

SimPowerSystems™ Release Notes



MATLAB® & SIMULINK®

How to Contact MathWorks



Latest news: www.mathworks.com
Sales and services: www.mathworks.com/sales_and_services
User community: www.mathworks.com/matlabcentral
Technical support: www.mathworks.com/support/contact_us



Phone: 508-647-7000



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

SimPowerSystems™ Release Notes

© COPYRIGHT 2003–2015 by 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.

R2015a

Simscape Components	1-2
Asynchronous machines with SI parameterization	1-2
Synchronous Machine Model 2.1 blocks	1-2
Zigzag-Delta1-Wye and Zigzag-Delta11-Wye Transformer blocks	1-2
Average-Value Inverter block	1-2
Ideal Rectifier block name change	1-2
Featured Examples	1-2
Specialized Technology	1-4
PV Array block	1-4
Annotation and export options for Load Flow Tool	1-4
power_customize function for creating custom Specialized Technology blocks	1-4
Three-limb core option for three-phase transformer blocks ..	1-5
Interpolation and Store State-Space Matrices options for Tustin solver	1-5
New machine block dialog boxes and function access	1-5
New powergui dialog box and tools	1-6
Block library changes	1-6
Phasor Simulation Method examples	1-7

R2014b

Simscape Components	2-2
Harmonics option in Voltage Source block	2-2

Wye-Connected Variable Load blocks	2-2
Three-Level Converter block	2-2
Machine Mechanical Power and Synchronous Machine Field Circuit interface blocks	2-3
Ideal Rectifier block	2-3
Removal of Initial state parameter on Circuit Breaker block .	2-3
New examples	2-3
Specialized Technology	2-4
Tustin/Backward Euler discretization method	2-4
Bipolar PWM option in PWM Generator (2-Level) block	2-4
Parameter estimation for Stern model in Supercapacitor block	2-4
Usability enhancements for Switched Reluctance block	2-4
Initialization enhancements for examples using initial condition vectors	2-5

R2014a

New names for SimPowerSystems technologies	3-2
Simscape Components	3-3
Time-Based Fault and Enabled Fault blocks	3-3
Saturation option for Synchronous Machine models	3-3
Harmonic analysis functions	3-3
Primary and secondary winding blocks with new variables tab	3-4
Specialized Technology	3-6
Aircraft fuel cell hybrid emergency power system example . .	3-6
Text labels for Machine Block Bus output signals	3-6
Three-phase 3-Level Inverter with direct specification of diode and IGBT characteristics	3-7
Improved breaker block set interfaces	3-7
power_cableparam function documentation enhancement . .	3-7

Third Generation 4-2

- Third Generation block libraries based on Simscape technology 4-2
- Expandable three-phase electrical ports for single-line diagrams 4-2
- Machines and transformers customizable using Simscape language, with examples 4-2

Second Generation 4-4

- Electric Drive blocks with improved measurement outputs . . 4-4
- Examples that use blocks from the Control and Measurements library 4-4
- New examples 4-4

Control and measurement blocks automatically configured based on simulation mode 5-2

Supercapacitor block 5-2

power_analyze function that now calculates state-space matrix based on switch state 5-2

Additional diagnostic controls for SimPowerSystems models 5-3

New examples 5-3

Seven IEEE automatic voltage regulator (AVR) excitation blocks	6-2
power_new function that creates new Simulink model with recommended settings for SimPowerSystems models ...	6-2
Optional Simscape mechanical rotational port for SimPowerSystems electric drives models	6-2
Simscape Interface blocks now permitting user-defined initial conditions	6-2
power_fftscope function now generating plots directly from MATLAB command prompt, with additional display options	6-3
power_lineparam function accessible programmatically ...	6-3
SimPowerSystems blocks supporting Simulink Parameter Objects as dialog box parameters	6-4

Simscape Rotational Port Available as Mechanical Input for Machine Blocks	7-2
New Setup Function for Permanent Magnet Synchronous Machine Block	7-2
Additional Solver Type Option Available for Discrete Mode	7-2
Distributed Resources Library Renamed to Renewable Energy	7-3
New SimPowerSystems Demos	7-3

R2011b

SimPowerSystems Software Now Requires Simscape Product	8-2
Interfacing with Simscape Electrical Domains	8-2
Sharing Models Using Simscape Editing Modes	8-2
Block Library Links Must Be Resolved	8-2
Changes to SimPowerSystems Demos	8-3

R2011a

New Load Flow Tool	9-2
New Asynchronous Machine Block Setup Function	9-2
5-Phase Synchronous Machine Model Available	9-3
SimState Support	9-3
New SimPowerSystems Demo	9-3

R2010b

Double Squirrel-Cage Rotor Option Available in Asynchronous Machine Block	10-2
Enhanced Code Generation Capabilities	10-2
New SimPowerSystems Demos	10-2

R2010a

Permanent Magnet Model Available in DC Machine Block	11-2
---	-------------

R2009b

Enhanced <code>power_cableparam</code> Function	12-2
Changes to the Fuel Cell Stack Block	12-2
New SimPowerSystems Demos	12-2

R2009a

Powergui Tools Are Also Available as Standalone Command-Line Functions	13-2
Enhancements to the Ideal Switching Algorithm	13-2
Powergui Block No Longer Added Automatically	13-2
Changes to the Battery Block	13-2
New SimPowerSystems Demos	13-3

R2008b

New Ideal Switching Algorithm	14-2
--	-------------

Changes to the Universal Bridge Block	14-2
New SimPowerSystems Demos	14-2

R2008a

New Fuel Cell Stack Block	15-2
Initial Conditions Can Be Specified for the Permanent Magnet Synchronous Machine Block	15-2
Multiple Discretization Rates within a Model Now Available	15-2
New SimPowerSystems Demos	15-3

R2007b

New Battery Block	16-2
New Stepper Motor Block	16-2
Three New Transformer Blocks	16-2
New Measurement Option Available for the PI Section Line Block	16-3
New SimPowerSystems Demos	16-3
Renamed psbhysteresis Command	16-4

R2007a

New Brushless DC Motor Drive Block	17-2
Automated Conversion of Version 2 Models Is No Longer Supported	17-2
New SimPowerSystems Demos	17-2

R2006b

Mechanical Input Parameter Lets You Connect SimMechanics or SimDriveline Blocks to Electric Drives by Specifying Motor Speed as Block Input	18-2
--	-------------

R2006a

Average Values of electricdrivelib Blocks	19-2
Transformer Blocks with SI Units Are Available	19-2
Open Circuit Option Is Added for the RLC Blocks	19-2
New Demos and Enhancements to Existing Demos	19-3

R14SP3

Bug Fixes

New Blocks in the Machines Library of powerlib	21-2
Enhancements to Existing Blocks of the Machines Library	21-2
Branch Type Parameter of the RLC Branch Blocks	21-2
Average Values of electricdrivelib Blocks	21-2
Obsolete Blocks	21-3

Bug Fixes

R2015a

Version: 6.3

New Features

Bug Fixes

Simscape Components

Asynchronous machines with SI parameterization

The Asynchronous Machine Squirrel Cage (fundamental, SI) and the Asynchronous Machine Wound Rotor (fundamental, SI) blocks allow you to use SI values for parameterization.

Synchronous Machine Model 2.1 blocks

The Synchronous Machine Model 2.1 (fundamental) and the Synchronous Machine Model 2.1 (standard) blocks use a simplified version of Park's transform and a simplified representation suitable for large-scale stability studies. The blocks neglect the effect of speed variation on stator voltages.

Zigzag-Delta1-Wye and Zigzag-Delta11-Wye Transformer blocks

The Zigzag-Delta1-Wye Transformer and Zigzag-Delta11-Wye Transformer blocks model linear, nonideal transformers that you can use to reduce the total harmonic distortion in your model. The blocks have three-limb cores with primary windings configured in a zigzag connection. The secondary windings have delta (1 or 11 o'clock) and wye configurations.

Average-Value Inverter block

The Average-Value Inverter block models an average-value, full-wave inverter that converts DC voltage to three-phase AC voltages and converts three-phase AC power demand to DC power demand. The Average-Value Inverter block does not yield the harmonics that are typically associated with a detailed inverter representation. The block converts power with fixed losses.

Ideal Rectifier block name change

The Ideal Rectifier block name has changed to Average-Value Rectifier.

Featured Examples

Four examples are introduced in this version:

- The Custom Inductor (B-H Curve) example compares the behaviors of linear and nonlinear inductors. The inductors are implemented in the Simscape™ language.
- The Custom Transformer (B-H Curve) example shows the calculation and confirmation of a nonlinear transformer core magnetization characteristic. The transformers are implemented in the Simscape language.
- The Marine Full Electric Propulsion Power System example shows a representative marine electrical power system with base load, hotel load, bow thrusters and an average-value propulsion rectifier.
- The Three-Phase Custom Transforms example compares the results of four different three-phase transforms.

