. The **Line-to-neutral voltages [Va Vb Vc] (Vrms)** parameter is also set to [1.021 1.042 1.0174]*2401.78 Vrms.

Open the Load Flow Results subsystem and start the simulation.

Observe voltage magnitudes and PQ powers on the Display blocks. These values correspond to values displayed in the load flow report.

Parameters

Model — Name of model

no default

Name of model to perform load flow analysis on.

Update — Update model

button

Click to get the latest changes in the model. Any previous load flow solution is cleared from the table.

Compute — Solve load flow

button

Click to solve the load flow. The solution is displayed in the **V_LF**, **Vangle_LF**, **P_LF**, and **Q_LF** columns of the table. The load flow is performed at the frequency, base power, PQ tolerance, and max iterations specified in the **Preferences** tab of the powergui block.

Apply — Apply solution to model

button

Click to apply the load flow solution to the model.

Add bus blocks — Add Load Flow Bus blocks to model

button

Click to add Load Flow Bus blocks to the model. The Load Flow Analyzer app determines the load flow bus required for your model and adds Load Flow Bus blocks only in places where there is no Load Flow Bus block already connected.

Report — Save load flow report

button

Click to save a load flow report that shows the power flowing toward each bus. You can save the report in either Excel or MATLAB format.

Version History

Introduced in R2021a

See Also

Functions

`power_loadflow`

Topics

“IEEE 13 Node Test Feeder”

Measurements and States Analyzer

Compute initial and steady-state values of voltages and currents of model

Description

The Measurements and States Analyzer app computes the initial states and steady-state values of the inductor currents and capacitor voltages (electrical state variables), and the steady-state voltages and currents of measurement blocks, sources, and nonlinear blocks in a model. The app uses the `power_initstates` function to compute the initial state values, and the `power_steadystate` function to compute the steady-state values.

#	Type	Block	Value	Units
1	II_phase_A	25 kV 1000 MVA	-2.499	A
2	II_phase_B	25 kV 1000 MVA	-169.2	A
3	II_phase_C	25 kV 1000 MVA	171.7	A
4	II_transfo_1_Lm	Transformer 25 kV // 2400V 6 MVA	-0.9786	A
5	II_transfo_1_winding_1	Transformer 25 kV // 2400V 6 MVA	-1.516	A
6	II_transfo_1_winding_2	Transformer 25 kV // 2400V 6 MVA	3.169	A
7	II_transfo_2_Lm	Transformer 25 kV // 2400V 6 MVA	0.4988	A
8	II_transfo_2_winding_1	Transformer 25 kV // 2400V 6 MVA	-28.32	A
9	II_transfo_2_winding_2	Transformer 25 kV // 2400V 6 MVA	168.3	A
10	II_transfo_3_Lm	Transformer 25 kV // 2400V 6 MVA	0.4799	A
11	II_transfo_3_winding_1	Transformer 25 kV // 2400V 6 MVA	29.84	A
12	II_transfo_3_winding_2	Transformer 25 kV // 2400V 6 MVA	-171.4	A

Open the Measurements and States Analyzer App

- powergui Block Parameters dialog box: On the **Tools** tab, click **Measurements and States Analyzer**.
- MATLAB command prompt: Enter `powerStates` or `powerStates('sys')`, where `sys` is the name of the model.

Parameters

Model — Name of model

text

Enter the name of the model for which you want to compute the initial states and steady-state values.

Update — Recompute measurements

button

Recompute and display the steady-state measurements.

Steady State

Format — Value formatting

112.3 (best of) (default) | 4.503e+004 (floating point) | 2590571.12

Format in which you want your measurements displayed. 4.503e+004 (floating point) displays values in mantissa-exponent form with five significant figures. 112.3 (best of) displays values with four significant figures and uses mantissa-exponent form only for numbers larger than 9999. 2590571.12 displays in plain numbers with two figures to the right of the decimal point.

Units — Define displayed units

Peak values (default) | RMS values

Set to Peak values to display the peak values of the selected values. Set to RMS values to display the root-mean-square (RMS) values of the selected values.

Frequency — Steady-state frequency

positive scalar

Choose the frequency, in hertz (Hz), in which you want to display the voltage and current phasors. The **Frequency** parameter lists all the different frequencies of the electrical sources of the model.

States — Display states

off (default) | on

Select to display the steady-state values of the inductor currents and capacitor voltages.

Measurements — Display measurements

on (default) | off

Select to display the steady-state values of the measurement blocks in the circuit.

Sources — Display sources

off (default) | on

Select to display the steady-state voltage and currents of the electrical sources of the circuit.

Nonlinear elements — Display nonlinear elements

off (default) | on

Select to display the steady-state voltage and currents of the nonlinear blocks of the circuit.

Initial States

Format — Value format

112.3 (best of) (default) | 4.503e+004 (floating point) | 2590571.12

Format in which you want your measurements displayed. 4.503e+004 (floating point) displays values in mantissa-exponent form with five significant figures. 112.3 (best of) displays values with four significant figures and uses mantissa-exponent form only for numbers larger than 9999. 2590571.12 displays in plain numbers with two figures to the right of the decimal point.

Sort values by — Sort values

State number (default) | Type

State number displays the value according to the states ordering in the state-space model. Type displays the values grouped by capacitors and inductors.

Initialize the model — Model initialization

from steady-state (default) | from zero | from user settings

Select from steady-state to force all initial electrical state values to steady-state values. When this option is selected, the **Value** column of the right pane is disabled because all states are forced to steady-state values. Select from zero to force all initial electrical state values to zero. When this option is selected, the **Value** column of the right pane is disabled because all states are forced to zero. Select from user settings to enable the **Value** column of the right pane and edit the initial state values to custom values.

Load states from file — Load from file

button

Load the initial state settings of the model from a file.

Save states to file — Save to file

button

Save the initial state settings of the model in a file.

Apply to Model — Apply steady-state values to model

button

Apply the current parameter settings to the model.

Version History

Introduced in R2021a

See Also

Functions

power_initstates | power_steadystate

Line Parameter Calculator

Compute RLC parameters of overhead transmission line from its conductor characteristics and tower geometry

Description

The Line Parameter Calculator app provides a tool to compute the RLC line parameters of the Distributed Parameters Line and PI Section Line blocks and the frequency-dependent parameters of a Distributed Parameters Line (Frequency-Dependent) block. The tool uses the `power_lineparam` function to compute the line parameters based on the geometry of the line and the type of conductors.

Line_735kV.mat

Parameters Report Help

Example of a 735 KV three-phase line

Three bundles of 4 Bersfort ACSR 1355 MCM conductors
Two 1/2 inch diameter steel grounds wires

Ytower = Ymin = average heights of conductors

General Parameters

Units: english

Ground Resistivity (ohm.m): 100

Nominal Frequency (Hz): 60

Frequency-Dependent Line Parameters

Frequency Range Logspace (Hz): [-2 5 141]

Line Length (km): 100

Compute

Frequency Dependent Model Parameters

RLC Line Parameters

Computed Parameters

1. Frequency-dependent line parameters are saved in WB structure. Select a block in your model and click the 'Send Frequency-dependent parameters to block' button

2. RLC matrices, at 60Hz:

R_matrix (ohm/km) =

0.10710	0.09729	0.09499
0.09729	0.11050	0.09729
0.09499	0.09729	0.10710

L_matrix (H/km) =

0.00158	0.00075	0.00062
0.00075	0.00157	0.00075
0.00062	0.00075	0.00158

C_matrix (F/km) =

1.16613e-08	-2.12683e-09	-5.83623e-10
-2.12683e-09	1.21174e-08	-2.12683e-09
-5.83623e-10	-2.12683e-09	1.16613e-08

Block: - No block specified

Send RLC matrices to block Send Sequences to block Send to workspace

Phase conductors (bundles): 3

	Label	Phase Number	X (ft)	Y tower (ft)	Y min (ft)	Conductor type
1	p1	1	-42	68	68	1
2	p2	2	0	68	68	1
3	p3	3	42	68	68	1

Ground conductors (bundles): 2

	Label	Phase Number	X (ft)	Y tower (ft)	Y min (ft)	Conductor type
1	g1	0	-29.5000	108	108	2
2	g2	0	29.5000	108	108	2

Conductor types: 2

☒ skin effect Internal conductor inductance evaluated from: T/D ratio

	D out (inches)	T/D ratio	unused	DC res (Ohms/mile)	mu_r	Nb_cond	Db (inches)
1	1.4000	0.3750	0.5627	0.0693	1	4	25.456
2	0.5000	0.5000	0.1947	5.0000	1	1	

Open the Line Parameter Calculator App

- powergui Block Parameters dialog box: On the **Tools** tab, click Line Parameter Calculator.
- MATLAB command prompt: Enter `powerLineParameters`

Parameters

Comments — Custom comments

no default

Use this text box to type comments that you want to save with the line parameters, for example, voltage level, conductor types and characteristics, etc.

