

Simscape™ Electrical™ 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.
1 Apple Hill Drive
Natick, MA 01760-2098

Simscape™ Electrical™ Release Notes

© COPYRIGHT 2018- 2022 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.

Simscape Electrical	1-2
N-Channel MOSFET and P-Channel MOSFET Enhancement: Model faults with different failure modes and triggers	1-2
PMSM (Four-Phase) Block: Model a four-phase permanent synchronous motor	1-2
Electrolyzer Block: Model the electric load of an electrolyzer	1-2
Harmonic Analyzer App: Analyze and plot the harmonic spectrum of Simscape logging nodes	1-2
Average-Value Voltage Source Converter (Three-Phase) and Average-Value Rectifier (Three-Phase) Block Enhancement: Specify AC frequency to simulate block in frequency-and-time mode	1-3
Battery (Table-Based) Block Enhancement: Model capacity calendar aging	1-3
Battery (Table-Based) Block Enhancements: Expose state-of-charge and thermal ports in the block mask	1-3
ee.spice.semiconductorSubcircuit2lookup Function: Generate table lookup data for semiconductors from a SPICE subcircuit	1-3
Controlled Current Source (Three-Phase) and Controlled Voltage Source (Three-Phase) Block Enhancement: Specify input type to simulate block in frequency-and-time mode	1-3
Busbar (DC) Block: Model a DC connector for load-flow analysis	1-3
Three-Phase Transformer Enhancements: Model three-phase transformers with ideal options	1-4
Simplified Synchronous Machine Block Enhancement: Initialize the machine from terminal voltage and power requirements or from load flow analysis	1-4
FEM-Parameterized PMSM Block Enhancement: Tabulate iron losses from current and speed	1-4
FEM-Parameterized Linear Actuator and FEM-Parameterized Rotary Actuator Block Enhancements: Model iron losses using magnetizing resistance	1-4
Switched Reluctance Machine (Multi-Phase) Block Enhancement: Model a six-phase switched reluctance machine	1-4
Switched Reluctance Machine and Switched Reluctance Machine (Multi-Phase) Block Enhancements: Tabulate iron losses from current, speed, and rotor angle	1-5
Synchronous Reluctance Machine Block Enhancement: Model iron losses	1-5
Motor & Drive (System Level) Block Enhancement: Plot an efficiency map and tabulate the torque envelope from speed and supply voltage	1-5
Bidirectional DC-DC Converter Enhancement: Model a dual active bridge converter with eight switching devices	1-5
DC-DC Converter Enhancement: Tabulate efficiency as a function of input voltage	1-5

Eight-Pulse Gate Multiplexer Block: Multiplex gate input signals to a Bidirectional DC-DC Converter block	1-6
Single-Phase Half-Bridge Inverter Examples: Simulate inverters with an optional thermal port	1-6
Featured Examples	1-6
Functionality being removed or changed	1-7
Specialized Power Systems	1-8
Power Converters Blocks Enhancement: Implement average model converters without simulating rectifier mode	1-8
Multi-Winding Transformer Block Enhancement: Specify initial flux value	1-8
Featured Examples	1-8

R2021b

Simscape Electrical	2-2
Circuit Simulation Onramp: Self-paced, interactive tutorial included with Simscape Electrical license	2-2
BLDC Block: Model a specific brushless DC motor from multiple built-in parameterizations	2-2
Diode Block: Use reverse recovery energy to model charge dynamics	2-3
Machine Blocks: Simulate the effects of generated heat and device temperature by exposing the thermal port	2-3
Solar Cell Block: Model a specific solar cell from multiple built-in parameterizations	2-3
Transmission Line (Three-Phase) Block: Model faults	2-4
Delta-Connected Variable Load Block: Model a three-phase variable load wired in delta configuration	2-4
Diode Block Enhancement: Tabulate voltage as a function of temperature and current	2-4
GTO and Thyristor (Piecewise Linear) Block Enhancements: Measure the current at a specific time after reaching the current threshold	2-4
Half-Bridge (Ideal, Switching) Block: Model a half-bridge with ideal switches and thermal port	2-5
Heatsink Block: Dissipate heat from power semiconductors to ambient temperature	2-5
Cauer Thermal Model Block: Simulate heat transfer through multiple layers of a semiconductor module	2-5
Foster Thermal Model Block: Initialize block in steady state or specify initial conditions	2-5
Thermal Network for Semiconductor Devices Enhancement: Simulate the heat generated by switching events and conduction losses with a Cauer model with N elements	2-5
Stepper Motor Driver Block: Simulate microstepping mode to increase the resolution of the position of the rotor	2-6
Solar Cell Block: Specify parallel-connected strings in series-connected cells	2-6
Induction Machine Enhancements: Switch between induction machine types in block mask	2-6

Composite or Expanded Ports Enhancement: Switch between composite and expanded ports in block mask	2-6
ee_convertedMosfetValidation Function: Compare MOSFET conversions results with SIMetrix	2-7
subcircuit2ssc Function: Support smoothing functions	2-7
ee_importDeviceParameters Function: Import device parameters from an XML file to a Simscape block	2-7
SM DC4C Block: Model a discrete-time or continuous-time synchronous machine DC4C excitation system	2-7
Moving Average and Fourier Analysis Block Enhancements: Specify buffer size	2-7
PWM Generator Block: Specify phase delay	2-8
Featured Examples	2-8

Specialized Power Systems 2-9

Three-Phase Tap-Changing Transformer (Two-Windings) and Three-Phase Tap-Changing Transformer (Three-Windings) Blocks: Model two-winding and three-winding three-phase tap changing transformers	2-9
Three-Phase Autotransformer with Tertiary Winding: Model a three-phase autotransformer	2-9
Impedance Measurement Tool App: Compute impedance versus frequency of Impedance Measurement blocks in a model	2-9
Linear System Analyzer App: Create a state-space object of the linear part of an electrical circuit	2-9
Transformer Hysteresis Design Tool App: View and edit hysteresis characteristic	2-9
Report Generator App: Generate report of steady-state variables, initial states, and machine load flow for model	2-9
Statistical Flicker Level Analyzer App: Perform statistical analysis of instantaneous flicker level signal	2-10
Featured Examples	2-10

R2021a

Simscape Electrical 3-2

ee_comtradeDataset Function: Read and extract values from a COMTRADE record	3-2
N-Channel MOSFET and P-Channel MOSFET Block Enhancements: Parameterize MOSFET by a 2-D or 3-D lookup table approximation to the I-V curve	3-2
N-Channel MOSFET and P-Channel MOSFET Block Enhancements: Tabulate body diode to the I-V curve	3-2
ee_getBillOfMaterials Function: Generate report for model of current system and all subsystems	3-2
SPICE Conversion Assistant: Model parasitic values in capacitors and inductors	3-2
Diode Block Enhancement: Model faults and trigger mechanisms	3-2
ee_getPowerLossSummary and ee_getPowerLossTimeSeries Function Enhancements: Calculate switching losses for semiconductor blocks in a model	3-3

SPICE-Imported MOSFET Block Enhancement: Model additional pre-parameterized MOSFET Devices	3-3
Ideal Switching Device Block Enhancements: Improve switching loss implementation	3-3
Piezo Bender Block: Model a piezoelectric bimorph beam of rectangular cross-section	3-3
Multiple Machine Block Enhancement: Hide neutral phase port when zero sequence is excluded	3-4
Fuel Cell Block: Model a fuel cell electrical system with different levels of details	3-4
FEM-Parameterized PMSM Block Enhancement: Additional iron losses modeling and optional neutral port for 3-D flux linkage variant	3-4
DC-DC Converter Block Enhancement: Tabulate efficiencies and calculate converter losses as a function of output current and temperature	3-5
Compound Motor Block: Model a compound motor with electrical and torque characteristics and fault modeling	3-5
Battery and Battery (Table-Based) Block Enhancements: Model battery degradation due to calendar aging	3-5
Expanded Mathematical Transforms Library: Transform five-phase or six-phase components to components in a rotating reference frame and vice versa	3-5
Expanded Pulse Width Modulation Library: Generate gating times and controlling pulses for five-phase power converter control	3-6
Winding Block Enhancement: Model temporal and behavioral faults	3-6
Rotating Air Gap Block: Model an air gap between a stator tooth and a rotating permanent magnet rotor	3-6
Average-Value Voltage Source Converter Block Enhancement: Parameterize converter losses and expose thermal port	3-6
Constant Current Load (Three-Phase) Block: Model a faultable constant current load for a three-phase supply	3-7
Constant Current Load Block: Model a faultable constant current load for a DC or AC supply	3-7
PMSM Block Enhancement: Model a specific permanent magnet synchronous motor from multiple built-in parameterizations	3-7
Converters Block Enhancements: Model inductor series resistance	3-7
Three-Phase Motor Block Enhancements: Model thermal ports	3-7
Featured Examples	3-8
Specialized Power Systems	3-9
Library Restructure: Find critical Specialized Power Systems blocks more easily	3-9
Asynchronous Machine Blocks Enhancement: Specify the turn ratio	3-9
PV Array Block Enhancement: Updated default value for modules parameter	3-9
Measurements and States Analyzer App: Compute initial and steady-state values of voltages and currents of model	3-9
FFT Analyzer App: Perform Fourier analysis of simulation data signals	3-9
Load Flow Analyzer App: Perform positive-sequence load flow or unbalanced load flow and initialize models containing load flow blocks	3-9
Featured Examples	3-10