Specialized Technology

PV Array block

The PV Array block allows you to model series-connected and parallel-connected PV modules. You can use a preset PV module from NREL System Advisor Model or your own PV modules.

These examples use the PV Array block:

- 250-kW Grid-Connected PV Array
- 400-kW Grid-Connected PV Farm (Average Model)
- Partial Shading of a PV Module
- Single-Phase, 240 Vrms, 3500 W Transformerless Grid-Connected PV Array

Annotation and export options for Load Flow Tool

The Load Flow Tool has a new option to annotate the load flow calculations for models that contain a Load Flow Bus block. To use this option, at the MATLAB® command prompt, enter `power_loadflow(' -v2', gcs, 'AddBuses')`. You can also add the annotation using the `powergui` block in your model. To open the tool, open the `powergui` block. In the dialog box, on the **Tools** tab, click **Load Flow**, and then click **Add bus blocks**.

The Load Flow Tool also has a new option to generate reports. To open the tool, open the `powergui` block. In the dialog box, on the **Tools** tab, click **Load Flow**, and then click **Compute**. When the dialog box displays **Load Flow Converged!**, click **Report**. Specify the report format and save the `LoadFlowResults_power_machines` file. When the load flow report dialog box indicates that your load flow report has been created, click **Open file** to view the report.

`power_customize` function for creating custom Specialized Technology blocks

The `power_customize` function helps you create custom SimPowerSystems™ Specialized Technology blocks. To open the function, open the `powergui` block. In the dialog box, on the **Tools** tab, click **Customize SPS blocks**. You can also open the tool by entering `power_customize` at the MATLAB command prompt.

Three-limb core option for three-phase transformer blocks

In prior releases, the Three-Phase Transformer (Three Windings) and Three-Phase Transformer (Two Windings) blocks modeled three-phase transformers using three separate single-phase transformers. Now you can model a transformer using a single three-limb or five-limb core representation for high-validity results. Use the **Core type** parameter on the **Configuration** tab of either three-phase transformer block dialog box to specify the representation.

These examples show the validity of core representation simulation results:

- Three-Phase Three-Limb (Core-Type) Two-Winding Transformer
- Three-Phase Five-Limb (Shell-Type) Three-Winding Transformer

Interpolation and Store State-Space Matrices options for Tustin solver

The discrete Tustin solver on the powergui block has two options for increasing simulation speed without sacrificing simulation accuracy. The interpolation option increases simulation speed by enabling the solver to interpolate to identify power electronic device gate transitions that occur between large sample times. To use the interpolation option, open the powergui block. In the dialog box, on the **Solver** tab, select **Discrete** for the simulation type and **Tustin** for the solver type. Select the **Interpolate** check box.

The **Store state-space matrices** option accelerates simulation by enabling the solver to store and reuse matrix computation results. The option is visible only when the simulation type is set to **Discrete**, the solver type is set to **Tustin**, and the **Interpolate** option is selected. Select **Store state-space matrices** and specify a buffer size.

New machine block dialog boxes and function access

The **Configuration** tab on the Asynchronous Machine block dialog box now contains a **Preset parameters** section. The section includes the **Squirrel-cage preset model** parameter (formerly called **Preset model**). It also includes the **Double squirrel-cage preset model** parameter, which has an option to open the parameter estimator (the `power_AynchronousMachineParams` function) for estimating machine parameters based on manufacturer specifications. The **Rotor type** parameter is now the first parameter on the **Configuration** tab and is no longer disabled when a preset model is selected.

The Permanent Magnet Synchronous Machine block dialog box **Advanced** tab now contains a **Machine parameters** section. It includes a **Compute from standard manufacturer specifications** option. The option opens the `power_PMSynchronousMachineParams` function.

New powergui dialog box and tools

You can use the new **Tools** tab on the powergui block to open these tools:

- Steady-State Voltages and Currents Tool
- Initial States Tool
- Load Flow Tool
- Machine Initialization Tool
- Impedance vs Frequency Measurement Tool
- FFT Analysis Tool
- Linear System Analyzer
- Hysteresis Design Tool
- Compute RLC Line Parameters Tool
- Generate Report Tool
- `power_customize`

The parameters that were on the **Load Flow** tab are now included on the **Preferences** tab on the powergui block. The **Preferences** dialog box also includes a new option that allows you to use a different powergui block in each subsystem in your model.

Block library changes

The `powerlib` library is no longer the main SimPowerSystems Specialized Technology library. The main library is now the `simpowersystems_ST` library. To access the `simpowersystems_ST` library through the Simulink® Library Browser, go to **Simscape > SimPowerSystems > Specialized Technology**. To open the main library at the MATLAB command prompt, enter `simpowersystems_ST`.

The `simpowersystems_ST` library contains these libraries:

- Fundamental Blocks

- Control & Measurements
- Electric Drives — Formerly included in the Applications library.
- FACTS — Formerly named the Flexible AC Transmission Systems (FACTS) library and included in the Applications library.
- Renewables — Formerly named the Renewable Energy library and included in the Applications library.

You can access the Fundamental Blocks library in the Simulink Library Browser, or by entering, `powerlib` at the command prompt. The Fundamental Blocks library contains the `powergui` block. It also contains these libraries that were previously included in the `powerlib` library:

- Electrical Sources
- Elements
- Interface Elements
- Machines
- Measurements
- Power Electronics

The Solar library is a new library that contains the PV Array block. The Solar library is included in the FACTS library.

The Excitation Systems library is included in the Machines library.

The Applications library is no longer included in the Specialized Technology library.

Phasor Simulation Method examples

The Simplified Model of a Small Scaled Micro-Grid and 24-hour Simulation of a Vehicle-to-Grid (V2G) System examples use the phasor simulation method to accelerate micro-grid simulations.

R2014b

Version: 6.2

New Features

Bug Fixes

Simscape Components

Harmonics option in Voltage Source block

The Voltage Source block has a **Harmonics** tab that allows you to configure the block as a three-phase voltage source that maintains a fundamental frequency and harmonics across its terminals. You can specify the harmonic orders and the harmonic to peak magnitude ratios of the AC voltage waveform.

Wye-Connected Variable Load blocks

The Wye-Connected Variable Load and Wye-Connected Variable Load (lagging) blocks model three-phase, wye-wired configurations that are capable of representing real, or real and reactive, time-varying loads. You parameterize the blocks via their physical input signals.

The Wye-Connected Variable Load block calculates the resistance required to dissipate the real power that its physical input signal specifies. The Wye-Connected Variable Load (lagging) block calculates the resistance and inductance required to dissipate the real and reactive powers that its physical input signals specify.

Three-Level Converter block

The Three-Level Converter block models a twelve-pulse, three-phase, three-level controlled converter circuit that you can use to connect a three-phase AC network to a three-level DC network. Each of its three bridge arms contains four switching devices and the associated anti-parallel diodes. You can use any of these blocks for the switching devices:

- GTO
- Ideal Semiconductor Switch
- IGBT
- MOSFET

You use the Twelve-Pulse Gate Multiplexer block to multiplex the connections into a single vector to provide input to the Three Level Converter block.

Machine Mechanical Power and Synchronous Machine Field Circuit interface blocks

The machine mechanical power blocks allow you to specify mechanical power to or from a machine block using either an SI, Machine Mechanical Power (SI), or per-unit, Machine Mechanical Power (pu), physical signal. The machine power blocks contain mechanical rotational reference and machine inertia components in order to accurately represent transient behavior.

The synchronous machine field circuit blocks allow you to use a physical signal to apply either SI, Synchronous Machine Field Circuit (SI), or per-unit, Synchronous Machine Field Circuit (pu), voltage to a machine field circuit. It also allows you to use a physical signal to measure the current in the field circuit.

Ideal Rectifier block

The Ideal Rectifier block models an ideal, full-wave, six-pulse rectifier that converts three-phase AC voltages to DC voltage and DC power demand to three-phase AC power demand. The Ideal Rectifier block does not yield the harmonics that are typically associated with a detailed rectifier representation. It converts power ideally to provide a faster simulation than the non-idealized Rectifier block provides.

Removal of Initial state parameter on Circuit Breaker block

In previous releases, there was an **Initial state** parameter on the dialog box of the Circuit Breaker and Single-Phase Circuit Breaker blocks. You set the parameter to **open** or **closed** to indicate the state of the breaker at the start of the simulation. The breaker blocks now calculate the initial state as a function of the input control voltage, v_T , at $\text{time} = 0$. Therefore, you are no longer required to set the initial state parameter.