Menu Commands

Load > Typical parameters — Load typical parameters

no default

Opens a browser window where you can select examples of line configurations provided with Simscape Electrical Specialized Power Systems software. Select the desired `.mat` file.

Selecting **Load typical parameters** allows you to load one of the following line configurations:

Line_25kV_4wires.mat	25-kV, three-phase distribution feeder with accessible neutral conductor.
Line_315kV_2circ.mat	315-kV, three-phase, double-circuit line using bundles of two conductors. Phase numbering is set to obtain the RLC parameters of the two individual circuits (six-phase line).
Line_450kV.mat	Bipolar ± 450 -kV DC line using bundles of four conductors.
Line_500kV_2circ.mat	500-kV, three-phase, double-circuit line using bundles of three conductors. Phase numbering is set to obtain the RLC parameters of the three-phase line circuit equivalent to the two circuits connected in parallel.
Line_735kV.mat	735-kV, three-phase line using bundles of four conductors.

Load > User parameters — Load user parameters

no default

Opens a browser window where you can select your own line data. Select the desired `.mat` file.

Save — Save line data

no default

Saves your line data by generating a `.mat` file that contains the GUI information and the line data.

Report — Create report

no default

Creates a file containing the line input parameters and the computed RLC parameters. The MATLAB Editor opens to display the contents of the file.

General Parameters

Units — Conductor diameter, GMR, and bundle diameter units

english (default) | metric

Select **metric** to specify conductor diameter, GMR, and bundle diameter in centimeters and conductor positions in meters. Select **english** to specify conductor diameter, GMR, and bundle diameter in inches and conductor positions in feet.

Ground Resistivity — Ground Resistivity

100 (default) | positive scalar

Specify the ground resistivity, in ohm-meters. A zero value (perfectly conducting ground) is allowed.

Nominal Frequency — Frequency to evaluate RLC parameters

60 (default) | positive scalar

Specify the frequency, in hertz, to evaluate RLC parameters.

Line Geometry**Phase conductors (bundles)** — Number of phase conductors (bundles)

3 (default) | positive scalar

Specify the number of phase conductors (single conductors or bundles of subconductors).

Ground conductors (bundles) — Number of ground wires (bundles)

2 (default) | nonnegative scalar

Specify the number of ground wires (single conductors or bundles of subconductors). Ground wires are not usually bundled.

Label — Conductor or bundle identifiers

no default

Lists the conductor or bundle identifiers. Phase conductors are identified as p1, p2,..., pn. Ground wires are identified as g1,g2,..., gn.

Phase, Phase Number — Phase number

no default

Specify the phase number to which the conductor belongs. Several conductors may have the same phase number. All conductors that have the same phase number are lumped together and are considered as a single equivalent conductor in the R, L, and C matrices. For example, if you want to compute the line parameters of a three-phase line equivalent to a double-circuit line such as the one represented in the figure “Configuration of a Three-Phase Double-Circuit Line” on page 2-41, you specify phase numbers 1, 2, 3 for conductors p1, p2, p3 (circuit 1) and phase numbers 3, 2, 1 for conductors p4, p5, p6 (circuit 2), respectively. If you prefer to simulate this line as two individual circuits and have access to the six phase conductors, you specify phase numbers 1, 2, 3, 6, 5, 4 respectively for conductors p1, p2, p3, p4, p5 and p6.

In three-phase systems, the three phases are usually labeled A, B, and C. The correspondence with the phase number is:

1, 2, 3, 4, 5, 6, 7, 8, 9,... = A, B, C, A, B, C, A, B, C,...

You can also use the phase number to lump conductors of an asymmetrical bundle.

For ground wires, the phase number is forced to zero. All ground wires are lumped with the ground and they do not contribute to the R, L, and C matrix dimensions. If you need to access the ground wire connections in your model, you must specify these ground wires as normal phase conductors and manually connect them to the ground.

X — Horizontal position of conductor

positive scalar

Specify the horizontal position of the conductor, in meters or feet. The location of the zero reference position is arbitrary. For a symmetrical line, you typically choose $X = 0$ at the center of the line.

Y tower — Vertical position of conductor at tower

positive scalar

Specify the vertical position of the conductor (at the tower) with respect to ground, in meters or feet.

Y min — Vertical position of the conductor at mid-span

positive scalar

Specify the vertical position of the conductor with respect to ground at mid-span, in meters or feet.

The average height of the conductor (see the figure “Configuration of a Three-Phase Double-Circuit Line” on page 2-41) is produced by this equation:

$$Y_{average} = Y_{min} + \frac{sag}{3} = \frac{2Y_{min} + Y_{tower}}{3}$$

Y_{tower} = height of conductor at tower

Y_{min} = height of conductor at mid span

sag = $Y_{tower} - Y_{min}$

Instead of specifying two different values for Y_{tower} and Y_{min} , you may specify the same $Y_{average}$ value.

Conductor type — Conductor or bundle type numbers

positive integer

Specify one of the conductor or bundle type numbers listed in the first column of the table of conductor characteristics.

Conductors

Conductor types — Number of conductor types

positive integer

Specify the number of conductor types (single conductor or bundle of subconductors). This parameter determines the number of rows in the table of conductors. The phase conductors and ground conductors can be either single conductors or bundles of subconductors. For voltage levels of 230 kV and higher, phase conductors are usually bundled to reduce losses and electromagnetic interferences due to corona effect. Ground wires are usually not bundled.

For a simple AC three-phase line, single- or double-circuit, there are usually two types of conductors: one type for the phase conductors and one type for the ground wires. You need more than two types for several lines in the same corridor, DC bipolar lines or distribution feeders, where neutral and sheaths of TV and telephone cables are represented.

Internal conductor inductance evaluated from — Computation method for conductor internal inductance

T/D ratio (default) | Geometric Mean Radius (GMR) | Reactance Xa at 1-foot spacing |
Reactance Xa at 1-meter spacing

Select one of the following three parameters to specify how the conductor internal inductance is computed: T/D ratio, Geometric Mean Radius (GMR), or Reactance Xa at 1-foot spacing (or Reactance Xa at 1-meter spacing if the **Units** parameter is set to metric).

If you select **T/D ratio**, the internal inductance is computed from the T/D value specified in the table of conductors, assuming a hollow or solid conductor. D is the conductor diameter and T is the thickness of the conducting material (see the figure “Configuration of a Three-Phase Double-Circuit Line” on page 2-41). The conductor self-inductance and resistance are computed from the conductor diameter, T/D ratio, DC resistance, and relative permeability of conducting material and specified frequency.

If you select **Geometric Mean Radius (GMR)**, the conductor GMR evaluates the internal inductance. When the conductor inductance is evaluated from the GMR, the specified frequency does not affect the conductor inductance. You must provide the manufacturer's GMR for the desired frequency (usually 50 Hz or 60 Hz). When you are using the **T/D ratio** option, the corresponding conductor GMR at the specified frequency is displayed in the **Conductors** table.

Selecting **Reactance Xa at 1-foot spacing** (or **Reactance Xa at 1-meter spacing**) uses the positive-sequence reactance at the specified frequency of a three-phase line having 1-foot (or 1-meter) spacing between the three phases to compute the conductor internal inductance.

Include conductor skin effect — Include impact of frequency on conductor AC resistance and inductance

on (default) | off

Select this check box to include the impact of frequency on conductor AC resistance and inductance (skin effect). If this parameter is cleared, the resistance is kept constant at the value specified by the **Conductor DC resistance** parameter and the inductance is kept constant at the value computed in DC, using the **D out** (conductor outside diameter) and the **T/D ratio** parameters of the **Conductors** table. When skin effect is included, the conductor AC resistance and inductance are evaluated considering a hollow conductor with T/D ratio (or solid conductor if $T/D = 0.5$). The T/D ratio evaluates the AC resistance even if the conductor inductance is evaluated from the GMR or from the reactance at 1-foot spacing or 1-meter spacing. The ground skin effect is always considered and it depends on the ground resistivity.