Simscape Electrical	4-2
Modular Multilevel Converter Blocks: Model modular multilevel converters as a number of series-connected power submodules	4-2
Battery and Battery (Table-Based) Block Enhancements: Model the dependency of the terminal resistance on current directionality	4-2
PMSM (Six-Phase) Block: Model a six-phase permanent synchronous motor	4-2
Synchronous Machine (Six-Phase) Block: Model a six-phase synchronous machine	4-2
Ideal Switching Device Block Enhancements: Improve switching loss implementation and thermal modeling options	4-2
Diode Block Enhancement: Model a tabulated I-V diode	4-3
SPICE-Imported MOSFET Block: Model a predefined MOSFET parameterized by an external SPICE subcircuit	4-3
Battery (Table-Based) Block Enhancement: Model a specific lithium-ion battery from multiple built-in parameterizations	4-3
ee_convertedmosfetvalidation Function: Validate MOSFET conversions by generating characteristics and comparing results with LTspice	4-3
SPICE Conversion Assistant: Use table SPICE function	4-4
SPICE NMOS and SPICE PMOS Block Enhancements: Model the transistor capacitances with Meyer gate or charge conservation models	4-4
Current-Controlled Switch and Voltage-Controlled Switch Blocks Enhancements: Model a variable-resistance or a short-transition switch	4-4
Motor & Drive (System Level) Block: Represent a generic motor and drive with closed-loop torque control	4-4
DC Motor Block Enhancement: Model armature winding or field winding faults	4-4
Unipolar Stepper Motor Block: Model mutual inductance	4-5
Simplified Synchronous Machine Block: Model a simplified synchronous machine with electromotive force (EMF)	4-5
Simplified Synchronous Machine Measurement Block: Output a per-unit measurement associated with a connected Simplified Synchronous Machine block	4-5
Delta-Wound Machines: Support zero sequence dynamics	4-5
Power Sensor (Three-Phase) Block: Model an ideal sensor for active and reactive power measurement	4-5
Load-Flow Analyzer App Enhancements: Perform a load-flow analysis through an improved toolstrip and model highlighting	4-5
Dynamic Load Block: Model a faultable dynamic load for a DC or AC supply	4-6
Dynamic Load (Three-Phase) Block Enhancement: Model a faultable balanced three-phase dynamic load for a three-phase supply	4-6
Constant Power Load Block Enhancement: Model a faultable constant power load for an AC supply	4-6
Constant Power Load (Three-Phase) Block: Model a faultable constant power load for a three-phase supply	4-6
Delta-Connected Load, Wye-Connected Load, and RLC (Three-Phase) Block Enhancements: Model initial conditions in the Variables tab	4-6
Expanded Support for Power Dissipation Data: View power_dissipated simulation logging data for all three-phase transformer blocks	4-7

PWM Generator (Multilevel) Block: Generate pulse width modulated signal or waveform for modular multilevel converters	4-7
Expanded Pulse Width Modulation Library: Generate gating times and controlling pulses for power converter control	4-7
Featured Examples	4-7
Specialized Power Systems	4-9
Distributed Parameters Line (Frequency-Dependent) Block: Model a multiphase distributed parameter line with high-fidelity frequency-dependent parameters	4-9
Decoupling Line Block: Model a single-phase distributed parameters line	4-9
Decoupling Line (Three-Phase) Block: Model a three-phase distributed parameters line	4-9
Power Line Parameters App: Compute RLC parameters of overhead transmission lines from conductor characteristics and tower geometry	4-9
Specify Decoupling Lines App: Replace selected Distributed Parameters Line blocks with Decoupling Line blocks	4-9
PI Section Line Block Enhancement: Implement multi-phase transmission line with lumped parameters	4-9
Featured Examples	4-10

R2020a

Simscape Electrical	5-2
Converters and Choppers Blocks: Select averaged switch option for switching device type	5-2
Load-Flow Analyzer App: Generate, visualize, sort, tune, and export steady-state voltage and power data for three-phase AC power transmission systems	5-2
PMSM and BLDC Blocks Enhancements: Model thermal variants with empirical losses	5-3
SPICE NIGBT block: Model a SPICE-compatible N-Channel insulated gate bipolar transistor	5-3
Expanded SM Control Library: Speed up modeling by using prebuilt and documented automatic voltage regulators and power system stabilizers blocks	5-3
Stepper Motor Block: Select a predefined parameterization	5-3
DC Motor Block Enhancement: Generate magnetic fields with new motor field option	5-4
PMSM (Five-Phase) Block: Model a five-phase permanent synchronous motor	5-4
PMLSM Block: Model a three-phase permanent magnet linear synchronous motor	5-4
Electromechanical Library Enhancement: Model some machines with different fidelity levels	5-4
Induction Machine Squirrel Cage and Induction Machine Wound Rotor Blocks Enhancements: Update Saturation Characteristic	5-4
Inductive Rotor Position Sensor Block: Obtain the rotor position	5-5

Power Transformers Blocks Enhancements: Represent magnetic saturation	5-5
Distributed Parameter Line Model for Transmission Line Block: Obtain more accurate simulations at a particular frequency	5-5
SPICE Diode Block Enhancements: Calculate high-injection knee current, IKF, and recombination current, ISR	5-5
N-Channel MOSFET and P-Channel MOSFET Block Enhancements: Parameterize four terminals	5-6
SPMT Switch Block: Model a single-pole multiple-throw switch	5-6
Average-Value Chopper and Average-Value DC-DC Converter Blocks Enhancement: Model conduction loss dependencies by using tabulated efficiency parameterization	5-6
Average-Value DC-DC Converter Block Enhancement: Use voltage and current reference as control inputs	5-6
DC-DC Converter Block Enhancement: Model the voltage reference externally	5-6
Battery (Table-Based) Block Enhancement: Select additional extrapolation and interpolation options for table-based parameters	5-7
Turbine-Governors Blocks: Implement speed-governing steam or hydropower turbine models and load controllers	5-7
Phasor Measurement Unit (PMU, Three-Phase) Block: Measure the positive sequence component of the frequency, phase, and amplitude of the input signal	5-7
Programmable Signal Generator block: Generate a three-phase sinusoidal signal with programmable magnitude, phase, and frequency over time	5-7
Moving Average (Variable Frequency) Block: Output the moving average value of an input signal of variable frequency	5-7
Sine Wave Generator (Three-Phase) Block: Generate a balanced three-phase sine wave	5-8
Stair Generator Block: Generate a stair pattern signal based on specified values	5-8
Constant Power Load Block: Model a faultable constant power load for a DC supply	5-8
General Control Blocks Enhancement: Specify initialization options	5-8
SPICE Conversion Assistant: Use derivative SPICE function	5-8
SPICE Current Sources Blocks: Specify behavior of parallel conductance	5-8
P-Channel MOSFET Block: Changes on order and location of s and d ports	5-9
Featured Examples	5-9
Specialized Power Systems	5-11
Powergui Block: Use the discrete phasor solver type to discretize and solve electrical networks in phasor mode	5-11
PV Array Block: Use the robust discrete model to discretize PV Array blocks	5-11
Power_flicker Function: Analyze the short-term and long-term flicker severity of an instantaneous flicker level signal shorter than 10 minutes	5-11
Featured Examples	5-11