New examples

Examples introduced in this version are:

- Three-Phase Asynchronous Wind Turbine Generator
- Three-Phase Custom Simplified Synchronous Machine

Specialized Technology

Tustin/Backward Euler discretization method

The powergui block provides a discretization solver option that combines the advantages of the Tustin and the Backward-Euler methods in a single method. The Tustin/Backward Euler (TBE) solver is now the recommended discretization method for all discrete simulation.

Bipolar PWM option in PWM Generator (2-Level) block

You can specify the generator type for the PWM Generator (2-Level) block to use either unipolar or bipolar PWM modulation for single-phase full-bridge control. The single-phase full-bridge generator uses unipolar modulation, which produces higher quality AC waveforms. The single-phase full-bridge, bipolar generator uses bipolar modulation, which produces low-variation, common-mode voltage.

The block allows you to choose from three different techniques to sample the reference signal Uref. Natural sampling models the behavior of an analog implementation of a PWM generator. Asymmetrical regular sampling is a single-edge technique that samples Uref only at the valley of the carrier. Symmetrical regular sampling is a double-edge technique that samples Uref at both the valley and the peak of the carrier.

Parameter estimation for Stern model in Supercapacitor block

The Supercapacitor block no longer requires you to specify hard-to-find values for surge voltage, leakage current on the **Parameters** tab. The block uses default values for these parameters and for the overpotential and charge transfer parameters on the **Stern** tab. You can set the **Stern** parameter to values that you find, via trial-and-error simulation (or data sheet specification testing using the Optimization Toolbox™), to match the performance of your supercapacitor.

The block can simulate your supercapacitor's self-discharge characteristic using current and voltage information that you typically find on a data sheet.

Usability enhancements for Switched Reluctance block

The Switched Reluctance Motor block offers you three preset motors types. The block contains all of the model parameters that it needs to simulate a 10, 60, or 75 kW motor.

You can specify a model of your motor by entering the specifications directly in the dialog box or by referring to them in a MATLAB MAT-file.

Initialization enhancements for examples using initial condition vectors

These renewable energy examples use an initial state vector to start in steady state:

- Wind Farm (IG)
- Wind-Turbine Asynchronous Generator in Isolated Network
- Wind Farm (DFIG Phasor Model)
- Wind Farm - DFIG Detailed Model
- Wind Farm - DFIG Average Model
- Wind Farm - Synchronous Generator and Full Scale Converter (Type 4) Detailed Model
- Wind Farm - Synchronous Generator and Full Scale Converter (Type 4) Average Model

When you make changes to the model, you need to regenerate or disable the initial state vector to avoid getting a Simulink error when you start the simulation. Follow the instructions in the initialization file in the example's model properties to regenerate the initial state vector.

R2014a

Version: 6.1

New Features

Bug Fixes

Compatibility Considerations

New names for SimPowerSystems technologies

SimPowerSystems Version 6.1 introduces new names for the SimPowerSystems Second and Third Generation technologies. SimPowerSystems Simscape Components is the new name for the Third Generation technology. SimPowerSystems Specialized Technology is the new name for the Second Generation technology.

Note: The terms Simscape Components and Specialized Technology refer to the SimPowerSystems technology used. A SimPowerSystems *version* can contain one or both technologies.

For more information, see Comparison of Simscape Components and Specialized Technology.

Simscape Components

Time-Based Fault and Enabled Fault blocks

The Time-Based Fault block and the Enabled Fault block model any permutation of a single-phase, two-phase, or three-phase grounded or ungrounded fault. You specify the fault permutation and the values for resistance and conductance.

The Time-Based Fault block is activated and deactivated during simulation according to time-based parameters that you specify. The Enabled Fault block is activated when the input signal exceeds a threshold value that you specify. It is deactivated when the signal is equal to or less than the threshold value.

Saturation option for Synchronous Machine models

You can now simulate magnetic saturation on these synchronous machine blocks:

- Synchronous Machine Round Rotor (fundamental)
- Synchronous Machine Round Rotor (standard)
- Synchronous Machine Salient Pole (fundamental)
- Synchronous Machine Salient Pole (standard)

These blocks have a new **Saturation** tab, which contains the **Magnetic saturation representation** parameter. When you specify the field current and air-gap voltage per-unit saturation data, the block generates a per-unit air-gap voltage versus field current open-circuit lookup table. The block then uses the lookup table data to calculate a saturation factor.

Harmonic analysis functions

There are three new functions for performing harmonic analysis:

- `pe_getHarmonics`
- `pe_calculateThdPercent`
- `pe_plotHarmonics`

The `pe_getHarmonics` function returns the harmonic orders, magnitude, and fundamental frequency when you input a Simscape logging node variable.

To run the `pe_getHarmonics` function, at the MATLAB command prompt, type:

```
pe_getHarmonics( logging_node )
```

where *logging_node* is the Simscape logging node variable.

The `pe_calculateThdPercent` function returns the Total Harmonic Distortion (THD) percentage when you use the harmonic order and the harmonic magnitude as input arguments.

To run the `pe_calculateThdPercent` function, at the MATLAB command prompt, type:

```
pe_calculateThdPercent( harmonicOrder, harmonicMagnitude )
```

where

- *harmonicOrder* is a vector of harmonic orders.
- *harmonicMagnitude* is a vector of harmonic magnitudes.

The `pe_plotHarmonics` function plots the harmonic data when you input a Simscape logging node variable.

To run the `pe_plotHarmonics` function, at the MATLAB command prompt, type:

```
pe_plotHarmonics( logging_node )
```

where *logging_node* is the Simscape logging node variable.

The Harmonic Analysis of a Three-Phase Rectifier example shows you how to use the three functions.

Primary and secondary winding blocks with new variables tab

The Primary Winding and Secondary Winding blocks now contain a **Variables** tab, which allows you to specify target value and priority for a new initialization process.

For more information see the Simscape Release Notes item, Variables tab for specifying target value and priority for new initialization process

Compatibility Considerations

In previous releases, the Primary Winding and Secondary Winding blocks contained parameters that let you specify an initial value for some internal block variables at the

start of simulation. These parameters have now been removed. The following table lists the initialization parameters that have been removed from block dialogs and the names of the corresponding block variables:

Block Name	Parameter Name	Variable Name
Primary Winding	Leakage inductance initial current	Leakage inductance current
	Magnetization inductance initial current	Magnetization inductance current
	Initial magnetic flux	Magnetic flux
Secondary Winding	Leakage inductance initial current	Leakage inductance current
	Initial magnetic flux	Magnetic flux

Legacy models using these blocks are affected by this change. If a block used the initialization parameter, then, once you open the model in the current release, this parameter value is no longer visible and is not automatically mapped to the corresponding variable value. The simulation results will only stay the same if you set the target value of the variable to be that of the original parameter and set the parameter priority to High.

Specialized Technology

Aircraft fuel cell hybrid emergency power system example

The Energy Management Systems for a Hybrid Electric Source (Application for a More Electric Aircraft) example illustrates a simulation model of a fuel-cell-based emergency power system for More Electric Aircraft (MEA). In MEA, as the landing-gear and flight-control systems become more electrically driven, the peak electrical load seen by the emergency power system increases. This increased load puts conventional ram air turbine (RAT) and air-driven generator (ADG) emergency power systems, which exhibit near-zero power production at lower speeds, at risk for overloading. This example presents an alternative emergency power system based on fuel cells, lithium-ion batteries, and supercapacitors that is more capable of handling the increased electrical load.

Text labels for Machine Block Bus output signals

In previous releases, you could only use alphanumeric signal names to identify bus labels on these SimPowerSystems machine blocks:

- Asynchronous Machine SI Units
- Asynchronous Machine pu Units
- DC Machine
- Permanent Magnet Synchronous Machine
- Simplified Synchronous Machine SI Units
- Simplified Synchronous Machine pu Units
- Single Phase Asynchronous Machine
- Synchronous Machine SI Fundamental
- Synchronous Machine pu Fundamental
- Synchronous Machine pu Standard

Machine blocks now have a **Measurement output** parameter that gives you the option to identify bus labels with alphanumeric signal names that are compatible with model referencing. To enable the option to convert bus signal names to valid signal names, on the machine block **Configuration** tab, under **Measurement output**, select the **Use signal names to identify bus labels** check box.

The **Use signal names to identify bus labels** check box is cleared by default.

Three-phase 3-Level Inverter with direct specification of diode and IGBT characteristics

In previous versions, the Loss Calculation in a Three-Phase 3-Level Inverter example allowed you to choose between three different pre-specified commercial (ABB® or Fuji) components for the **IGBT Type** and **Diode Type** parameters in the Half-bridge IGBT with Loss Calculation block dialog box. To use a value other than the prespecified commercial values, you entered your own IGBT and diode specifications via MATLAB MAT-files.