D out — Conductor outside diameter

positive scalar

Specify the conductor outside diameter, in centimeters or inches.

T/D ratio — Conductor T/D ratio

scalar between 0 and 0.5

Specify the T/D ratio of the hollow conductor. T is the thickness of the conducting material, and D is the outside diameter. This parameter can vary between 0 and 0.5. A T/D value of 0.5 indicates a solid conductor. For Aluminum Cable Steel Reinforced (ACSR) conductors, you can ignore the steel core and consider a hollow aluminum conductor (typical T/D ratios are between 0.3 and 0.4). The T/D ratio is used to compute the conductor AC resistance when the **Include conductor skin effect** parameter is selected. It is also used to compute the conductor self-inductance when the parameter **Internal conductor inductance evaluated from** is set to **T/D ratio**.

GMR — Geometric mean radius

positive scalar

This parameter is accessible only when the parameter **Internal conductor inductance evaluated from** is set to **Geometric Mean Radius (GMR)**. Specify the GMR in centimeters or inches. The GMR at 60 Hz or 50 Hz is usually provided by conductor manufacturers. When the parameter **Internal conductor inductance evaluated from** is set to **T/D ratio**, the value of the

corresponding GMR giving the same conductor inductance is displayed. When the parameter **Internal conductor inductance evaluated from** is set to Reactance X_a at 1-foot spacing or Reactance X_a at 1-meter spacing, the title of the column changes to **X_a** .

X_a — Reactance X_a at 1-meter spacing or 1-foot spacing
positive scalar

This parameter is accessible only when **Internal conductor inductance evaluated from** is set to Reactance X_a at 1-meter spacing or Reactance X_a at 1-foot spacing. Specify the X_a value in ohms/km or ohms/mile at the specified frequency. The X_a value at 60 Hz or 50 Hz is usually provided by conductor manufacturers.

DC res — Conductor DC resistance
positive scalar

Specify the DC resistance of conductor in ohms/km or ohms/mile.

μ_r — Conductor relative permeability
positive scalar

Specify the relative permeability μ_r of the conducting material. $\mu_r = 1.0$ for nonmagnetic conductors (such as aluminum or copper). This parameter is not accessible when the **Include conductor skin effect** parameter is cleared.

Nb_cond — Number of conductors per bundle
positive integer

Specify the number of subconductors in the bundle or 1 for single conductors.

Db — Bundle diameter
positive scalar

Specify the bundle diameter, in centimeters or inches. This parameter is not accessible when the **Nb_cond** is set to 1. When you specify bundled conductors, the subconductors are assumed to be evenly spaced on a circle. If this is not the case, you must enter individual subconductor positions in the **Line Geometry** table and lump these subconductors by giving them the same **Phase Number** parameter.

Angle — Angle of conductor 1
positive scalar

Specify an angle, in degrees, that determines the position of the first conductor in the bundle with respect to a horizontal line parallel to ground. This angle determines the bundle orientation. This parameter is not accessible when the **Nb_cond** is set to 1.

Frequency-Dependent Line Parameters

Frequency range logspace — Frequency range for parameter computation
[-2,5,141] (default) | three-element vector

Specify a frequency range for the parameter computation. Enter a vector of three elements, [X_1 , X_2 , N]. This parameter defines a frequency vector of N logarithmically equally spaced points between decades 10^{X_1} and 10^{X_2} .

Line Length — Length of line
100 (default) | positive scalar

Specify the length of the line, in km.

Compute

RLC Line Parameters — Compute RLC line parameters

no default

Computes the RLC parameters. After completion of the parameters computation, results are displayed in the **Computed Parameters** section.

Note The R, L, and C parameters are always displayed respectively in ohms/km, henries/km, and farads/km, even if the English units specify the input parameters.

If the number of phase conductors is 3 or 6, the symmetrical component parameters are also displayed:

- For a three-phase line (one circuit), R10, L10, and C10 vectors of two values are displayed for positive-sequence and zero-sequence RLC values.
- For a six-phase line (two coupled three-phase circuits), R10, L10, and C10 are vectors of five values containing the following RLC sequence parameters: the positive-sequence and zero-sequence of circuit 1, the mutual zero-sequence between circuit 1 and circuit 2, and the positive-sequence and zero-sequence of circuit 2.

Frequency Dependent Model Parameters — Compute frequency dependent parameters

no default

Computes the frequency dependent parameters. After completion of the parameters computation, results are displayed in the **Computed Parameters** section.

Computed Parameters

Block — Selected block

no default

Select a Distributed Parameters Line block (either to set the matrices or sequence RLC parameters), a Pi Section Line block, or a Three-Phase PI Section Line block in your model, then click the button to confirm the block selection. The name of the selected block appears in the left window.

Send RLC matrices to block — Download RLC matrices to block

no default

Downloads RLC matrices into the selected block. This button is not visible when the selected block is a Distributed Parameters Line (Frequency-Dependent) block.

Send Sequences to block — Download RLC sequence parameters to block

no default

Downloads RLC sequence parameters into the selected block. This button is not visible when the selected block is a Distributed Parameters Line (Frequency-Dependent) block.

Send to workspace — Send matrices and component parameters to MATLAB workspace

no default

Sends the R, L, and C matrices, as well as the symmetrical component parameters, to the MATLAB workspace. The following variables are created in your workspace: `R_matrix`, `L_matrix`, `C_matrix`, and `R10`, `L10`, `C10` for symmetrical components.

Send Frequency-Dependent Parameters to block — Download frequency-dependent parameters to block
`button`

Downloads the frequency-dependent parameters into the selected block. This button is not visible when the block is not a Distributed Parameters Line (Frequency-Dependent) block.

Version History

Introduced in R2020b

See Also

Blocks

Distributed Parameters Line | Distributed Parameters Line (Frequency-Dependent) | PI Section Line | Three-Phase PI Section Line

Functions

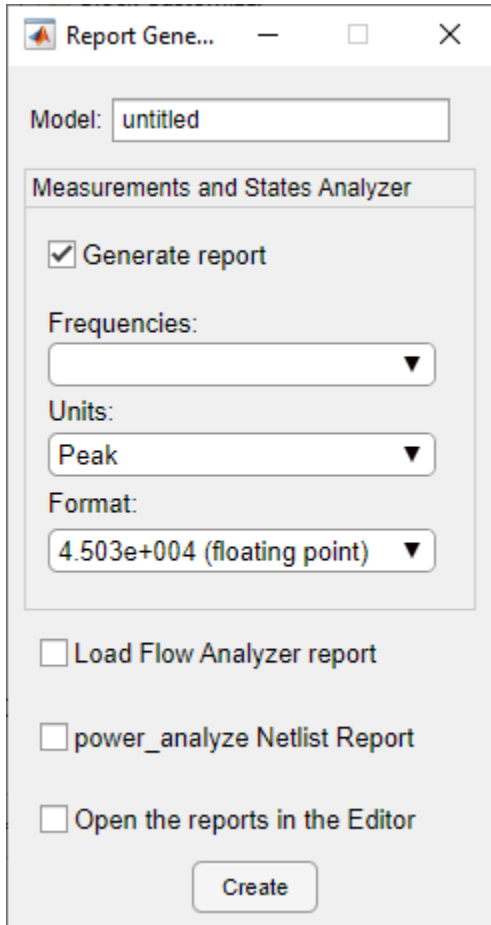
`power_lineparam`

Report Generator

Generate report of steady state variables, initial states, and machine load flow for model

Description

The Report Generator app generates reports of steady state variables, initial states, netlist representation, and machine load flow for a model.



Open the Report Generator App

- MATLAB command prompt: Enter `power_report`
- Powergui block: Click **Report Generator**

Parameters

Model — Name of model
no default

Name of the model for which you want to create the reports.

Generate report — Measurements and States Analyzer report

on (default) | off

Select to generate a Measurements and States Analyzer report when you click **Create**.

Frequencies — Model frequencies

no default

Frequency to use in the Measurements and States Analyzer report.