Simscape Electrical	6-2
Load-Flow Analysis: Determine nodal voltages and real and reactive power flows in Simscape Electrical networks	6-2
Isothermal and Fade Models for Battery Blocks: Specify fade data using temperature-dependent or temperature-independent parameterization	6-2
Thermal Model Improvements for Semiconductors: Model external thermal dynamics using Cauer and Foster representations	6-2
Expanded Support for Power Dissipation Data: View power_dissipated simulation logging data for all machine blocks	6-3
Average-Value PWM for Three-Phase Converter Block: Use averaged firing pulses for PWM-controlled power conversion	6-3
Japanese Language Localization Support: Block parameters, variables, and error messages are available in Japanese	6-3
Expanded Control Library: Speed up modeling by using prebuilt and documented algorithm components	6-3
Hall-Effect Rotary Encoder Block: Convert Hall-effects into voltage and phase angle or angular position	6-3
Passive Harmonic Filter (Three-Phase) Block: Improve power quality	6-4
Sequence Analyzer Block: Calculate magnitude and phase of the sequence of a three-phase signal	6-4
Single-Phase and Three-Phase Fault Block Improvements: Limit fault duration time and specify both event-based and temporal faults for both fault blocks	6-4
SPST and SPDT Relay Blocks: Model a controlled single-pole single-throw or double-throw relay with faults and delays	6-4
Nonlinear Reluctance with Saturation: Include different levels of linearity and saturation in magnetic reluctance	6-5
Three-Winding Nonlinear Transformer: Model varying levels of nonlinearity for a transformer with two secondary windings	6-5
Zigzag-Delta-Wye Transformer block: Model a linear, nonideal zigzag-delta-wye transformer with a zigzag interconnect port	6-5
Delta-Wound PMSM and BLDC Motor Models: Increase motor speed range by applying the full voltage across all three phases	6-6
Dielectric Relaxation Model for Capacitor Block: Model both resistive and dielectric losses	6-6
H-Bridge Block Voltage Supply Ports: Model external DC power for a PWM-controlled H-bridge motor drive	6-6
Harmonic Generation Time Limits for Voltage Source Blocks: Set start and stop times for generating source harmonics in the single and three-phase programmable voltage source blocks	6-6
Solenoid Block Orientation: Set the direction of force between the solenoid plunger and case	6-6
Voltage Plots for CMOS Logic Blocks: Visualize voltage data for integrated circuit models	6-7
Run-Time Tolerance Parameters: Modify fractional and random tolerance parameters for capacitance, inductance, and resistance without recompiling	6-7
Block Name Changes and Mergers: Easily identify and configure modeling components	6-7
Featured Examples	6-9

Specialized Power Systems	6-11
Expanded Automatic Advanced Settings Handling: Simplify discrete simulation settings for machine models by using the powergui block discrete solver	6-11
Discrete Load and Transformer Blocks: Build models for discrete-time simulation using nonlinear and variable blocks	6-11
Pulse-Averaging and Interpolating PWM Blocks: Use advanced pulse generation techniques for high-precision motor control	6-11
Featured Example	6-12

R2019a

Simscape Electrical	7-2
Dynamic Load (Three-Phase) Block: Model a four-quadrant dynamic load	7-2
Controlled Voltage Source (Three-Phase) Block: Apply an ideal controlled three-phase voltage to your network	7-2
Current and Voltage Sensor (Three-Phase) Block: Measure current and voltage at three-phase electrical nodes	7-2
PS Output Port for Fuse and Relay Blocks: Output block state as physical signal	7-2
Pressure Transducer Block Measurement Port: Select domain type for pressure measurement	7-2
Earthing Transformer Block: Create a neutral point for your three-phase power network	7-3
Frequency-Dependent Overhead Line (Three-Phase) Block: Model a three-phase line over a wide range of frequencies	7-3
Programmable Voltage Source Blocks: Provide a single-phase or three-phase voltage with programmable magnitude, frequency, and phase shift to your network	7-3
Restructured Libraries: Easily discover blocks for building networks that have the right mix of performance and fidelity for your model-based design goals	7-4
Block Name Changes and Mergers: Updated block names and merged blocks allow you to easily identify and configure model components ...	7-5
Function Name Changes: Functions renamed to ee_functionName	7-10
Example Model Name Changes: Example models renamed to ee_modelName	7-11
Three-Phase Electrical Domain Changes: The three-phase electrical domain is renamed and moved to the Simscape foundation package	7-11
Featured Examples	7-12
Specialized Power Systems	7-13
Robust Discrete Solver: New powergui option simplifies advanced simulation settings	7-13
CCCV Battery Charger Block: Model a constant-current constant-voltage battery charger device	7-13
Autotuning Option for Electric Drive Blocks: AC3, AC4, AC6, and AC7 electric drive blocks can calculate PID control parameters	7-13

Sensorless Speed Estimation for AC Drive Blocks: AC3 and AC4 electric drive blocks can estimate rotational velocity	7-13
Independent Simulation Type Selection: powergui solver settings are not inherited from Simulink automatic solver settings	7-14

R2018b

Simscape Electronics and Simscape Power Systems combined into Simscape Electrical	8-2
Electronics and Mechatronics	8-3
SPICE Conversion Assistant: Convert SPICE models to Simscape components	8-3
FEM-Parameterized PMSM Flux Parameterization: Specify flux linkage in polar or Cartesian coordinates	8-3
Servomotor Block Loss Parameterization: Tabulate losses and efficiency as a function of speed, torque, and DC supply voltage	8-3
Battery Characteristics Visualization: Plot voltage-charge characteristics for battery model parameter values	8-3
Peltier Device Block: Model conversion between electrical and thermal energy	8-3
elec_getNodeDvDtSummary and elec_getNodeDvDtTimeSeries Functions: Calculate derivatives of terminal voltages with respect to time	8-3
Featured Examples	8-4
Power Systems	8-5
Single-Phase Permanent Magnet Synchronous Motor Block: Model a single-phase PMSM with a squirrel-cage rotor	8-5
Multiphase Switched Reluctance Machines: Model low-current four-phase and five-phase SRMs	8-5
One-Quadrant Chopper Block: Convert from fixed to variable DC voltage bidirectionally	8-5
Average-Value Converters: Model DC-DC semiconductor converters that are suitable for real-time simulation	8-5
Run-Time Parameter Support for All Machine Blocks: Speed up simulation tasks and modify component parameter values without regenerating C code	8-6
Single-Phase Asynchronous Machine Blocks: Model a single-phase ASM using fundamental or fundamental SI parameterization	8-6
Asynchronous Machine Saturation: Include magnetic saturation in three-phase ASMs using open circuit lookup tables	8-6
Double-Squirrel Cage Asynchronous Machines: Specify a single or double cage for ASM squirrel cage blocks	8-7
Synchronous Machine Rotor Angle: Define the rotor axis alignment for synchronous machine blocks	8-7
Synchronous Machine Block Accuracy Improvement: SM blocks that use standard parameters and the SM field circuit block return more accurate results	8-7
Machine Plotting and Display Options: Perform plotting and display actions using the Electrical menu	8-8

Controlled Current Source (Three-Phase) Block: Control the current loop of a cascading control system	8-8
Expanded Control Library: Speed up modeling by using prebuilt and documented algorithm components	8-8
Battery Characteristics Visualization: Plot voltage-charge characteristics for battery model parameter values	8-9
Block Name Changes: Blocks names disambiguated from identically named blocks in the Simscape Electrical Electronics and Mechatronics library	8-9
HDL Code Generation from Simscape Electrical Power Systems Models: Convert models to HDL code for simulation on FPGA devices	8-10
Featured Examples	8-10
Specialized Power Systems	8-11
SVPWM Generator (3-Level) Block: Generate pulses for three-phase three-level neutral-point-clamped converters	8-11
Featured Example	8-11

R2022a

Version: 7.7

New Features

Bug Fixes

Version History

Simscape Electrical

N-Channel MOSFET and P-Channel MOSFET Enhancement: Model faults with different failure modes and triggers

You can now model faults in the N-Channel MOSFET and P-Channel MOSFET blocks. The blocks model five types of faults:

- `Open circuit` — Failure due to metallization burnout
- `Drain-source short` — Failure due to avalanche breakdown on drain-source channel
- `Drain-bulk short or source-bulk short` — Failure due to avalanche breakdown on drain-bulk or source-bulk channels
- `Gate oxide short` — Failure of the gate oxide dielectric layer
- `Parameter shift` — Failure due to aging

The blocks can trigger fault events at a specific time or when the current or voltage exceed the limit for longer than a specific time interval.

PMSM (Four-Phase) Block: Model a four-phase permanent synchronous motor

You can use the PMSM (Four-Phase) block to model a four-phase permanent magnet synchronous motor with sinusoidal flux distribution.