In the current version, you can enter your preferred IGBT and diode values directly to the Half-bridge IGBT with Loss Calculation block dialog box in the new **IGBT** and **Diode** tabs. You can click the **Save** button on either tab to save the values to MAT-files for future use. The tabs also contain a **Plot Characteristics** button that you can use to plot your IGBT or diode values.

The prespecified commercial (ABB or Fuji) component specifications are still available as MAT-files that you can upload by clicking the **Load** button on the **IGBT** or **Diode** tab.

Improved breaker block set interfaces

The dialog boxes for the breaker blocks in the Elements library have been standardized:

- Breaker
- Three-Phase Breaker
- Three-Phase Fault

power_cableparam function documentation enhancement

The power_cableparam function reference page has been expanded to include a formula for calculating the *Total Section* of a screen conductor.

R2013b

Version: 6.0

New Features

Bug Fixes

Third Generation

Third Generation block libraries based on Simscape technology

SimPowerSystems Version 6.0 introduces Third Generation technology to model, simulate, and analyze electrical power systems. Third Generation technology provides libraries of component models fully compatible with the Simscape Foundation library. These component models are also fully compatible with Simscape technology, including local solver and data logging.

Version 6.0 provides full support for both Second Generation and Third Generation functionality, including block sets and tools. With Version 6.0, you have the option to create and simulate a model using Second or Third Generation technology. Version 6.0 software gives you the two full block libraries, labeled Second Generation (SPS 2G) and Third Generation (SPS 3G). The Second Generation (SPS 2G) libraries contain all the blocks available prior to Version 6.0, and the Third Generation (SPS 3G) libraries contain 55 new blocks.

Note: The term *generation* — e.g., “Third Generation” — refers to the SimPowerSystems technology used. A SimPowerSystems *version* can contain one or both *generations*.

For more information, see Comparison of Second and Third Generation Technologies.

Expandable three-phase electrical ports for single-line diagrams

Third Generation library blocks are equipped with the new three-phase conserving electrical ports, which you can individually expand and collapse. Three-phase ports on Third Generation blocks are collapsed by default, to support single-line diagrams. You can optionally expand them to separate the phases, for example, if you need to inject a single-line-to-ground fault into your circuit. For more information, see Three-Phase Ports.

Machines and transformers customizable using Simscape language, with examples

The ThreePhaseExamples library, included in the product examples, contains the following custom three-phase components:

- Permanent Magnet Synchronous Motor
- Synchronous Machine
- Zigzag Transformer

You can use these simplified example models to write your own machine and transformer component files.

To open the custom library, type `ThreePhaseExamples_lib` at the MATLAB Command prompt. Double-click any block in the library to open its dialog box, and then click the `View source for BlockName` link in the block dialog box to open the Simscape source file for this block in the MATLAB Editor.

To customize the block for your application, edit the source file and save it in a package directory.

For more information on writing customized component files, see [Custom Components](#). For information on packaging and deploying Simscape component files, see [Simscape File Deployment](#).

Second Generation

Electric Drive blocks with improved measurement outputs

A new parameter, **Output bus mode**, has been added to all the AC and DC drives in the Electric Drives library. The parameter lets you select between two options:

- **Multiple output buses** — The block has separate Motor, Conv, and Ctrl outputs, as before. This is the default.
- **Single output bus** — The three separate outputs are combined into a single bus output called Out.

Examples that use blocks from the Control and Measurements library

Example models that were using blocks from the Extras library have been modified to use the blocks from the Control and Measurements library, introduced in R2013a.

New examples

Examples introduced in this version are:

- 100-kW Grid-Connected PV Array (Detailed Model)
- 100-kW Grid-Connected PV Array (Average Model)

R2013a

Version: 5.8

New Features

Bug Fixes

Compatibility Considerations

Control and measurement blocks automatically configured based on simulation mode

A new Control and Measurements library, with 47 new blocks, provides improved control and measurement blocks for use in power electrical models. The new library is a restructured and enhanced version of the old Extras library. All the blocks in the new library have associated block reference pages that refer to small simulation examples. New design techniques are used to improve model robustness and efficiency. Compared to the old Extras library, the new Control and Measurements library eliminates duplicate of continuous, discrete, and phasor versions of the same block, by merging them into a single block. These blocks feature a simple mechanism that automatically configures the block based on simulation mode.

Compatibility Considerations

The Control and Measurements library is intended to replace the Extras library. The Extras block library is no longer listed in the Library Browser. It is still available for compatibility purposes, and you can open it by typing `powerlib_extras` at the MATLAB command prompt.

If your legacy models contain blocks from the Extras library, they will continue to work. However, MathWorks recommends that you use the blocks from the Control and Measurements library in your new models.

Supercapacitor block

The Supercapacitor block in the Electric Drives library implements a generic model parameterized to represent most popular types of supercapacitor. The Supercapacitor Model example illustrates a simple hybridization of a supercapacitor with a battery.

power_analyze function that now calculates state-space matrix based on switch state

The `power_analyze` function now has an option that allows you to compute ABCD matrices of a given model based of any given switch status vector provided as an input argument.

Additional diagnostic controls for SimPowerSystems models

The **Preferences** tab of the powergui block parameters dialog box has been enhanced to give you better control on what you want to see displayed in the command window during simulation. The **Display SimPowerSystems warnings and messages** check box, available in previous releases, is now replaced with two new check boxes:

- **Disable SimPowerSystems warnings** allows you to deactivate all the warning messages that are displayed when you run a simulation. The warning messages are displayed by default. Select this check box if you are aware of the warning and do not want to be bothered by a message on subsequent simulation runs.
- **Display SimPowerSystems compilation messages** allows you to display the compilation messages issued during the analysis of the model. These messages are not displayed by default.

New examples

Examples introduced in this version are:

- Loss Calculation in a Three-Phase 3-Level Inverter
- Supercapacitor Model

R2012b

Version: 5.7

New Features

Bug Fixes

Seven IEEE automatic voltage regulator (AVR) excitation blocks

The Machines library now contains a sublibrary, named Excitation Systems, with seven new blocks. These blocks represent the standard types of IEEE® AVR excitation systems. For more information, see the block reference pages.

The `power_machines` example model, which previously used a generic excitation system, has been updated to use any of the seven models in the Excitation Systems (AVRlib) library.

`power_new` function that creates new Simulink model with recommended settings for SimPowerSystems models

When you type `power_new` at the MATLAB Command prompt, the software creates a new model, with the recommended solver `ode23tb` and with required Solver Configuration and Powergui blocks already on the canvas. It also opens the `power_new_palette` library.

After using `power_new`, continue developing your model by copying the blocks from the `power_new_palette` library, as needed, and adding other blocks from the Simulink and Simscape libraries.

Optional Simscape mechanical rotational port for SimPowerSystems electric drives models

Blocks in the Electric Drives library now have an additional option, `Mechanical rotational port`, for the **Mechanical input** parameter. When you select this option, the mechanical input port of the block (Tm or w), changes to a Simscape conserving rotational port S, which you can connect directly to a mechanical rotational port of a block from Simscape libraries. Therefore, for example, you can model the mechanical part of your system using Simscape and SimDriveline™ blocks, and then connect it directly to the electric drive through the mechanical rotational port S. For details, see the block reference pages.

Simscape Interface blocks now permitting user-defined initial conditions

There are several enhancements to the blocks in the Interface Elements library that let you connect SimPowerSystems and Simscape electrical circuits:

- Depending on the type of the interface block, you can specify the initial input and output voltage or current in the filter that is used to break an algebraic loop between SimPowerSystems and Simscape circuits. The Current-Voltage Simscape Interface and Current-Voltage Simscape Interface (gnd) blocks now have an additional **Initial Voltage** parameter. The Voltage-Current Simscape Interface and Voltage-Current Simscape Interface (gnd) blocks have an **Initial Current** parameter.
- The **Filter type** parameter in each of these blocks now has two additional values: **First-order input filtering** and **Second-order input filtering**. These options provide access to the input filtering functionality in the underlying Simulink-PS Converter block. For more information, see the Simulink-PS Converter block reference page.
- The block dialog boxes for each of these blocks now have an additional check box, **Show measurement ports**. If you select this check box and click **Apply** or **OK**, the block icon displays an additional port **m**. This is the output port of the underlying voltage or current sensor, depending on the type of the interface block. The Current-Voltage Simscape Interface and Current-Voltage Simscape Interface (gnd) blocks contain a current sensor. The Voltage-Current Simscape Interface and Voltage-Current Simscape Interface (gnd) blocks contain a voltage sensor. The output port **m** provides access to these measurements.

power_fftscope function now generating plots directly from MATLAB command prompt, with additional display options

You can now use the `power_fftscope` function directly at the command line, which provides the ability to automate the use of the tool. The THD computation has been improved to include the inter-harmonics in the THD calculations. Two new options are available in the **Display style** drop-down list:

- **Bar** (relative to DC component)
- **List** (relative to DC component)

For more information, see the `power_fftscope` reference page.

power_lineparam function accessible programmatically

You can now programmatically access a structure variable with default line geometry parameters, and use it as a template variable to configure a new line geometry and to compute RLC line parameters.