Units — Define displayed units

Peak values (default) | RMS values

Set this parameter to **Peak values** to display the peak values of the selected values. Set this parameter to **RMS values** to display the root mean square (RMS) values of the selected values.

Format — Value formatting

12.3 (best of) (default) | 4.503e+004 (floating point) | 2590571.12

Format in which you want your measurements displayed in the Measurements and States Analyzer report. 4.503e+004 (floating point) displays values in mantissa-exponent form with five significant figures. 12.3 (best of) displays values with four significant figures and uses mantissa-exponent form only for numbers larger than 9999. 2590571.12 displays values in plain numbers with two figures to the right of the decimal point.

Load Flow Analyzer report — Generate Load Flow Analyzer report

off (default) | on

Select to generate the Load Flow Analyzer App report for the model.

power_analyze Netlist report — Generate power_analyze function report

off (default) | on

Select to generate a netlist of the model. The report contains the node numbers automatically generated for the model, as well as the parameter values of all linear elements. See the formats described in the power_statespace reference page.

Open the reports in the Editor — Open selected reports

off (default) | on

Select to open the reports in the editor after you create the report.

Buttons

Create — Create specified reports

button

Create specified reports.

Version History

Introduced in R2021b

Apps
Functions

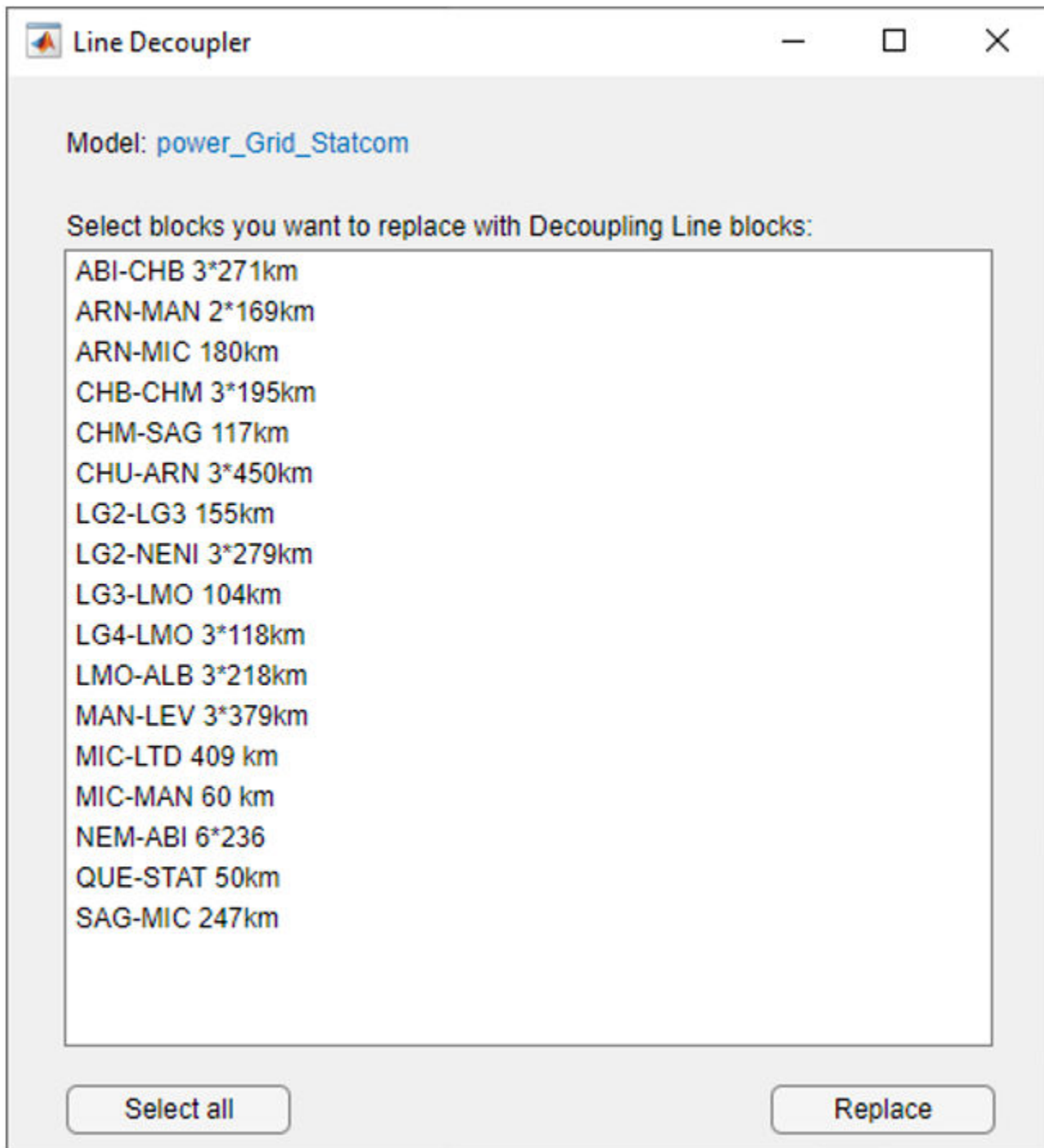
Line Decoupler

Replace selected Distributed Parameters Line blocks with Decoupling Line blocks

Description

The Line Decoupler app allows you to replace the single-phase and three-phase Distributed Parameters Line blocks in your model with Decoupling Line blocks. You can select one or several blocks in the list and apply the changes in one step. The app automatically sets all the parameters of the Decoupling Line blocks to match those of the replaced blocks. When done, you can run the simulation with no additional steps.

For more information on why, how, and when you should use Decoupling Line blocks, see [Decoupling Line](#).



Open the Line Decoupler App

- powergui Block Parameters dialog box: On the **Tools** tab, click **Line Decoupler**.

- MATLAB command prompt: `Enter DecouplingLineReplace(Model)`
`Model` is the name of a model that is loaded in memory.

Examples

- “MMC-STATCOM Model running in Real-Time (Discrete)”

Parameters

Select all — Select all blocks
button

Select all the Distributed Parameters Line blocks listed in the app.

Replace — Replace selected blocks
button

Replace the selected Distributed Parameters Line blocks with Decoupling Line blocks. The app sets the parameters of the Decoupling Line blocks with the parameters of the Distributed Parameters Line blocks. After the conversion, the list of blocks is automatically updated in the app. The update requires no additional modifications and provides the same simulation results as the original model.

Version History

Introduced in R2020b

See Also

[Decoupling Line](#) | [Decoupling Line \(Three-Phase\)](#) | [Distributed Parameters Line](#) | [powergui](#)

Topics

“MMC-STATCOM Model running in Real-Time (Discrete)”