Electrolyzer Block: Model the electric load of an electrolyzer

The Electrolyzer block represents the electric load associated with an electrolyzer. An electrolyzer consists of an anode and a cathode separated by an electrolyte. The block calculates the amount of produced hydrogen based on the electrical energy provided and the water temperature in the tank.

Harmonic Analyzer App: Analyze and plot the harmonic spectrum of Simscape logging nodes

The **Harmonic Analyzer** app calculates the harmonic orders, magnitudes, and fundamental frequency of a `simscape.logging.Node` of an AC or periodic variable and plots a bar chart of the percentage of fundamental magnitude versus harmonic order.

Use the **Harmonic Analyzer** app to:

- Import the simulation data of a model stored in a `simscape.logging.Node` variable.
- Analyze and plot the harmonic spectrum of the selected `simscape.logging.Node` object.
- Export the data in MATLAB® as a script or a function.

To open the app, at the MATLAB Command Window, enter: `ee_harmonicAnalyzer`.

Average-Value Voltage Source Converter (Three-Phase) and Average-Value Rectifier (Three-Phase) Block Enhancement: Specify AC frequency to simulate block in frequency-and-time mode

The Average-Value Voltage Source Converter (Three-Phase) and Average-Value Rectifier (Three-Phase) blocks now work in both time and frequency-and-time simulation modes when you set the **AC frequency** parameter to **Constant**.

Battery (Table-Based) Block Enhancement: Model capacity calendar aging

The Battery (Table-Based) block now allows you to model calendar aging for the capacity. To enable capacity calendar aging, in the **Calendar aging** settings, set **Capacity calendar aging** to **Enabled**.

Battery (Table-Based) Block Enhancements: Expose state-of-charge and thermal ports in the block mask

Prior to R2022a, to expose the **SOC** and **H** ports in the Battery (Table-Based) block, you had to right-click the block in your model and, from the context menu, select **Simscape > Block choices** and select one of the variants.

You can now expose the **SOC** and **H** ports by setting the **Expose SOC measurement port** parameter to **Yes** and the **Thermal port** parameter to **Model**, respectively.

ee.spice.semiconductorSubcircuit2lookup Function: Generate table lookup data for semiconductors from a SPICE subcircuit

Use the `semiconductorSubcircuit2lookup` function to generate table lookup data such as current-voltage characteristics, capacitance characteristics, and diode characteristics for an N-Channel MOSFET, a P-Channel MOSFET, or an N-Channel IGBT block.

Controlled Current Source (Three-Phase) and Controlled Voltage Source (Three-Phase) Block Enhancement: Specify input type to simulate block in frequency-and-time mode

The Controlled Current Source (Three-Phase) and Controlled Voltage Source (Three-Phase) blocks now work in both time and frequency-and-time simulation modes when you set the **Input type** parameter to **Sinusoidal magnitude and phase shift**.

Busbar (DC) Block: Model a DC connector for load-flow analysis

The Busbar (DC) block models a DC connector for load-flow analysis. Use this block with Simscape Electrical DC networks that are compatible with frequency-time simulation mode.

Three-Phase Transformer Enhancements: Model three-phase transformers with ideal options

You can now model the Three-Winding Transformer (Three-Phase), Two-Winding Transformer (Three-Phase), and Zigzag-Delta-Wye Transformer blocks in ideal conditions by omitting the leakage reactance, the magnetizing resistance, and the magnetizing reactance.

Simplified Synchronous Machine Block Enhancement: Initialize the machine from terminal voltage and power requirements or from load flow analysis

You can now initialize the Simplified Synchronous Machine block from terminal voltage and power requirements or from load flow analysis. To choose the initialization option, in the **Initial Conditions** settings, set **Initialization option** to either:

- Set real power, reactive power, terminal voltage and terminal phase — Set an operating point regardless of the connected network.
- Set targets for rotor angle and Park's transform variables — Specify the priority and initial target values for block variables before simulation by using the **Variables** settings.
- Set targets for load flow variables — Select a bus type and specify the related parameters for a load-flow analysis.

FEM-Parameterized PMSM Block Enhancement: Tabulate iron losses from current and speed

You can now model iron losses in the FEM-Parameterized PMSM block by tabulating them from current and speed.

To model the iron losses, in the **Iron Losses** setting, set the **Model** parameter to the new **Tabulate with current and speed** option.

FEM-Parameterized Linear Actuator and FEM-Parameterized Rotary Actuator Block Enhancements: Model iron losses using magnetizing resistance

You can now model iron losses in the FEM-Parameterized Linear Actuator and FEM-Parameterized Rotary Actuator blocks by using magnetizing resistance.

To model the iron losses, in the **Magnetic Force** setting, set the **Magnetizing resistance** parameter to a value other than `inf`.

Switched Reluctance Machine (Multi-Phase) Block Enhancement: Model a six-phase switched reluctance machine

The Switched Reluctance Machine (Multi-Phase) block now allows you to model six stator phases.

To model a six-phase switched reluctance machine, in the **Main** setting, set the **Number of stator phases** parameter to **Six-phase**.

Switched Reluctance Machine and Switched Reluctance Machine (Multi-Phase) Block Enhancements: Tabulate iron losses from current, speed, and rotor angle

You can now model iron losses in the Switched Reluctance Machine and Switched Reluctance Machine (Multi-Phase) blocks by tabulating them from current, speed, and rotor angle.

To model the iron losses, in the **Iron Losses** setting, set the **Model** parameter to either Tabulate losses from current and speed or Tabulate losses from current, speed, and angle.

Synchronous Reluctance Machine Block Enhancement: Model iron losses

You can now model iron losses in the Synchronous Reluctance Machine block.

To model the iron losses, in the **Iron Losses** setting, set the **Model** parameter to either Specify magnetizing resistance, Specify tabulated Steinmetz coefficients, or Tabulate with current and speed.

Motor & Drive (System Level) Block Enhancement: Plot an efficiency map and tabulate the torque envelope from speed and supply voltage

The Motor & Drive (System Level) block now allows you to tabulate the torque envelope from speed and supply voltage.

You can also visualize the map of the torque-speed envelope and the steady-state electrical efficiency of the motor and drive. To plot the efficiency map, double-click the Motor & Drive (System Level) block and, in the **Description** tab, click Plot efficiency map.

Bidirectional DC-DC Converter Enhancement: Model a dual active bridge converter with eight switching devices

You can now model a dual active bridge converter with eight switching devices by using the Bidirectional DC-DC Converter block. This bidirectional DC-DC converter variant contains two full bridges. The left bridge is the input or high-voltage side of the converter. The right bridge is the output or low-voltage side of the converter.

DC-DC Converter Enhancement: Tabulate efficiency as a function of input voltage

In previous releases, you could tabulate the efficiency of the DC-DC Converter block as a function of the output current and temperature. In R2022a, you can now tabulate the efficiency as a function of the input voltage.

To tabulate the efficiency as a function of the input voltage, set the new **Variables for tabulation** parameter to either Output current and input voltage or Output current, input voltage, and temperature.

Eight-Pulse Gate Multiplexer Block: Multiplex gate input signals to a Bidirectional DC-DC Converter block

The Eight-Pulse Gate Multiplexer block routes gate voltage signals to the eight switching devices in a Bidirectional DC-DC Converter block. The block multiplexes the eight separate gate signals into a single vector.

Single-Phase Half-Bridge Inverter Examples: Simulate inverters with an optional thermal port

The “Build and Simulate a Single-Phase Half-Bridge Inverter with Ideal Switches” and “Build and Simulate Single-Phase Half-Bridge Inverter With Ideal Switches and Thermal Port” tutorials show you how to build and analyze a single-phase half-bridge inverter controlled by using sinusoidal pulse-width modulation (SPWM), then simulate the effects of generated heat and device temperature.