SimPowerSystems blocks supporting Simulink Parameter Objects as dialog box parameters

You can now use Simulink Parameter Objects when entering parameter values into the block dialog boxes. For more information, see [Specify Parameter Values](#) in the Simulink documentation.

R2012a

Version: 5.6

New Features

Bug Fixes

Simscape Rotational Port Available as Mechanical Input for Machine Blocks

The **Mechanical input** parameter has a new option, `Mechanical rotational port`, which has been added for the following machines:

- Simplified Synchronous Machine
- Synchronous Machine
- Asynchronous Machine
- Single Phase Asynchronous Machine
- DC Machine
- Permanent Magnet Synchronous Machine

When you select this option, the mechanical input port of the block (`Tm` or `w`), changes to a Simscape conserving rotational port `S`, which you can connect directly to a mechanical rotational port of a block from Simscape libraries. This allows you, for example, to model the mechanical part of your system using Simscape and SimDriveline blocks, and then connect it directly to the electrical machine through the mechanical rotational port `S`. See the block reference pages for details.

New Setup Function for Permanent Magnet Synchronous Machine Block

The `power_PMSynchronousMachineParams` function lets you compute parameters of a Permanent Magnet Synchronous Machine block based on standard manufacturer specifications. It lets you input manufacturer data and returns the computed machine parameters, along with additional derived data such as synchronous speed, number of pole pairs, nominal slip, starting torque, and so on. The function also lets you display relative errors between the input manufacturer data and the equivalent data obtained with the computed parameters. The function comes with a graphical user interface that allows you to compute the block parameters and apply it to selected block, as well as display the detailed results in the Command window.

Additional Solver Type Option Available for Discrete Mode

When you set the **Simulation type** parameter of the Powergui block to `Discrete`, the dialog box now contains a new parameter, **Solver type**, which lets you select between `Tustin` and `Backward Euler` options. `Tustin` is the method used in previous versions. It is still the default, therefore there is no compatibility impact on existing models.

If your model shows numerical oscillations upon simulation in discrete mode, using Backward Euler method to discretize the state-space matrices may help eliminate the oscillations.

Distributed Resources Library Renamed to Renewable Energy

The Distributed Resources block library has been renamed, and is now called Renewable Energy library. All the block names are the same, and there is no compatibility impact on existing models.

New SimPowerSystems Demos

The following demos have been added in Version 5.6:

Demo Name	Description
Mechanical Coupling of Synchronous Generator with Exciter System Using the Simscape Mechanical Rotational Port (power_SM_exciter_SSC)	This is a modified version of the Mechanical Coupling of Synchronous Generator with Exciter System (power_SM_exciter) demo. It illustrates how using the new <code>Mechanical rotational port</code> option for mechanical coupling of an excitation system simplifies the model.
Initializing a 29-Bus, 7-Power Plant Network With the Load Flow Tool of Powergui (power_LFnetwork_29bus)	Illustrates the use of the Load Flow tool of Powergui to initialize a 29-bus 735 kV network with detailed modeling of power plants using hydraulic turbines, speed regulation, excitation systems, and power system stabilizers.

R2011b

Version: 5.5

New Features

Bug Fixes

Compatibility Considerations

SimPowerSystems Software Now Requires Simscape Product

SimPowerSystems software now depends on and requires Simscape software, the foundation for Physical Modeling products. Simscape software includes common Physical Modeling utilities and block libraries. See the [Simscape documentation](#) for details.

SimPowerSystems block libraries are now located under the **Simscape** node in the Library Browser.

Interfacing with Simscape Electrical Domains

Four blocks in the new Interface Elements library allow you to connect a SimPowerSystems electrical connection line to a Physical Networks line connected to Simscape blocks. These blocks have SimPowerSystems ports on one side and Simscape ports on the other, and transfer voltage and current as Physical Networks Across and Through variables, respectively, without energy loss.

Sharing Models Using Simscape Editing Modes

SimPowerSystems software now features a selection of two Simscape editing modes that allow full or restricted editing of models.

- The Restricted mode requires SimPowerSystems product to be installed, but does not require a license. It allows you to change a limited set of model parameters, but not the blocks or connections, in a SimPowerSystems model.
- The Full mode requires SimPowerSystems product to be installed with a license. It allows you to change anything in a SimPowerSystems model.

For more details, see [About the Simscape Editing Mode](#) .

Block Library Links Must Be Resolved

All core SimPowerSystems blocks in your models must now have resolved block library links. You can neither disable nor break these library links. This is a global Simscape requirement. Consult the [Simscape documentation](#) for further details.

This restriction does not apply to the blocks from Application libraries (Electric Drives, Distributed Resources, and FACTS) and the Extra library.

Compatibility Considerations

If you have an existing SimPowerSystems model with disabled or broken links from the blocks in the model to the SimPowerSystems block libraries (other than the Applications and Extra library), you must restore all the broken block library links for your model to be valid.

If you have disabled or broken the SimPowerSystems library link for blocks that you have customized and want to keep these modified blocks in your model, you must move these modified blocks to your own custom library or libraries, then copy the block instances that you need to your model.

You must still restore the block link to its parent library, whether that parent is the SimPowerSystems block library or your own.

Changes to SimPowerSystems Demos

In Version 5.5 (R20011b), demos involving SimPowerSystems plus other Simscape products have been moved to the File Exchange (<http://www.mathworks.com/matlabcentral/fileexchange>). You can retrieve the demo models by following these links:

Demo Name	File Exchange Location
Fuel Cell Vehicle (FCV) Power Train (power_FCV_powertrain)	33309-fuel-cell-vehicle-fcv-power-train
Hybrid Electric Vehicle (HEV) Power Train (power_HEV_powertrain)	33310-hybrid-electric-vehicle-hev-power-train-using-battery-model
Electrically-Driven Hydraulic Motor Pump (power_Hydraulic_Pump)	33313-electrically-driven-hydraulic-motor-pump
Multi-Level Modeling for Rapid Prototyping of Complex Systems (power_HEV_MultiFidelity)	33315-multi-level-modeling-for-rapid-prototyping-of-complex-systems

R2011a

Version: 5.4

New Features

Bug Fixes

New Load Flow Tool

The **Load Flow and Machine Initialization** tool that was available in previous releases is renamed **Machine Initialization**. You can use it to set initial conditions of three-phase machines and regulators in order to start simulation in steady-state. The new **Load Flow** tool now provides an improved load flow solution for SimPowerSystems models, with several new features and capabilities:

- The load flow solution uses the Newton-Raphson method. It is more robust and provides a faster convergence than the **Machine Initialization** tool.
- The new **Load Flow** tool comes with a graphical user interface that allows you to display load flow solution at all buses.
- The Three-Phase Source and Three-Phase Programmable Voltage Source blocks are now taken into account in the load flow. You can now specify power and terminal voltage of the Three-Phase Source and Three-Phase Programmable Voltage Source blocks. Similar to the Synchronous Machine, you can declare these voltage sources as PV, PQ, or swing type. Once the load flow is solved, the source internal voltage magnitude and angle are automatically adjusted.
- You can now specify Three-Phase Series and Parallel RLC Load blocks either as constant impedance (constant Z) or as constant power (constant PQ). In the previous tool, the Three-Phase Series and Parallel RLC Load blocks were only considered as constant impedance loads based on the nominal voltage and active and reactive powers specified in the block menu.

The Load Flow Bus block has been added to the Measurements library. The `power_loadflow` function syntax has been modified, and a new **Load Flow** tab has been added to the Powergui block parameters dialog box, as well as to the dialog boxes of all the blocks considered in the load flow solution. For more information on using the new **Load Flow** tool, see Load Flow Tool.

New Asynchronous Machine Block Setup Function

The `power_AsynchronousMachineParams` function lets you compute parameters of a double-cage Asynchronous Machine block based on standard manufacturer specifications. It lets you input manufacturer data and returns the computed machine parameters, along with additional derived data such as synchronous speed, number of pole pairs, nominal slip, starting torque, and so on. The function also lets you display relative errors between the input manufacturer data and the equivalent data obtained with the computed parameters. The function comes with a graphical user interface that allows

you to compute the block parameters and apply it to selected block, as well as display the detailed results in the Command window.

5-Phase Synchronous Machine Model Available

The Permanent Magnet Synchronous Machine block has been modified to allow modeling a 5-phase synchronous machine with sinusoidal back EMF and round rotor type. For details, see the block reference page.