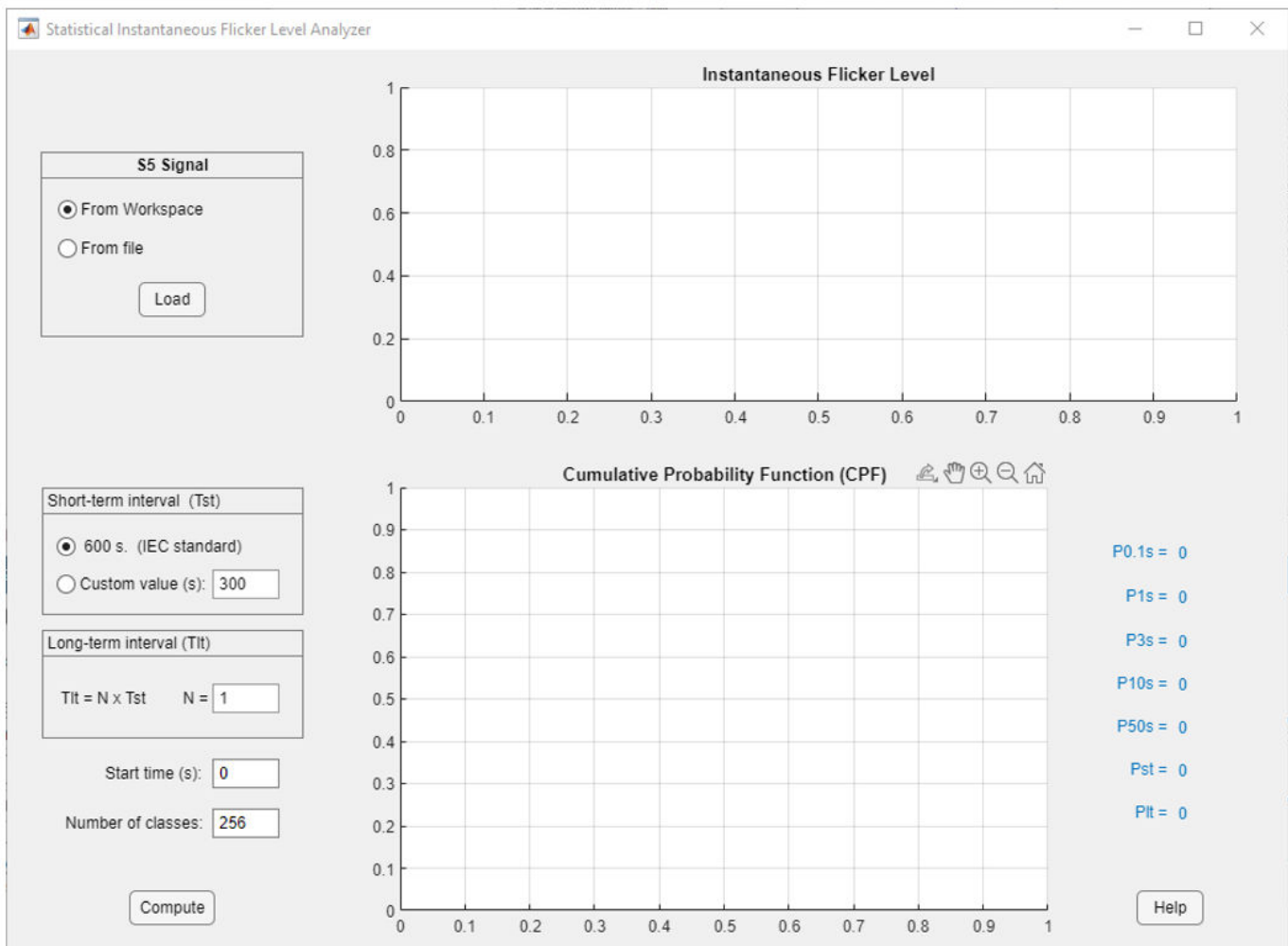
Statistical Flicker Level Analyzer

Perform statistical analysis of instantaneous flicker level signal

Description

The **Statistical Flicker Level Analyzer** app computes the short-term flicker severity (Pst) and long-term flicker severity (Plt) and displays the cumulative probability function (CPF) of an instantaneous flicker level signal (S5 signal).

You must have a Control System Toolbox license to use the **Statistical Flicker Level Analyzer** app.



Open the Statistical Flicker Level Analyzer App

- MATLAB command prompt: Enter `power_flicker`
- Digital Flickermeter block: Click **Open statistical analysis**

Parameters

Short-term interval (Tst) — Short-term interval value

IEC Standard = 600 s (default) | Custom value (s)

When you select **IEC standard = 600 s**, the short-term interval observation period is 10 minutes (600 seconds). This setting corresponds to the time specified in the IEC 61000-4-15 standard.

Select **Custom value (s)** to specify the short-term interval observation period, in seconds. The default value is 300.

N — Number of short-term intervals

1 (default) | positive integer

Number of short-term intervals used to measure the long-term flicker severity (Plt).

Start time (s) — Analysis start time

0 (default) | positive scalar

Time of the beginning of the observed period, in seconds.

Number of classes — Number of classes

256 (default) | positive integer

Number of classes that are used by the tool to compute the cumulative probability function of the instantaneous flicker level.

Buttons

Load — Load S5 signal

button

Load the S5 signal into the app. Choose between **From Workspace** and **From file**.

Help — Open documentation

button

Open the documentation page.

Compute — Compute cumulative probability function and flicker levels

button

Compute the percentiles P0.1s, P1s, P3s, P10s, and P50s, which are the flicker levels exceeded for 0.1%, 1%, 3%, 10%, and 50% of the time during the observed period. Computes the short-term flicker severity (Pst) and long-term flicker severity (Plt) and displays the cumulative probability function (CPF) of the S5 signal.

Version History

Introduced in R2021b

See Also

Digital Flickermeter

Hysteresis Designer

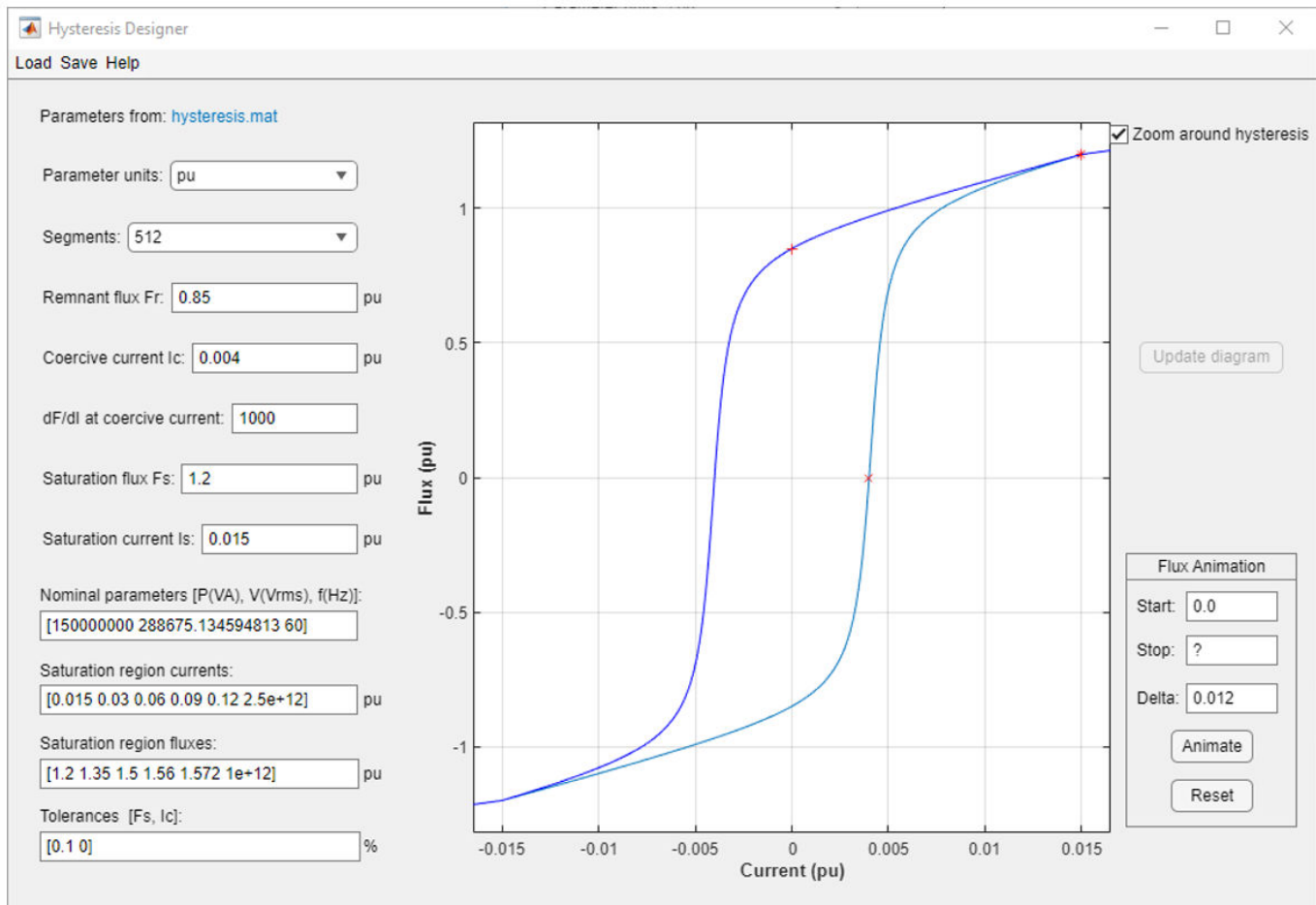
View and edit hysteresis characteristic for saturable core of Saturable Transformer blocks

Description

The Hysteresis Designer app allows you to view and edit a hysteresis characteristic for the saturable core of the Multi-Winding Transformer, Saturable Transformer, Three-Phase Transformer (Two Windings), and Three-Phase Transformer (Three Windings) blocks. The hysteresis characteristic includes the saturation region located at the limits of the hysteresis loop.