Featured Examples

New examples in this release include:

- “Six-Phase Switched Reluctance Machine Control” — This example shows how to control the rotor speed in an electrical drive based on a six-phase switched reluctance machine (SRM).
- “Mixed AC/DC System Loadflow” — This example shows how to use the Load Flow Analyzer app to review the load flow results of a mixed AC/DC system.
- “Four-Phase PMSM Velocity Control” — This example shows how to control the rotor angular velocity in an electrical-traction drive based on a four-phase permanent magnet synchronous machine (PMSM).
- “Four-Phase PMSM Torque Control” — This example shows how to control the torque in an electrical-traction drive based on a four-phase permanent magnet synchronous machine (PMSM).
- “Dual Active Bridge Control” — This example shows how to control the output voltage of a dual active bridge DC-DC converter.
- “Green Hydrogen Microgrid” — This example shows a DC islanded microgrid that provides power to an electrolyzer using a solar array and an energy storage system.
- “Wind Turbine” — This example shows how to model, parameterize, and test a wind turbine with a supervisory, pitch angle, and maximum power point tracking (MPPT) and derating control.
- “Microgrid with Electric Vehicles V2G (Vehicle-to-Grid) Support” — This example shows how to model a microgrid and regulate its frequency by using vehicle-to-grid (V2G) support from electric vehicles (EVs).
- “Power Converter Voltage Stabilizer” — This example shows a voltage stabilizer circuit that uses a full-wave rectifier, a single-wave inverter, and a buck-boost transformer to achieve voltage regulation.
- “Manage Model Fidelity Using Variants” — This example shows how to compare modeling different fidelity levels by using variants.
- “Test Harness to Generate IV Characteristics of N-Channel IGBT” — This example provides a test harness for estimating the current-voltage characteristics of an N-Channel IGBT.
- “Test Harness to Generate High-Power IGBT Device Characteristics” — This example provides a test harness for estimating the switching loss for different parameters of an N-Channel IGBT.

- “Three-Phase High-Power Converter Design and Analysis Workflow” — This example shows how to design and analyze high-power converters.

Functionality being removed or changed

Classes older than R2018b have been removed

Errors

Classes from Simscape Electronics™ and Simscape Power Systems™ have been removed. Update to use Simscape Electrical instead.

Monostable Flip-Flop Block: Initial condition now relates to the output

Warns

The **Initial condition** parameter of the Monostable Flip-Flop block has been renamed to **Initial output** and now describes the output of the block instead of the input.

As a result of this change, at $t=0$, the Monostable Flip-Flop block simulation results might be different.

Specialized Power Systems

Power Converters Blocks Enhancement: Implement average model converters without simulating rectifier mode

You can now set the **Model type** parameter to Average model (No rectifier mode) for these blocks:

- Boost Converter
- Buck Converter
- Two-Level Converter
- Half-Bridge Converter
- Full-Bridge Converter
- Two-Quadrant DC/DC Converter
- Three-Level NPC Converter

The Average model (No rectifier mode) option allows you to model converters by using a voltage source directly controlled by the reference voltage. The model does not require a PWM generator and does not simulate rectifier mode.

Multi-Winding Transformer Block Enhancement: Specify initial flux value

You can now specify the initial flux for the Multi-Winding Transformer block. To enable the **Initial flux** parameter, in the **Parameters** setting, select the **Specify initial flux** box.

Featured Examples

New examples in this release include:

- “Single-Phase Dual Active Bridge Converter DC/DC Converter” — This example shows the operation of a single-phase dual active bridge (DAB) DC/DC converter.
- “Running a Grid-Connected PV Array Model in Real-Time Using Decoupling Line Blocks” — This example shows how to simulate a grid-connected PV array model in real-time and how to use Decoupling Line blocks to avoid CPU overload by dividing the model into two concurrent tasks on a Speedgoat real-time target machine.

R2021b

Version: 7.6

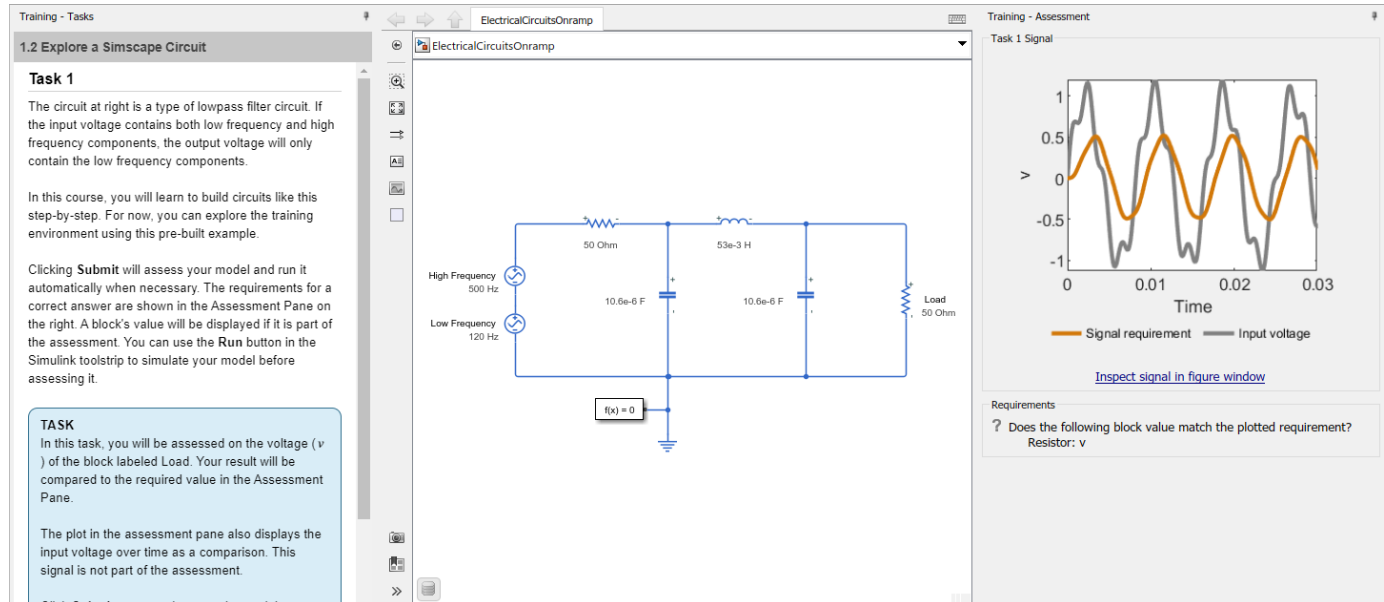
New Features

Bug Fixes

Simscape Electrical

Circuit Simulation Onramp: Self-paced, interactive tutorial included with Simscape Electrical license

Circuit Simulation Onramp is a self-paced, interactive tutorial that helps you get started with Simscape Electrical. To teach concepts incrementally, Circuit Simulation Onramp uses tasks. You receive automated assessments and feedback after submitting tasks.



Your progress is saved when you exit the training, so you can complete the training in multiple sessions.

To open Circuit Simulation Onramp, perform one of these actions:

- On the Simulink® Start Page, click **Circuit Simulation Onramp**.
- In the MATLAB Command Window, enter `learning.simulink.launchOnramp('circuits')`.

Using Circuit Simulation Onramp, you can:

- Review Simscape basics
- Explore RC and RLC circuits (where R is a resistor, C is a capacitor, and L is an inductor)
- Implement nonlinear elements to simulate a power supply circuit, including fault protection
- Model circuits that contain operational amplifiers (op amps)
- Simulate electrical filters and analyzes their performance
- Practice what you learn with a Wein Bridge oscillator project

BLDC Block: Model a specific brushless DC motor from multiple built-in parameterizations

You can now parameterize the BLDC block by using multiple built-in parameterizations that represent components by specific suppliers. The parameterizations of these brushless DC motors match the

manufacturer data sheets. To load a predefined parameterization, double-click on the BLDC block, click the **Select a predefined parameterization** hyperlink, and select the part you want to use from the list of available components.

Diode Block: Use reverse recovery energy to model charge dynamics

You can now model the charge dynamics of the Diode block using reverse recovery energy. To use reverse recovery energy, in the **Capacitance** tab, set the **Charge Dynamics** parameter to Use peak reverse current and reverse recovery energy.

This option is also available in the **Integral Diode** setting of these ideal switching semiconductor blocks:

- GTO
- IGBT (Ideal, Switching)
- MOSFET (Ideal, Switching)
- Thyristor (Piecewise Linear)

Machine Blocks: Simulate the effects of generated heat and device temperature by exposing the thermal port

You can now expose the thermal port and enable thermal parameters for these blocks:

- Induction Machine Squirrel Cage
- Induction Machine Wound Rotor
- Simplified Synchronous Machine
- Synchronous Machine Salient Pole
- Synchronous Machine Round Rotor
- Synchronous Machine Model 2.1
- Induction Machine (Single-Phase)
- PMSM (Single-Phase)
- PMSM (Five-Phase)
- PMSM (Six-Phase)
- Synchronous Machine (Six-Phase)
- Switched Reluctance Machine

Solar Cell Block: Model a specific solar cell from multiple built-in parameterizations

You can now parameterize the Solar Cell block by using multiple built-in parameterizations that represent components by specific suppliers. The parameterizations of these solar cell modules match the manufacturer data sheets. To load a predefined parameterization, double-click the Solar Cell block, click the **Select a predefined parameterization** hyperlink, and select the part you want to use from the list of available components.