SimState Support

SimPowerSystems software now supports Simulink SimState feature, introduced in R2009a. This feature allows you to save all runtime data necessary for restoring the simulation state of a model. For more information, see Saving and Restoring the Simulation State as the SimState in the *Simulink User's Guide*.

New SimPowerSystems Demo

The following demo has been added in Version 5.4:

Demo Name	Description
Flickermeter on a Distribution STATCOM (power_flickermeter)	Flickermeter model designed according to functional specifications of the international standard IEC 6100-4-15.

R2010b

Version: 5.3

New Features

Bug Fixes

Double Squirrel-Cage Rotor Option Available in Asynchronous Machine Block

The Asynchronous Machine block now lets you simulate a double squirrel-cage rotor. The **Rotor type** parameter in the block dialog box has a new option, **Double squirrel-cage**, in addition to the existing rotor modeling options of **Wound** and **Squirrel-cage**. For more information, see the block reference page.

Enhanced Code Generation Capabilities

Code generation support has been added for:

- Models in Phasor mode
- Models employing Ideal Switch mode (under Continuous)

For more information, see [Improving Simulation Performance](#).

New SimPowerSystems Demos

The following demos have been added in Version 5.3:

Demo Name	Description
Synchronous Buck Converter (power_switching_power_supply)	Illustrates an abstracted version of a synchronous buck converter that uses ideal switching to give faster simulation times.
AC3 - Sensorless Field-Oriented Control Induction Motor Drive (ac3_sensorless)	Models a sensorless field-oriented control (FOC) induction motor drive with a braking chopper for a 200HP AC motor, using a modified version of the AC3 block.
AC7 - Sensorless Brushless DC Motor Drive During Speed Regulation (ac7_sensorless)	Models a sensorless brushless DC motor drive with a braking chopper for a 3HP motor, using a modified version of the AC7 block.
Electrically-Driven Hydraulic Motor Pump (power_Hydraulic_Pump)	Models a hydraulic pump driven by an electric motor, using SimPowerSystems and SimHydraulics® blocks. Note: This demo has been moved to the File Exchange in Version 5.5 (R20011b).

R2010a

Version: 5.2.1

New Features

Bug Fixes

Permanent Magnet Model Available in DC Machine Block

The DC Machine block now lets you model a permanent magnet DC machine, parameterized either by torque (torque per current constant) or by back-emf (voltage per speed constant). A new parameter, **Field type**, allows you to select between the wound-field and the permanent magnet DC machine. For more information, see the block reference page.

R2009b

Version: 5.2

New Features

Bug Fixes

Compatibility Considerations

Enhanced power_cableparam Function

The power_cableparam function, introduced in Version 5.1 (R2009a) as part of the Computation of R L and C Cable Parameters (power_cable) demo, is now available as a standalone command-line function with associated graphical user interface. It lets you compute RLC parameters of radial copper cables with single screen, based on conductor and insulator characteristics. For more information, see the power_cableparam reference page.

Changes to the Fuel Cell Stack Block

The Fuel Cell Stack block has been improved to better represent the cell dynamics. The model parameters and the meaning of some detailed parameters have changed since the last release, as described in the following section.

Compatibility Considerations

The Fuel Cell Stack block parameters have been changed in Version 5.2 (R2009b). If you used the **(No) User-Defined** option for the **Preset model** parameter in previous releases and defined particular detailed parameters for your Fuel Cell Stack block, the software will automatically convert your old block parameters into new values corresponding to the block changes.

The following table compares the old parameter names to the new ones. It also provides details on how the new values are computed:

Old Parameters	New Parameters and Values
Open circuit voltage	Voltage at 0A and 1A [V_0(V), V_1(V)] = [oldvalue, oldvalue*0.95]

New SimPowerSystems Demos

The following demos have been added in Version 5.2:

Demo Name	Description
Six-Pulse Cycloconverter (power_cycloconverter)	Illustrates a six-pulse cycloconverter driving a static load.

Speed Control of a DC Motor Using BJT H-Bridge (power_Hbridge)	Illustrates simulation of an H-bridge used to generate a chopped voltage and control speed of a DC motor, in open loop, in both directions.
Five-Cell Multi-Level Converter (power_fivecells)	Illustrates a five-cell multilevel converter driving a static load.
Multi-Level Modeling for Rapid Prototyping of Complex Systems (power_HEV_MultiFidelity)	Illustrates how to use different detail level in model simulation. For more information, see Multi-Level Modeling for Rapid Prototyping. Note: This demo has been moved to the File Exchange in Version 5.5 (R20011b).

R2009a

Version: 5.1

New Features

Bug Fixes

Compatibility Considerations

Powergui Tools Are Also Available as Standalone Command-Line Functions

The graphical user interface analysis tools, available in the Powergui block, are now also implemented as standalone command-line functions. Each of these tools can be activated by entering the appropriate command at the MATLAB prompt. For more information, see the following reference pages: `power_fftscope`, `power_hysteresis`, `power_initstates`, `power_lineparam`, `power_loadflow`, `power_ltiview`, `power_report`, `power_steadystate`, `power_zmeter`.

Enhancements to the Ideal Switching Algorithm

The **Display circuit differential equations** option, available in the Powergui block parameters dialog box, lets you display differential equations of the model in the command window when the simulation starts. This option is visible only if **Enable use of ideal switching devices** is selected. For more information, see [Using the Ideal Switching Device Method](#).

Powergui Block No Longer Added Automatically

The Powergui block is no longer automatically added to your model upon simulation. You need to explicitly add it to your model. For more information, see [Using the Powergui Block to Simulate SimPowerSystems Models](#).

Compatibility Considerations

If you have an old model without a Powergui block, which used to run in previous releases because Powergui was added automatically during simulation, you will now get an error trying to simulate it. Add a Powergui block and save the model to avoid the error.

Changes to the Battery Block

The Battery block has been improved to accurately represent the battery dynamics during the charge and the discharge processes. The model parameters and the meaning of some detailed parameters have changed since the last release, as described in the following section.

Compatibility Considerations

The Battery block parameters have been changed in Version 5.1 (R2009a). If you used the **(No) User-Defined** option for the **Battery Type** parameter in previous releases and defined particular detailed parameters for your Battery block, the software will automatically convert your old block parameters into new values corresponding to the block changes.

The following table compares the old parameter names to the new ones. It also provides details on how the new values are computed:

Old Parameters	New Parameters and Values
Battery Type, set to (No) User-Defined	Battery Type, set by default to Nickel-Metal-Hybrid
-	Maximum Capacity (Ah) = $\text{RatedCapacity} * 1.05$
Full charge voltage (%)	Fully charged voltage (v) = $[\text{oldvalue}] / 100 * \text{NominalVoltage}$
Nominal Discharge Current (% of Rated Capacity)	Nominal Discharge Current (A) = $[\text{oldvalue}] / 100 * \text{RatedCapacity}$
Capacity (% of Rated Capacity) @ Nominal Voltage	Capacity (Ah) @ Nominal Voltage = $[\text{oldvalue}] / 100 * \text{RatedCapacity}$
Exponential zone Voltage (%) Exponential zone Capacity (% of the Rated Capacity)	Exponential zone Voltage (V) = $[\text{oldvalue}] / 100 * \text{NominalVoltage}$ Exponential zone Capacity (Ah) = $[\text{oldvalue}] / 100 * \text{RatedCapacity}$

New SimPowerSystems Demos

The following demos have been added in Version 5.1:

Demo Name	Description
Computation of R L and C Cable Parameters (power_cable)	Illustrates use of the <code>power_cableparam</code> demo function to calculate the R,L, and C parameters of two 132kV 2-phase cables with screen cables.
Single-Phase Dynamic Load Block (power_1phdynamicload)	Illustrates an example of a single-phase dynamic load block built with Simulink blocks. This is the recommended template for interfacing custom-built Simulink blocks with SimPowerSystems models.
Zener Diode Regulator (power_zener)	Presents a model of the zener diode used in a voltage regulator.
Full Wave Rectifier (power_FullWaveRectifier)	Illustrates use of the Ideal Switching Device solution method to simulate a full wave rectifier using ideal diodes.
Multilevel Multiphase Space-Vector PWM (power_svpwm_multiPhasesLevel)	Illustrates modeling and operation of the Multilevel Multiphase Space-Vector PWM and Two-Level Multiphase Space-Vector PWM blocks. The demo includes a five-level five-phase inverter feeding a passive load.
Synchronous Generator and Full Scale Converter (Type 4) Detailed Model (power_wind_type_4_det)	Illustrates simulation of a 10 MW wind farm using a detailed model of a Type 4 wind turbine.
Synchronous Generator and Full Scale Converter (Type 4) Average Model (power_wind_type_4_avg)	Illustrates simulation of a 10 MW wind farm using an average model of a Type 4 wind turbine.