A default hysteresis characteristic displays when you open the app, but you can build as many hysteresis characteristics as you want and save them in different MAT-files. You can use the same characteristic for all your transformer blocks, or you can use different characteristics for each transformer block in the circuit. In the mask of a transformer block, select **Simulate hysteresis** and specify a MAT-file to use.

The Hysteresis Designer app includes a flux animation tool that can visualize how by Simscape Electrical Specialized Power Systems software performs the simulation of the hysteresis. This is an optional tool that is not necessary for the model parameterization. The app calculates the initial trajectory according to the defined hysteresis characteristic. The tool assumes that the last reversal point before starting flux is located on the major loop. The operating point travels until the defined stop flux.



Open the Hysteresis Designer App

- MATLAB command prompt: Enter `power_hysteresis`
- In the Powergui block, click **Hysteresis Designer**

Parameters

Hysteresis Parameters

Parameter units — App parameter units
pu (default) | SI

Convert the fluxes and currents that define the hysteresis characteristic from SI to pu or from pu to SI.

Segments — Number of linear segments
512 (default) | 32 | 64 | 128 | 256

Number of linear segments used to define the right side of the hysteresis loop. The left side of the loop is the symmetric image of the right side.

Remnant flux FR — Remnant flux

0.85 (default) | positive scalar

Remnant flux point of the hysteresis characteristic (flux at zero current). The remnant flux point is identified by a + mark in the plot.

Coercive current IC — Coercive current

0.004 (default) | positive scalar

Coercive current point of the hysteresis characteristic.

dF/dI at coercive current — Flux slope at coercive current

1000 (default) | positive scalar

Slope of the flux at the coercive current point (current at zero flux).

Saturation flux Fs — Saturation flux

1.2 (default) | positive scalar

Saturation flux point where the hysteresis loop becomes a single-valued saturation curve. The saturation flux point is identified by a * mark in the plot.

Saturation current Is — Saturation current

0.015 (default) | positive scalar

Saturation current point where the hysteresis loop becomes a single-valued saturation curve. The saturation region is defined by the **Saturation region currents** parameter.

Nominal parameters [P(VA), V(Vrms), f(Hz)] — Transformer nominal parameters

[150e6, 500e3/sqrt(3), 60] (default) | vector of 3

Nominal parameters (nominal power in VA, nominal voltage of winding 1 in volts RMS, and nominal frequency in Hz) used in the conversion of the hysteresis parameters.

Saturation region currents — Saturation current

[0.015 0.03 0.06 0.09 0.12 2.5e+12] (default) | vector

Vector of current values that define the saturation characteristic. The number of specified points must be the same as the **Saturation region fluxes** parameter. You need to specify only the positive part of the characteristic.

Saturation region fluxes — Saturation current

[1.2 1.35 1.5 1.56 1.572 1e+12] (default) | positive scalar

Vector of flux values that define the saturation characteristic. The number of specified points must be the same as for the **Saturation region currents** parameter. You need to specify only the positive part of the characteristic.

Tolerances [Tol_F, Tol_I] — Saturation current

0.015 (default) | positive scalar

This parameter is mainly used to minimize the generation of superfluous, very small internal loops or new trajectories because they have little effect and they consume computer memory space. The model can memorize up to 50 embedded minor loops at any time.

The **TOL_F** parameter is the tolerance value used to detect whether, after a flux reversal, the operating point remains on the same minor loop or a new embedded loop is created. The smaller the value, the lesser the effect on the normal trajectory behavior. The bigger the value, the lesser the generated number of embedded minor loops.

Finally, when the distance between the I coordinate of the actual point of reversal and the penultimate one is less than **TOL_I**, then evolution within these two points follows a line segment instead of a loop.

Zoom around hysteresis — Saturation current

on (default) | off

Select to zoom the plot around the hysteresis curve.

Flux Animation

Start — Starting flux

0.0 (default) | scalar

Starting flux.

Stop — Ending flux

? (default) | scalar

Flux at which the flux animation stops.

Delta — Flux delta

1,78e-08 (default) | positive scalar

Flux increment, in pu or SI, that is used to go from start flux to stop flux.

Buttons

Update diagram — Update hysteresis diagram

Update the hysteresis diagram after you changed hysteresis parameters.

Animate — Animate flux animation

Start the flux animation tool.

Reset — Reset flux animation

Reset the flux animation tool.

Menu

Load — Load hysteresis parameters

Opens a browser window where you can select your own cable data. Select the desired MAT-file.

Save — Save hysteresis parameters

Saves your cable data by generating a MAT-file that contains the GUI information and the cable data.

Help — App documentation

Open the documentation page of the app.

Version History

Introduced in R2021b

See Also

Blocks

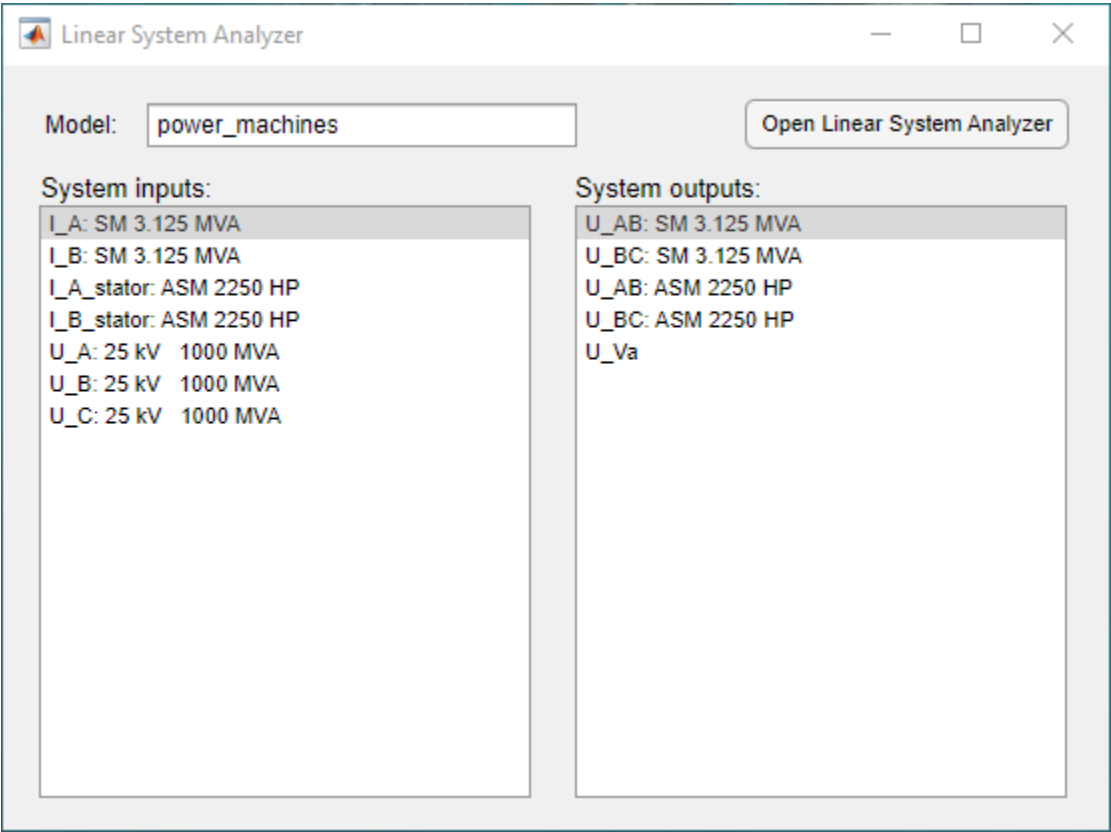
Saturable Transformer

Linear System Analyzer

Create state-space object of model for use with the Linear System Analyzer tool

Description

The Linear System Analyzer app creates a state-space (SS) object of the linear part of the electrical circuit (ABCD state-space matrices representing the RLC devices) for the specified model. The app models the nonlinear elements, such as the Distributed Parameters Line blocks, as inputs and outputs to the SS object by modeling them as the current source in the feedback to the state-space system. You can analyze the time and frequency responses of this SS object with the Linear System Analyzer provided with Control System Toolbox software.