Transmission Line (Three-Phase) Block: Model faults

The Transmission Line (Three-Phase) block now supports fault modeling. To model faults, set the **Fault type** parameter to:

- Single-phase to ground (a-g)
- Single-phase to ground (b-g)
- Single-phase to ground (c-g)
- Two-phase (a-b)
- Two-phase (b-c)
- Two-phase (c-a)
- Two-phase to ground (a-b-g)
- Two-phase to ground (b-c-g)
- Two-phase to ground (c-a-g)
- Three-phase (a-b-c)
- Three-phase to ground (a-b-c-g)

Delta-Connected Variable Load Block: Model a three-phase variable load wired in delta configuration

The Delta-Connected Variable Load block models a three-phase variable load wired in a delta configuration. Each limb of the load contains a resistor. The block calculates the resistance required to draw the real power of the physical signal input **P** at the rated voltage that you specify.

Diode Block Enhancement: Tabulate voltage as a function of temperature and current

You can now parameterize the Diode block using a tabulated I-V curve by providing the voltage as a function of temperature and current. Set the **Diode model** and **Table type** parameters to Tabulated I-V curve and Table in $V_f(T_j, I_f)$ form, respectively.

This also applies to the integral diode of these ideal switching semiconductor blocks:

- GTO
- IGBT (Ideal, Switching)
- MOSFET (Ideal, Switching)
- Thyristor (Piecewise Linear)

GTO and Thyristor (Piecewise Linear) Block Enhancements: Measure the current at a specific time after reaching the current threshold

If you use the GTO and Thyristor (Piecewise Linear) blocks as partially controlled devices, you can now set the **Wait time before switch-on current measurement** parameter to specify the time after which you want to start measuring the turn-on current once the current value is greater than the holding current.

Half-Bridge (Ideal, Switching) Block: Model a half-bridge with ideal switches and thermal port

The Half-Bridge (Ideal, Switching) block models a half-bridge with ideal switches and a thermal port. To choose the ideal switching device, set the **Switching device** parameter to either MOSFET, IGBT, or GTO.

You can specify an integral protection diode for each switching device. An integral diode protects the semiconductor device by providing a conduction path for the reverse current. An inductive load can produce a high reverse-voltage spike when the semiconductor device suddenly switches off the voltage supply to the load.

Heatsink Block: Dissipate heat from power semiconductors to ambient temperature

The Heatsink block models a heatsink that dissipates heat from power semiconductors. The heat from the case conducts through the fins and dissipates to the ambient temperature through convection.

You can parameterize this block from a datasheet, from tabulated heat transfer properties, or from geometry that assumes an empirical convection correlation.

Cauer Thermal Model Block: Simulate heat transfer through multiple layers of a semiconductor module

The Cauer Thermal Model Element block is now the Cauer Thermal Model block.

Unlike the Cauer Thermal Model Element block, which simulated the heat transfer through a single individual layer of a semiconductor module, the Cauer Thermal Model block represents this heat transfer through multiple layers. A Cauer Thermal Model block contains multiple Cauer Thermal Model Element components.

Foster Thermal Model Block: Initialize block in steady state or specify initial conditions

You can initialize the Foster Thermal Model block in steady state or specify the initial node temperatures.

To initialize the block in steady state, set the **Start from steady state** parameter to On. To specify the initial node temperatures and enable the **Variables** tab, set **Start from steady state** to Off. The first node temperature corresponds to the temperature in the **A** port.

Thermal Network for Semiconductor Devices Enhancement: Simulate the heat generated by switching events and conduction losses with a Cauer model with N elements

You can now choose an additional option to model the thermal network of your semiconductor devices when designing models that include a Cauer Thermal Model block with N elements.

If you want to simulate the generated heat and device temperature, expose the thermal port by:

- 1 Right-clicking the block where you want to show the thermal port
- 2 Selecting **Simscape > Block choices > Show thermal port**

When you expose the thermal port, the parameters for that block contain an additional section, **Thermal Port**. The parameters in this section depend on the value you set for the **Thermal network** parameter:

- Specify junction and thermal case parameters
- Cauer Model
- Foster Model
- External Model

Stepper Motor Driver Block: Simulate microstepping mode to increase the resolution of the position of the rotor

To increase the resolution of the position of the rotor, the Stepper Motor Driver block now supports microstepping mode. To select the fractional stepping resolution, specify the value of the **Number of microsteps per step** parameter.

Solar Cell Block: Specify parallel-connected strings in series-connected cells

When using the Solar Cell block, you can now reduce the solar panel voltage and improve the converter design and panels connections by modeling parallel-connected strings. Specify the number of strings with the **Number of parallel-connected strings** parameter.

Induction Machine Enhancements: Switch between induction machine types in block mask

Prior to R2021b, to model different types of induction machine, you had to right-click the block in your model and, from the context menu, select **Simscape > Block choices** and select one of the variants.

You can now model the different types of induction machines by setting the **Type** parameter to either Split-phase, Capacitor-start, Capacitor-start-capacitor-run, or Main and auxiliary windings for these blocks:

- Induction Machine (Single-Phase)
- PMSM (Single-Phase)

Composite or Expanded Ports Enhancement: Switch between composite and expanded ports in block mask

Prior to R2021b, to switch between composite and expanded ports, you had to right-click the block in your model and, from the context menu, select **Simscape > Block choices** and select one of the variants.

You can now switch between composite and expanded ports by setting the **Electrical connection** parameter to either Composite three-phase ports or Expanded three-phase ports for these blocks:

- Induction Machine Squirrel Cage
- Induction Machine Wound Rotor
- Simplified Synchronous Machine
- Synchronous Machine Salient Pole
- Synchronous Machine Round Rotor
- Synchronous Machine Model 2.1

ee_convertedMosfetValidation Function: Compare MOSFET conversions results with SIMetrix

The `ee_convertedMosfetValidation` function now allows you to compare the simulation results of MOSFET conversions with the results from a SPICE netlist file simulated in SIMetrix.

subcircuit2ssc Function: Support smoothing functions

The `subcircuit2ssc` function now automatically smooths certain basic mathematical SPICE functions that are not continuous. To disable the smoothing option, double-click the generated Simscape component block and set **Specify function smoothing parameters** to No.

For more information, see the Mathematical Functions section in *Converting a SPICE Netlist to Simscape Blocks*.

ee_importDeviceParameters Function: Import device parameters from an XML file to a Simscape block

The `ee_importDeviceParameters` function extracts the parameters from an XML file and imports them to the specified Simscape™ block, based on the parameterization format of the file.

SM DC4C Block: Model a discrete-time or continuous-time synchronous machine DC4C excitation system

The SM DC4C block models a discrete-time or continuous-time synchronous machine type DC4C excitation system that conforms to IEEE 421.5-2016. Use this block to model the control and regulation of the field voltage of a synchronous machine that operates as a generator using a DC commutator rotating exciter.

Moving Average and Fourier Analysis Block Enhancements: Specify buffer size

The Moving Average, Moving Average (Variable Frequency), and Fourier Analysis blocks now allow you to specify the buffer size. To enable this parameter, set the **Sample time (-1 for inherited)** parameter to 0.

PWM Generator Block: Specify phase delay

The PWM Generator block now allows you to specify the phase delay to change the initial state of the counter.