R2008b

Version: 5.0

New Features

Bug Fixes

New Ideal Switching Algorithm

New Ideal Switching algorithm, available in the Powergui block, enables faster and more accurate simulation of power electronic devices. For more information, see Using the Ideal Switching Device Method.

Changes to the Universal Bridge Block

The Universal Bridge block has two new options for modeling voltage-sourced converters (VSC):

- Switching-function based VSC
- Average-model based VSC

New SimPowerSystems Demos

The following demos have been added in Version 5.0:

Demo Name	Description
Switching an Inductive Circuit Using a Breaker With no Snubber (power_breaker)	Illustrates the Ideal Switching device solution method of the Powergui block.
Fuel Cell Vehicle (FCV) Power Train (power_FCV_powertrain)	Demonstration of a Fuel Cell Vehicle (FCV) power train using SimPowerSystems and SimDriveline. The FCV power train is of the series type. This FCV is propelled by one electric motor powered by a fuel cell and a battery. Note: This demo has been moved to the File Exchange in Version 5.5 (R20011b).

R2008a

Version: 4.6

New Features

Bug Fixes

Compatibility Considerations

New Fuel Cell Stack Block

A new block, Fuel Cell Stack, has been added to the Extra Sources sublibrary of the Electric Drives library. It implements a generic model parameterized to represent most popular types of fuel cell stacks fed with hydrogen and air. The 6 kW 45 Vdc Fuel Cell Stack demo (`power_fuel_cell`) shows how to use the Fuel Cell Stack block to model a Proton Exchange Membrane (PEM) Fuel Cell Stack feeding an average value 100Vdc DC/DC converter.

The Battery block from the Electrical Sources library is now included in the Extra Sources sublibrary of the Electric Drives library as well.

Initial Conditions Can Be Specified for the Permanent Magnet Synchronous Machine Block

The following enhancements have been implemented for the Permanent Magnet Synchronous Machine block:

- A new parameter, **Initial conditions**, allows you to specify the initial mechanical speed (rad/s), mechanical angle Θ (degrees) and instantaneous stator current (A).
- A new drop-down list lets you select the machine constant that you wish to specify for block parameterization: the flux linkage, the voltage constant, or the torque constant. Once you select a constant, you can enter its value in the appropriate parameter field, while the other two parameters become inaccessible and are only shown for information.
- The dialog box has been rearranged into three tabs, **Configuration**, **Parameters**, and **Advanced**, to improve usability.

Multiple Discretization Rates within a Model Now Available

For certain blocks, you can specify a different sample time than the one specified by the powergui block. This allows you to discretize different parts of a model at different rates in a fixed time step simulation. For example, if one block needs to run at a smaller time step ($t1$) than the rest of the simulation ($t2$), you can speed up simulation of the whole model by specifying a different time step for this block, as long as $t2 = n * t1$ (where n is an integer).

The following is a list of blocks that currently can be discretized at a different rate:

- Asynchronous Machine

- DC Machine
- Permanent Magnet Synchronous Machine
- Simplified Synchronous Machine
- Single Phase Asynchronous Machine
- Stepper Motor
- Switched Reluctance Motor
- Synchronous Machine

Dialog boxes for most of these blocks have also been rearranged into three tabs, **Configuration**, **Parameters**, and **Advanced**, to improve usability.

Compatibility Considerations

The DC Machine block can be discretized now. It is recommended that you use it instead of the Discrete DC Machine block, which will be deprecated in the future.

New SimPowerSystems Demos

The following demos have been added in Version 4.6:

Demo Name	Description
6 kW 45 Vdc Fuel Cell Stack (power_fuel_cell)	Demonstration of the Proton Exchange Membrane (PEM) Fuel Cell Stack model feeding an average value 100Vdc DC/DC converter. The nominal Fuel Cell Stack voltage is 45Vdc and the nominal power is 6kW.
Solid-Oxide Fuel Cell Connected to Three-Phase Electrical Power System (power_SOFC)	This demo illustrates a model of a solid oxide fuel cell (SOFC). The system consists of a SOFC, which is connected to a three-phase infinite bus through an IGBT inverter.
Mechanical Coupling of Synchronous Generator with Exciter System (power_SM_exciter)	In large alternators, the excitation system is provided by a small synchronous machine connected on the same shaft as the main synchronous machine. This demo illustrates interconnecting two machines on the same shaft by use of speed input.

Three-Phase Core-Type Transformer (power_Transfo3phCoreType)	This demo illustrates use of the Three-Phase Transformer Inductance Matrix Type block to model a three-phase core-type saturable transformer. It also demonstrates that using three single-phase transformers to simulate a Yg/Yg core-type transformer is not acceptable.
Three-Phase Matrix Converter (power_three_phase_matrix_converter)	This demo illustrates a three-phase matrix converter driving a static load. Indirect space-vector modulation allows direct control of input current and output voltage and hence allows the power factor of the source to be controlled. As a result, the demo outputs the unity power factor at the source.
Three-Phase Active Harmonic Filter (power_active_filter)	This demo illustrates the use of a shunt active harmonic filter (AHF) to minimize the harmonic content propagated to the source from a nonlinear load.

R2007b

Version: 4.5

New Features

Bug Fixes

Compatibility Considerations

New Battery Block

A new block, Battery, has been added to the Electrical Sources library. It implements a generic battery that models most popular battery types, such as Nickel-Metal-Hybride, Lead-Acid, Lithium-Ion, and Nickel-Cadmium. User-Defined Battery type allows you to modify detailed parameters to represent any particular discharge characteristics.

The Hybrid Electric Vehicle (HEV) Power Train demo (`power_HEV_powertrain`), which was introduced in Version 4.4 (R2007a) and shows a multi-domain simulation of a HEV power train based on SimPowerSystems and SimDriveline blocks, has been modified to use the Battery block. It is now called Hybrid Electric Vehicle (HEV) Power Train Using Battery Model.

New Stepper Motor Block

A new block, Stepper Motor, has been added to the Machines library. Depending on the motor configuration specified by the Motor type parameter, this block models:

- A two- or four-phase permanent magnet or hybrid stepper motor
- A three-, four-, or five-phase variable reluctance stepper motor

Three New Transformer Blocks

Three new transformer blocks have been added to the Elements library:

- Grounding Transformer implements a transformer that is used to provide a neutral in a three-phase, three-wire system. The transformer consists of three two-winding transformers connected in a zigzag. The nominal voltage of each of the six windings is $V_n/3$.
- Three-Phase Transformer Inductance Matrix Type (Two Windings) represents inductive coupling between windings located on different phases of a three-limb or a five-limb core. It also allows modeling of a three-phase transformer built with three single-phase units (no coupling between phases). The transformer R L parameters are obtained from no-load excitation tests and short-circuit tests in positive- and zero-sequence. When core type is specified as **Three-limb or five-limb core**, the transformer is modeled by 9 coupled windings; otherwise, it is modeled by 3 sets of 2 coupled windings ($Z_0=Z_1$).
- Three-Phase Transformer Inductance Matrix Type (Three Windings) represents coupling between windings located on different phases of a three-limb or a five-

limb core. It also allows modeling of a three-phase transformer built with three single-phase units (no coupling between phases). The transformer R L parameters are obtained from no-load excitation tests and short-circuit tests in positive- and zero-sequence. When core type is specified as **Three-limb or five-limb core**, the transformer is modeled by 9 coupled windings; otherwise, it is modeled by 3 sets of 3 coupled windings ($Z_0=Z_1$).

New Measurement Option Available for the PI Section Line Block

A new measurement option, All pi-section voltages and currents, is available for the PI Section Line block. It allows you to measure voltages and currents at the start and end of each pi-section.

New SimPowerSystems Demos

The following demos have been added in Version 4.5:

Demo Name	Description
Ni-MH Battery Model (power_battery)	Demonstration of the battery model during charge and discharge process. The demo models a 200 V, 6.5 Ah Ni-MH battery.
Stepper Motor Drive (power_steppermotor)	Demonstration of a hybrid stepper motor drive. The parameters are those of a small stepper motor (size 23).
D-STATCOM (Average Model) (power_dstatcom_avg)	In the average model of a Distribution STATCOM, the IGBT Voltage-Sourced Converters (VSC) are represented by equivalent voltage sources generating the AC voltage averaged over one cycle of the switching frequency. This model does not represent harmonics, but the dynamics resulting from the control system and power system interaction are preserved. This model allows using much larger time steps (typically 40-50 microseconds), thus allowing simulations of several seconds.
D-STATCOM (Detailed Model) (power_dstatcom_pwm)	The detailed model of a Distribution STATCOM includes detailed representation of power electronic IGBT converters. In order to achieve

	an acceptable accuracy with the 1680 Hz switching frequency used in this demo, the model must be discretized at a relatively small time step (5 microseconds). This model is well suited for observing harmonics and control system dynamic performance over relatively short periods of times (typically hundreds of milliseconds to one second).
UPFC (Detailed Model) (power_upfc_gto48p)	Detailed model of a 48-pulse, GTO-based Unified Power Flow Controller (500 kV, 100 MVA).