Open the Linear System Analyzer App

- MATLAB command prompt: Enter powerCableParameters
- Powergui block: Click **Use Linear System Analyzer**

Parameters

model — Analyzed model
no default | string

Model to analyze.

System inputs — Linear system inputs of model
no default | string

Inputs of the state-space model of your circuit for the Linear System Analyzer to use.

System outputs — Linear system outputs of model
no default | string

Outputs of the state-space model of your circuit for the Linear System Analyzer to use.

Buttons

Open Linear System Analyzer — Open Linear System Analyzer
button

Generates the state-space model of the circuit and opens the Linear System Analyzer for the selected system inputs and outputs.

Version History

Introduced in R2021b

Apps

Functions

Technical Conventions

Per-Unit System of Units

In this section...
“International System of Units” on page A-2
“What Is the Per-Unit System?” on page A-2
“Example 1: Three-Phase Transformer” on page A-4
“Example 2: Asynchronous Machine” on page A-5
“Base Values for Instantaneous Voltage and Current Waveforms” on page A-6
“Why Use the Per-Unit System Instead of the Standard SI Units?” on page A-6

International System of Units

Simscape Electrical Specialized Power Systems software uses the International System of Units (SI), as described in the following table.

Quantity	Unit	Symbol
Time	second	s
Length	meter	m
Mass	kilogram	kg
Energy	joule	J
Current	ampere	A
Voltage	volt	V
Active power	watt	W
Apparent power	volt-ampere	VA
Reactive power	var	var
Impedance	ohm	Ω
Resistance	ohm	Ω
Inductance	henry	H
Capacitance	farad	F
Flux linkage	volt-second	V. s
Rotation speed	radians per second revolutions per minute	rad/s rpm
Torque	newton-meter	N.m
Inertia	kilogram-meter ²	kg.m ²
Friction factor	newton-meter-second	N.m.s

What Is the Per-Unit System?

The per-unit system is widely used in the power system industry to express values of voltages, currents, powers, and impedances of various power equipment. It is mainly used for transformers and AC machines.

For a given quantity (voltage, current, power, impedance, torque, etc.) the per-unit value is the value related to a base quantity.

$$\text{base value in p.u.} = \frac{\text{quantity expressed in SI units}}{\text{base value}}$$

Generally the following two base values are chosen:

- The base power = nominal power of the equipment
- The base voltage = nominal voltage of the equipment

All other base quantities are derived from these two base quantities. Once the base power and the base voltage are chosen, the base current and the base impedance are determined by the natural laws of electrical circuits.

$$\text{base current} = \frac{\text{base power}}{\text{base voltage}}$$

$$\text{base impedance} = \frac{\text{base voltage}}{\text{base current}} = \frac{(\text{base voltage})^2}{\text{base power}}$$

For a transformer with multiple windings, each having a different nominal voltage, the same base power is used for all windings (nominal power of the transformer). However, according to the above definitions, there are as many base values as windings for voltages, currents, and impedances.

The saturation characteristic of saturable transformer is given in the form of an instantaneous current versus instantaneous flux-linkage curve: [$i_1 \phi_1$; $i_2 \phi_2$; ... , in ϕ_{in}].

When the Per-Unit system is used to specify the transformer R L parameters, the flux linkage and current in the saturation characteristic must be also specified in pu. The corresponding base values are

$$\text{base instantaneous current} = (\text{base rms current}) \times \sqrt{2}$$

$$\text{base flux linkage} = \frac{(\text{base rms voltage}) \times \sqrt{2}}{2\pi \times (\text{base frequency})}$$

where current, voltage, and flux linkage are expressed respectively in volts, amperes, and volt-seconds.

For AC machines, the torque and speed can be also expressed in pu. The following base quantities are chosen:

- The base speed = synchronous speed
- The base torque = torque corresponding at base power and synchronous speed

$$\text{base torque} = \frac{\text{base power (3 phases) in VA}}{\text{base speed in radians/second}}$$

Instead of specifying the rotor inertia in $\text{kg}\cdot\text{m}^2$, you would generally give the inertia constant H defined as

$$H = \frac{\text{kinetic energy stored in the rotor at synchronous speed in joules}}{\text{machine nominal power in VA}}$$

$$H = \frac{\frac{1}{2} \times J \cdot \omega^2}{P_{nom}}$$

The inertia constant is expressed in seconds. For large machines, this constant is around 3 to 5 seconds. An inertia constant of 3 seconds means that the energy stored in the rotating part could supply the nominal load during 3 seconds. For small machines, H is lower. For example, for a 3 HP motor, it can be between 0.5 and 0.7 second.

Example 1: Three-Phase Transformer

Consider, for example, a three-phase two-winding transformer. The following typical parameters could be provided by the manufacturer:

- Nominal power = 300 kVA total for three phases
- Nominal frequency = 60 Hz
- Winding 1: connected in wye, nominal voltage = 25 kV RMS line-to-line
resistance 0.01 pu, leakage reactance = 0.02 pu
- Winding 2: connected in delta, nominal voltage = 600 V RMS line-to-line
resistance 0.01 pu, leakage reactance = 0.02 pu
- Magnetizing losses at nominal voltage in % of nominal current:
Resistive 1%, Inductive 1%

The base values for each single-phase transformer are first calculated:

- For winding 1:

Base power	$300 \text{ kVA}/3 = 100\text{e}3 \text{ VA/phase}$
Base voltage	$25 \text{ kV}/\sqrt{3} = 14434 \text{ V RMS}$
Base current	$100\text{e}3/14434 = 6.928 \text{ A RMS}$
Base impedance	$14434/6.928 = 2083 \Omega$
Base resistance	$14434/6.928 = 2083 \Omega$
Base inductance	$2083/(2\pi*60) = 5.525 \text{ H}$

- For winding 2:

Base power	$300 \text{ kVA}/3 = 100\text{e}3 \text{ VA}$
Base voltage	600 V RMS
Base current	$100\text{e}3/600 = 166.7 \text{ A RMS}$
Base impedance	$600/166.7 = 3.60 \Omega$
Base resistance	$600/166.7 = 3.60 \Omega$
Base inductance	$3.60/(2\pi*60) = 0.009549 \text{ H}$

The values of the winding resistances and leakage inductances expressed in SI units are therefore

- For winding 1: $R_1 = 0.01 * 2083 = 20.83 \Omega$; $L_1 = 0.02 * 5.525 = 0.1105 \text{ H}$
- For winding 2: $R_2 = 0.01 * 3.60 = 0.0360 \Omega$; $L_2 = 0.02 * 0.009549 = 0.191 \text{ mH}$

For the magnetizing branch, magnetizing losses of 1% resistive and 1% inductive mean a magnetizing resistance R_m of 100 pu and a magnetizing inductance L_m of 100 pu. Therefore, the values expressed in SI units referred to winding 1 are

- $R_m = 100 \times 2083 = 208.3 \text{ k}\Omega$
- $L_m = 100 \times 5.525 = 552.5 \text{ H}$

Example 2: Asynchronous Machine

Now consider the three-phase four-pole Asynchronous Machine block in SI units provided in the Machines library of Fundamental Blocks library. It is rated 3 HP, 220 V RMS line-to-line, 60 Hz.

The stator and rotor resistance and inductance referred to stator are

- $R_s = 0.435 \text{ }\Omega$; $L_s = 2 \text{ mH}$
- $R_r = 0.816 \text{ }\Omega$; $L_r = 2 \text{ mH}$

The mutual inductance is $L_m = 69.31 \text{ mH}$. The rotor inertia is $J = 0.089 \text{ kg.m}^2$.