Featured Examples

New examples in this release include:

- Mars Helicopter System-Level Design — This example shows how to use Simscape Electrical to design a helicopter with coaxial rotors that is capable of flying on Mars.
- Import a Dual Three-Phase SPMSM from Motor-CAD — This example shows how you can import a six-phase motor design from Motor-CAD into a Simscape Electrical simulation.
- Bidirectional DC-DC Converter Current Control — This example shows how to control the inductor current of a Bidirectional DC-DC Converter.
- Lithium Pack Fault Protection — This example shows how to model fault and fault protection by using a fuse in an automotive battery pack.
- Import PMSM Flux Linkage Data from JMAG-RT — This example shows how you can import a motor design from JMAG(R)-RT into a Simscape simulation.
- Totem-Pole PFC — This example shows how to control the rectified voltage in a totem-pole power factor correction (PFC).
- Import IGBT Parameters From Hitachi — This example shows how you can import parameters from a Hitachi IGBT module and save them in an IGBT(Ideal, Switching) block.
- Fuel Cell System — This example shows a fuel cell system that operates at stoichiometric conditions and nominal parameters.
- Two-Phase DC-DC Converter Current Control — This example shows how to control the current of a two-phase interleaved bidirectional DC-DC converter.
- Distance Relay Protection in AC Microgrid — This example shows how to model a distance relay in an AC microgrid.
- Washing Machine Fault Analysis — This example shows how to model a washing machine and introduce a fault in its operation.
- Overcurrent Relay Protection in AC Microgrid — This example shows how to model an overcurrent relay in an AC microgrid.
- Smartphone Charging — This example shows how to set up constant current (CC) and constant voltage (CV) charging on a typical battery in a smartphone.
- DC Fast Charger for Electric Vehicle — This example models a DC fast charging station connected with the battery pack of an Electric Vehicle (EV).
- Analysis of Synchronous Buck Converter with Self Turn-On — This example analyzes the self turn-on mechanism in a synchronous buck converter.
- Calculate Losses in a Power Converter — This example shows how to calculate the losses in a power converter by using the `ee_getPowerLossSummary` function and the `power_dissipated` variable.

Specialized Power Systems

Three-Phase Tap-Changing Transformer (Two-Windings) and Three-Phase Tap-Changing Transformer (Three-Windings) Blocks: Model two-winding and three-winding three-phase tap changing transformers

The Three-Phase Tap-Changing Transformer (Two-Windings) and Three-Phase Tap-Changing Transformer (Three-Windings) blocks model a two-winding and a three-winding three-phase tap changing transformer. The position of the tap and the method used to control it is determine the transformer voltage ratio.

Three-Phase Autotransformer with Tertiary Winding: Model a three-phase autotransformer

The Three-Phase Autotransformer with Tertiary Winding block models a three-phase autotransformer. Each phase consists of three windings: w1, a series winding that is connected in Yn, w2, a common winding connected in Yn, and w3, a tertiary winding connected in delta.

Impedance Measurement Tool App: Compute impedance versus frequency of Impedance Measurement blocks in a model

The Impedance Measurement Tool app computes the impedance versus frequency of Impedance Measurement blocks in a model.

Linear System Analyzer App: Create a state-space object of the linear part of an electrical circuit

The Linear System Analyzer app creates a state-space (SS) object of the linear part of the electrical circuit for the specified model. The app models the nonlinear elements, such as the Distributed Parameters Line and Distributed Parameters Line (Frequency-Dependent) blocks, as inputs and outputs to the SS object by modeling them as the current source in the feedback to the state-space system.

Transformer Hysteresis Design Tool App: View and edit hysteresis characteristic

The Transformer Hysteresis Design Tool 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.

Report Generator App: Generate report of steady-state variables, initial states, and machine load flow for model

The Report Generator app generates reports of steady-state variables, initial states, netlist representation, and machine load flow for a model.

Statistical Flicker Level Analyzer App: Perform statistical analysis of instantaneous flicker level signal

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).

Featured Examples

New examples in this release include:

- **Impact of GIC on Autotransformer Saturation** — This example illustrates the impact of geomagnetic-induced-currents (GIC) on autotransformer saturation.
- **Islanded Operation of an Inverter-based Microgrid Using Droop Control Technique** — This example shows the islanded operation of an inverter-based microgrid by using the droop control technique.

R2021a

Version: 7.5

New Features

Bug Fixes

Version History

Simscape Electrical

ee_comtradeDataset Function: Read and extract values from a COMTRADE record

The `ee_comtradeDataset` function reads a COMTRADE record and outputs a dataset object with configuration, header, data, and information properties.

N-Channel MOSFET and P-Channel MOSFET Block Enhancements: Parameterize MOSFET by a 2-D or 3-D lookup table approximation to the I-V curve

You can now parameterize the **Threshold-based** variant of the N-Channel MOSFET and P-Channel MOSFET blocks by a 2-D or 3-D lookup table approximation to the I-V (current-voltage) curve.

To access this new parameterization, set the **Parameterization** parameter to either `Lookup table (2-D, temperature independent)` or `Lookup table (3-D, temperature dependent)`.

N-Channel MOSFET and P-Channel MOSFET Block Enhancements: Tabulate body diode to the I-V curve

The N-Channel MOSFET and P-Channel MOSFET blocks now model the body diode either as an ideal, exponential diode or as a tabulated diode.

To model a tabulated diode, set the **Model body diode** parameter to `Tabulated I-V curve`.

ee_getBillOfMaterials Function: Generate report for model of current system and all subsystems

The `ee_getBillOfMaterials` function generates a report of all subsystems for the top-level SLX model. The report is an HTML page that contains an overview of the model that generated the report, a list of physical components from the Simscape Foundation and Simscape Electrical libraries used inside the model, and the values of the parameters for each component in the model.

SPICE Conversion Assistant: Model parasitic values in capacitors and inductors

The SPICE Conversion Assistant now supports parasitic value modeling. To introduce parasitic values in the generated Simscape component files, set the **Specify parasitics values** parameter to `Yes`. Then specify the value of the **Capacitor parasitic series resistance** or **Inductor parasitic parallel conductance** parameters. For more information, see [Converting a SPICE Netlist to Simscape Blocks](#).

Diode Block Enhancement: Model faults and trigger mechanisms

The Diode block now allows you to model three types of faults:

- `Open` — Failure due to metallization burnout

- **Short** — Failure due to punch-through
- **Parameter shift** — Failure due to aging

The block can trigger fault events:

- At a specific time
- When a current limit, a voltage limit, or a temperature limit are exceeded for longer than a specific time interval

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

ee_getPowerLossSummary and ee_getPowerLossTimeSeries Function Enhancements: Calculate switching losses for semiconductor blocks in a model

The `ee_getPowerLossSummary` and `ee_getPowerLossTimeSeries` functions now also calculate switching losses for semiconductor blocks in a model, if available, based on logged simulation data.

SPICE-Imported MOSFET Block Enhancement: Model additional pre-parameterized MOSFET Devices

You can now choose several additional parameterizations for the SPICE-Imported MOSFET block. For more information on the available parameterizations, see the Available Parameterizations table.

Ideal Switching Device Block Enhancements: Improve switching loss implementation

A step in junction temperature now reflects the switching losses for these blocks:

- GTO
- IGBT (Ideal, Switching)
- MOSFET (Ideal, Switching)
- Thyristor (Piecewise Linear)

Version History

As a result of this change, the **Energy dissipation time constant** parameter has been removed. A step in junction temperature now reflects the switching losses. If your model contains a thermal mass directly connected to the thermal port of these blocks, remove it and model the thermal mass inside the component itself.

Piezo Bender Block: Model a piezoelectric bimorph beam of rectangular cross-section

The Piezo Bender block models a piezoelectric bimorph beam of rectangular cross-section.

A piezo bender is a piezoelectric device that bends when you apply an electrical potential between its plates. Conversely, when a piezo bender bends, it generates an electrical potential.

Multiple Machine Block Enhancement: Hide neutral phase port when zero sequence is excluded

If you set the **Zero sequence** parameter to `Exclude`, the neutral phase port, **n**, is hidden for these blocks:

- BLDC
- Hybrid Excitation PMSM
- PMSM
- PMSM (Five-Phase)
- PMSM (Six-Phase)
- PMLSM
- Synchronous Reluctance Machine
- Synchronous Machine (Six-Phase)
- Synchronous Machine Model 1.0
- Synchronous Machine Round Rotor
- Synchronous Machine Salient Pole

Version History

If your existing model contains any of these blocks with the **Zero sequence** parameter set to `Exclude`, you need to update the model to reflect the change in visibility of the neutral phase port, **n**.

Fuel Cell Block: Model a fuel cell electrical system with different levels of details

The Fuel Cell block models a fuel cell that converts the chemical energy of hydrogen into electrical energy. You can use the **Model fidelity** parameter to set the Fuel Cell block to two different levels of fidelity:

- `Simplified - nominal conditions` — The block calculates the Nernst voltage at the nominal condition of temperature and pressure.
- `Detailed with physical inputs` — The block calculates the Nernst voltage by considering the pressures and flow rates of fuel and air.