Renamed `psbhysteresis` Command

In Version 4.5 (R2007b), `power_hysteresis` is the new name for the old `psbhysteresis` command. You use it exactly the same way you would use the `psbhysteresis` command.

Compatibility Considerations

Currently, if you issue the `psbhysteresis` command, it will automatically redirect to its new name, `power_hysteresis`. However, it is recommended that you update your scripts and use the new command name going forward.

R2007a

Version: 4.4

New Features

Bug Fixes

Compatibility Considerations

New Brushless DC Motor Drive Block

A new block, Brushless DC Motor Drive, has been added to the Electric Drives/AC Drives library. It implements a brushless DC motor drive using a Permanent Magnet Synchronous Motor (PMSM) with trapezoidal back electromotive force (BEMF). It is possible to use a simplified version of the drive containing an average-value model of the inverter for faster simulation. In SimPowerSystems software, the Brushless DC Motor Drive block is commonly called the AC7 motor drive.

Automated Conversion of Version 2 Models Is No Longer Supported

The automated conversion of old models, created with blocks from SimPowerSystems 2.3 or Power System Blockset™ 2 libraries, is no longer supported in Version 4.4 (R2007a).

Compatibility Considerations

The `psbupdate` function is obsolete as of Version 4.4 (R2007a).

New SimPowerSystems Demos

The following demos have been added in Version 4.4:

Demo Name	Description
Hybrid Electric Vehicle (HEV) Power Train (power_HEV_powertrain)	<p>Multi-domain simulation of a HEV power train based on SimPowerSystems and SimDriveline blocks. The HEV power train is of the series-parallel type, such as the one found in the Toyota Prius car. This HEV has two kinds of motive power sources, an electric motor and an internal combustion engine (ICE), in order to increase the drive train efficiency and reduce air pollution. It combines the advantages of the electric motor drive (no pollution and high available power at low speed) and the advantages of an internal combustion engine (high dynamic performances and low pollution at high speeds).</p> <p>Note: This demo has been moved to the File Exchange in Version 5.5 (R20011b).</p>

Aircraft Electrical Power Generation and Distribution (power_aircraft_distribution)	This circuit illustrates a generic aircraft Electrical Power Generation & Distribution System. The AC power frequency is variable and depends of the engine speed.
--	--

R2006b

Version: 4.3

New Features

Bug Fixes

Mechanical Input Parameter Lets You Connect SimMechanics or SimDriveline Blocks to Electric Drives by Specifying Motor Speed as Block Input

The AC and DC Electric Drive blocks have a new parameter called **Mechanical input**, which lets you specify either the load torque or the motor speed as block input.

As of V4.3 (R2006b), mechanical input is available for the following blocks:

- Six-Step VSI Induction Motor Drive (AC1)
- Space Vector PWM VSI Induction Motor Drive (AC2)
- Field-Oriented Control Induction Motor Drive (AC3)
- DTC Induction Motor Drive (AC4)
- Self-Controlled Synchronous Motor Drive (AC5)
- PM Synchronous Motor Drive (AC6)
- Two-Quadrant Single-Phase Rectifier DC Drive (DC1)
- Four-Quadrant Single-Phase Rectifier DC Drive (DC2)
- Two-Quadrant Three-Phase Rectifier DC Drive (DC3)
- Four-Quadrant Three-Phase Rectifier DC Drive (DC4)
- One-Quadrant Chopper DC Drive (DC5)
- Two-Quadrant Chopper DC Drive (DC6)
- Four-Quadrant Chopper DC Drive (DC7)

To switch to the motor speed as mechanical input, open the block dialog box and set the **Mechanical input** option in the bottom portion of the dialog box to **Speed**. Note that if you select the motor speed as mechanical input, the internal mechanical system is not used and the inertia and viscous friction parameters are not displayed. You have to include these parameters in the external mechanical system.

R2006a

Version: 4.2

New Features

Bug Fixes

Average Values of electricdrivelib Blocks

Two more AC drives blocks, AC3 and AC5, in the `electricdrivelib` library now have a new parameter that allows you to specify average-value models, as opposed to detailed models, for the converters. This parameter was first introduced for some of the `electricdrivelib` library blocks in V4.1 (R14SP2+).

As of V4.2 (R2006a), average-value models are available for the following blocks:

- Space Vector PWM VSI Induction Motor Drive (AC2)
- Field-Oriented Control Induction Motor Drive (AC3)
- Self-Controlled Synchronous Motor Drive (AC5)
- PM Synchronous Motor Drive (AC6)
- Two-Quadrant Single-Phase Rectifier DC Drive (DC1)
- Four-Quadrant Single-Phase Rectifier DC Drive (DC2)
- Two-Quadrant Three-Phase Rectifier DC Drive (DC3)
- Four-Quadrant Three-Phase Rectifier DC Drive (DC4)
- One-Quadrant Chopper DC Drive (DC5)
- Two-Quadrant Chopper DC Drive (DC6)
- Four-Quadrant Chopper DC Drive (DC7)

To switch to the average-value representation, open the block dialog box and set the **Model detail level** option in the bottom portion of the dialog box to **Average**.

Transformer Blocks with SI Units Are Available

The Transformer blocks now have a parameter named **Units**, which allows you to specify the **SI units** or the **pu units**. In addition, this parameter can be used to automatically convert pu units into SI units, or the reverse.

Open Circuit Option Is Added for the RLC Blocks

The RLC branch blocks now have an extra option under the **Branch Type** parameter that allows you to specify an **Open Circuit** branch. This is particularly useful if you want to temporarily get rid of an RLC element in the circuit without deleting the block.

New Demos and Enhancements to Existing Demos

The following demos have been added.

Demo Name	Description
power_tcsc_phasor.mdl power_tcsc.mdl	Thyristor Controlled Series Capacitor (TCSC) test systems from Dr. Dragan Jovicic from University of Aberdeen, UK
power_asm1ph_auxcontrol power_asm1ph_vectorcontrol	The single-phase Asynchronous Machine block using the Main & auxiliary windings configuration

In the Wind Farm DFIG demos of the Distributed Resources library, control systems have been enhanced.

R14SP3

Version: 4.1.1

Bug Fixes

R14SP2+

Version: 4.1

New Features

Bug Fixes

Compatibility Considerations

New Blocks in the Machines Library of powerlib

The machines library of powerlib contains a model of a Switched Reluctance Motor that allow you to model three typical configurations of such a motor. The library also contains a model of a Single-Phase Asynchronous Machine that can be configured as a split phase, a capacitor-start, or as a capacitor-start-run motor mode.

Enhancements to Existing Blocks of the Machines Library

The core saturation can now be specified for the Asynchronous Machine block when the block is used in a phasor simulation. The saturation model of the Asynchronous Machine is based on the fundamental component of the current and does not include the third harmonic. The saturation parameter is available only when the simulation is in phasor mode. When the powergui block is set to continuous or discrete mode, the parameter is disabled in the mask of the block.

The Permanent Magnet Synchronous Machine block allows you to specify a trapezoidal flux distribution as an alternative to the sinusoidal flux option of the previous version of the block.

Branch Type Parameter of the RLC Branch Blocks

The Series RLC Branch block, Parallel RLC Branch block, Three-Phase Series RLC Branch block, and Three-Phase Parallel RLC Branch block now have a new parameter that allows you to directly specify the elements that are present in the branch: the R, L, C, RL, LC, RC, and RLC configurations can be specified. It is no longer required to specify an Inf value for the Capacitance in a Series RLC Branch block to get rid of the capacitor device in the branch or to specify 0 value of resistance to get rid of the resistor of a Parallel RLC Branch block.

Average Values of electricdrivelib Blocks

The seven DC drives blocks, and the AC2 and AC6 AC drives of the `electricdrivelib` library now have a new parameter that allows you to specify average value models, as opposed to detailed models, for the converters.

To switch to the average-value representation, open the block dialog box and set the **Model detail level** option in the bottom portion of the dialog box to **Average**.

Obsolete Blocks

The Discrete System block and the Machine Measurement Demux block are no longer supported in V4.1 (R14SP2+).

Compatibility Considerations

The table below indicates blocks that are obsolete as of the current version, and lists blocks that you can use as replacement for the obsolete blocks.

Obsolete Block	Removed from Version	Replacement
Discrete System	4.1	Powergui
Machine Measurement Demux	4.1	Bus Selector

R14SP2

Version: 4.0.1

Bug Fixes