The base quantities for one phase are calculated as follows:

Base power	$3 \text{ HP} \times 746 \text{ VA} / 3 = 746 \text{ VA/phase}$
Base voltage	$220 \text{ V} / \sqrt{3} = 127.0 \text{ V RMS}$
Base current	$746 / 127.0 = 5.874 \text{ A RMS}$
Base impedance	$127.0 / 5.874 = 21.62 \text{ }\Omega$
Base resistance	$127.0 / 5.874 = 21.62 \text{ }\Omega$
Base inductance	$21.62 / (2\pi \times 60) = 0.05735 \text{ H} = 57.35 \text{ mH}$
Base speed	$1800 \text{ rpm} = 1800 \times (2\pi) / 60 = 188.5 \text{ radians/second}$
Base torque (3-phase)	$746 \times 3 / 188.5 = 11.87 \text{ newton-meters}$

Using the above base values, you can compute the values in per units.

$$R_s = 0.435 / 21.62 = 0.0201 \text{ pu} \quad L_s = 2 / 57.35 = 0.0349 \text{ pu}$$

$$R_r = 0.816 / 21.62 = 0.0377 \text{ pu} \quad L_r = 2 / 57.35 = 0.0349 \text{ pu}$$

$$L_m = 69.31 / 57.35 = 1.208 \text{ pu}$$

The inertia is calculated from inertia J , synchronous speed, and nominal power.

$$H = \frac{\frac{1}{2} \times J \cdot \omega^2}{P_{nom}} = \frac{\frac{1}{2} \times 0.089 \times (188.5)^2}{3 \times 746} = 0.7065 \text{ seconds}$$

If you open the dialog box of the Asynchronous Machine block in pu units provided in the Machines library of Fundamental Blocks (powerlib) library, you find that the parameters in pu are the ones calculated above.

Base Values for Instantaneous Voltage and Current Waveforms

When displaying instantaneous voltage and current waveforms on graphs or oscilloscopes, you normally consider the peak value of the nominal sinusoidal voltage as 1 pu. In other words, the base values used for voltage and currents are the RMS values given above multiplied by $\sqrt{2}$.

Why Use the Per-Unit System Instead of the Standard SI Units?

Here are the main reasons for using the per-unit system:

- When values are expressed in pu, the comparison of electrical quantities with their "normal" values is straightforward.

For example, a transient voltage reaching a maximum of 1.42 pu indicates immediately that this voltage exceeds the nominal value by 42%.

- The values of impedances expressed in pu stay fairly constant whatever the power and voltage ratings.

For example, for all transformers in the 3 kVA to 300 kVA power range, the leakage reactance varies approximately between 0.01 pu and 0.03 pu, whereas the winding resistances vary between 0.01 pu and 0.005 pu, whatever the nominal voltage. For transformers in the 300 kVA to 300 MVA range, the leakage reactance varies approximately between 0.03 pu and 0.12 pu, whereas the winding resistances vary between 0.005 pu and 0.002 pu.

Similarly, for salient pole synchronous machines, the synchronous reactance X_d is generally between 0.60 and 1.50 pu, whereas the subtransient reactance X'_d is generally between 0.20 and 0.50 pu.

It means that if you do not know the parameters for a 10 kVA transformer, you are not making a major error by assuming an average value of 0.02 pu for leakage reactances and 0.0075 pu for winding resistances.

The calculations using the per-unit system are simplified. When all impedances in a multivoltage power system are expressed on a common power base and on the nominal voltages of the different subnetworks, the total impedance in pu seen at one bus is obtained by simply adding all impedances in pu, without taking into consideration the transformer ratios.

Park Transformation

The Park transformation used in Simscape Electrical Specialized Power Systems models and functions corresponds to the definition provided in [1 on page A-7].

It transforms three quantities (direct axis, quadratic axis, and zero-sequence components) expressed in a two-axis reference frame back to phase quantities.

The following transformation is used:

$$\begin{aligned} V_d &= V_d \sin(\omega t) + V_q \cos(\omega t) + V_0 \\ V_b &= V_d \sin(\omega t - 2\pi/3) + V_q \cos(\omega t - 2\pi/3) + V_0 \\ V_c &= V_d \sin(\omega t + 2\pi/3) + V_q \cos(\omega t + 2\pi/3) + V_0, \end{aligned}$$

where ω = rotation speed (rad/s) of the rotating frame.

The following reverse transformation is used:

$$\begin{aligned} V_d &= \frac{2}{3}(V_a \sin(\omega t) + V_b \sin(\omega t - 2\pi/3) + V_c \sin(\omega t + 2\pi/3)) \\ V_q &= \frac{2}{3}(V_a \cos(\omega t) + V_b \cos(\omega t - 2\pi/3) + V_c \cos(\omega t + 2\pi/3)) \\ V_0 &= \frac{1}{3}(V_a + V_b + V_c), \end{aligned}$$

where ω = rotation speed (rad/s) of the rotating frame.

The transformations are the same for the case of a three-phase current; you simply replace the V_a , V_b , V_c , V_d , V_q , and V_0 variables with the I_a , I_b , I_c , I_d , I_q , and I_0 variables.

References

[1] Krause, P. C. *Analysis of Electric Machinery*. New York: McGraw-Hill, 1994, p.135.

Bibliography

- [1] Boldea, Ion, and S.A. Nasar. *Electric Drives*. Boca Raton, Florida: CRC Press, 1999.
- [2] Bose, B. K. *Modern Power Electronics and AC Drives*. Upper Saddle River, NJ: Prentice-Hall, 2002.
- [3] Casoria, S., P. Brunelle, and G. Sybille. "Hysteresis Modeling in the MATLAB/Power System Blockset." *Electrimacs 2002*, École de technologie supérieure, Montreal, 2002.
- [4] Dommel, H. "Digital Computer Solution of Electromagnetic Transients in Single and Multiple Networks." *IEEE Transactions on Power Apparatus and Systems*. Vol. PAS-88, No. 4, April, 1969.
- [5] Frame, J.G., N. Mohan, and Tsu-huei Liu. "Hysteresis modeling in an Electro-Magnetic Transients Program." *IEEE PES winter meeting*, New York, January 31 to February 5, 1982.
- [6] Grelet, G. and G. Clerc. *Actionneurs électriques*. Paris: Éditions Eyrolles, 1997.
- [7] Grenier, D., L.-A. Dessaint, O. Akhrif, Y. Bonnassieux, and B. LePioufle. "Experimental Nonlinear Torque Control of a Permanent Magnet Synchronous Motor Using Saliency." *IEEE Transactions on Industrial Electronics*. Vol. 44, No. 5, October 1997, pp. 680-687.
- [8] Grondin, R., I. Kamwa, L. Soulieres, J. Potvin, and R. Champagne. "An approach to PSS design for transient stability improvement through supplementary damping of the common low frequency." *IEEE Transactions on Power Systems*, 8(3), August 1993, pp. 954-963.
- [9] Harunur, M. R. *Power Electronics*. Upper Saddle River, NJ: Prentice-Hall, 1988.
- [10] Hingorani, N. G. and L. Gyugyi. "Understanding FACTS; Concepts and Technology of Flexible AC Transmission Systems." *IEEE Press book*, 2000.
- [11] Krause, P. C. *Analysis of Electric Machinery*. New York: McGraw-Hill, 1986.
- [12] Krause, P. C., O. Wasynczuk, and S. D. Sudhoff. *Analysis of Electric Machinery*. IEEE Press, 2002.
- [13] Kundur, P. *Power System Stability and Control*. New York: McGraw-Hill, 1994.
- [14] Mohan, N., T. M. Undeland, and W. P. Robbins. *Power Electronics: Converters, Applications, and Design*. New York: John Wiley & Sons, 1995.
- [15] Nise, Norman S. *Control Systems Engineering*. . Addison-Wesley Publishing Company, 1995.
- [16] Norton, Robert L. *Machine Design*. . Upper Saddle River, NJ: Prentice Hall, 1998.
- [17] Rajagopalan, V. *Computer-Aided Analysis of Power Electronic Systems*. New York: Marcel Dekker, 1987.
- [18] "Recommended Practice for Excitation System Models for Power System Stability Studies." *IEEE Standard 421.5*, 2002.

- [19] Séguier, Guy. *Electronique de puissance*. Paris: Dunod, 1999.
- [20] Sen, P.C. *Thyristor DC Drives*. New York: John Wiley & Sons, 1981.
- [21] Shepherd, C. M. "Design of Primary and Secondary Cells - Part 2. An equation describing battery discharge." *Journal of Electrochemical Society*. Volume 112, Jul. 1965, pp. 657-664.
- [22] Tremblay, O. *Modélisation, simulation et commande de la machine synchrone à aimants à force contre-électromotrice trapézoïdale*. École de Technologie Supérieure, 2006.
- [23] Tremblay, O., Dessaint, L.-A., Dekkiche, A.-I. "A Generic Battery Model for the Dynamic Simulation of Hybrid Electric Vehicles." *Vehicle Power and Propulsion Conference, 2007. VPPC 2007. IEEE* 9-12 Sept. 2007, pp. 284-289.