FEM-Parameterized PMSM Block Enhancement: Additional iron losses modeling and optional neutral port for 3-D flux linkage variant

You can now model iron losses for the FEM-Parameterized PMSM block by using Steinmetz equations.

You can also now expose the neutral port, **n**, for the **3-D flux linkage data** variant, with or without thermal ports.

DC-DC Converter Block Enhancement: Tabulate efficiencies and calculate converter losses as a function of output current and temperature

You can now tabulate efficiencies in the DC-DC Converter block and model losses based on the output current and temperature.

Compound Motor Block: Model a compound motor with electrical and torque characteristics and fault modeling

The Compound Motor block represents the electrical and torque characteristics of a compound motor.

Use this block to simulate a compound motor system with series and parallel field windings and rotating armature.

Battery and Battery (Table-Based) Block Enhancements: Model battery degradation due to calendar aging

You can model the battery performance deterioration that occurs when the battery is not used for the Battery and Battery (Table-Based) blocks. Calendar aging affects the internal resistance.

To model the calendar aging in the Battery block, in the **Main** tab, set the **Battery charge capacity** parameter to **Finite** and, in the **Calendar aging** tab, set the **Calendar aging** parameter to **Enabled**.

To model the calendar aging in the Battery (Table-Based) block, set the **Calendar aging model** parameter to either:

- Equation-based
- Tabulated: temperature
- Tabulated: time and temperature

Expanded Mathematical Transforms Library: Transform five-phase or six-phase components to components in a rotating reference frame and vice versa

You can use the new mathematical transform blocks for five-phase and six-phase systems. These Simulink blocks are modular, decoupled, customizable, and compatible with Embedded Coder®. Each block provides open access to the implementation source code.

To access the new blocks, in the Simulink Library Browser, select **Simscape > Electrical > Control > Mathematical Transforms** to access:

- Coordinate Transform (Five-Phase)
- Decoupled Transform (Six-Phase)
- Inverse Coordinate Transform (Five-Phase)
- Inverse Decoupled Transform (Six-Phase)

Expanded Pulse Width Modulation Library: Generate gating times and controlling pulses for five-phase power converter control

You can use the new pulse width modulation (PWM) blocks to generate PWM waves that can be used to control power converters. These Simulink blocks are modular, decoupled, customizable, and compatible with Embedded Coder. Each block provides open access to the implementation source code.

To access the new blocks, in the Simulink Library Browser, select **Simscape > Electrical > Control > Pulse Width Modulation** to access:

- PWM Generator (Five-phase, Two-level)
- PWM Gate Signal Generator (Five-phase, Two-level)
- PWM Timing and Waveform Generator (Five-phase, Two-level)

Winding Block Enhancement: Model temporal and behavioral faults

You can now model temporal and behavioral faults in the Winding block. To model faults, set the **Enable faults** parameter to Yes.

The block can trigger the start of fault transition:

- At a specific time
- After voltage exceeds the maximum permissible value a certain number of times
- When current exceeds the maximum permissible value for longer than a specific time interval

You can enable or disable these trigger mechanisms separately, or use them together if more than one trigger mechanism is required in a simulation.

Rotating Air Gap Block: Model an air gap between a stator tooth and a rotating permanent magnet rotor

The Rotating Air Gap block models an air gap between a stator tooth and a rotating permanent magnet rotor. This block assumes that the rotor magnets are surface mounted and that the associated induced voltage is sinusoidal.

Use this block as a basic building block to create a magnetic representation of a permanent magnet synchronous motor (PMSM).

Average-Value Voltage Source Converter Block Enhancement: Parameterize converter losses and expose thermal port

To better model the main heat sources for a converter, you can now parameterize the power losses of the Average-Value Voltage Source Converter (Three-Phase) block, by setting the **Losses parameterization** parameter to the desired parameterization.

To expose the thermal port, right-click the block, select **Simscape > Block choices**, and then select the desired block variant with thermal ports: **Composite three-phase ports | Show thermal port** or **Expanded three-phase ports | Show thermal port**. This action displays the thermal port on the block icon and enables the **Thermal mass** parameter.

Constant Current Load (Three-Phase) Block: Model a faultable constant current load for a three-phase supply

The Constant Current Load (Three-Phase) block implements a constant current load for a three-phase supply. The load behavior changes according to the voltage received from the three-phase supply.

You can model an electrical temporal or behavioral fault as an open circuit by setting the **Enable open circuit fault** parameter to Yes.

Constant Current Load Block: Model a faultable constant current load for a DC or AC supply

The Constant Current Load block implements a constant current load for a DC or AC supply. The load behavior changes according to the voltage received from the supply.

You can model an electrical temporal or behavioral fault as an open circuit by setting the **Enable open circuit fault** parameter to Yes.

PMSM Block Enhancement: Model a specific permanent magnet synchronous motor from multiple built-in parameterizations

There are now multiple available built-in parameterizations for the PMSM block.

This pre-parameterization data allows you to use the block to represent components by specific suppliers. The parameterizations of these batteries match the manufacturer data sheets. To load a predefined parameterization, click on the **Select a predefined parameterization** hyperlink in the PMSM block mask and select the part you want to use from the list of available components.

Converters Block Enhancements: Model inductor series resistance

To accurately model damping, you can now specify the series resistance of the inductor by setting the **Inductor series resistance** parameter in these blocks:

- Bidirectional DC-DC Converter
- Boost Converter
- Buck Converter
- Buck-Boost Converter

Three-Phase Motor Block Enhancements: Model thermal ports

You can now model the thermal effects for these blocks:

- Hybrid Excitation PMSM
- Synchronous Machine Model 1.0
- Synchronous Reluctance Machine

To expose the thermal ports, right-click the block in your model, select **Simscape > Block choices**, and then select the desired block variant with thermal ports: **Composite three-phase ports | Show**

thermal port or **Expanded three-phase ports | Show thermal port**. This action displays the thermal ports on the block icon, and exposes the **Thermal** parameters.

Featured Examples

New examples in this release include:

- **Lithium Pack Cell Balancing** — This example shows how to implement passive cell balancing for a lithium-ion battery pack.
- **Lithium Pack Cooling** — This example shows how to model an automotive battery pack for thermal management tasks.
- **Lithium Pack Short Circuit** — This example shows how to model a short circuit in a lithium-ion battery module.
- **Lithium Pack DCFC** — This example shows how to model an automotive battery pack for DC fast charging tasks.
- **SPICE Conversion of a MOSFET Subcircuit and Validation** — This example shows how to convert a metal-oxide-semiconductor field-effect transistor (MOSFET) subcircuit into an equivalent Simscape component and compare the SPICE and Simscape plots for some standard MOSFET characteristics.
- **Three-Phase Grid-Tied Inverter Optimal Current Control** — This example shows how to control the currents in a grid-tied inverter system.
- **Input Admittance Response of RLC Ladder Network with Mutual Coupling Between Multiple Coils** — This example shows how to model a four-section ladder network that comprises RLC components with mutual coupling between multiple coils.
- **Permanent Magnet Synchronous Generator Battery Charging** — This example shows how to use a permanent magnet synchronous generator (PMSG) to charge a battery.
- **Three-Phase Matrix Converter with Venturini Modulation** — This example shows how to use Venturini modulation techniques to compute the duty cycles and logic statements of a three-phase matrix converter that drives a static load.
- **Scalar Control in Matrix Converter-Fed Induction Machine Drive** — This example shows how to control the rotor speed in a matrix converter-fed induction machine drive by using the scalar V/f control method.
- **Piezo Bender Energy Harvester** — This example shows how to model a device that harvests energy from a vibrating object by using a piezo bender.
- **Compound Motor Design Optimization** — This example shows how to find design parameters that optimize a compound motor torque-speed curve to match the desired curve.
- **Faulted PMSM** — This example shows how to model a faulted PMSM using Simscape Electrical.
- **Parameterize the Lookup Table-Based MOSFET from SPICE** — This example shows how to use the SPICE simulation results of a metal-oxide-semiconductor field-effect transistor (MOSFET) to set the parameter values of an N-Channel MOSFET (Lookup table-based) block in Simscape.
- **Import Efficiency Map Data from Motor-CAD** — This example shows how you can import efficiency map data from Motor-CAD to parameterize the Simscape Electrical Motor & Drive (System Level) block.
- **Analysis of Solar Photovoltaic System Shading** — This example shows how to implement shading effects in a solar photovoltaics (PV) plant or module.

Specialized Power Systems

Library Restructure: Find critical Specialized Power Systems blocks more easily

The Specialized Power System libraries have undergone a complete overhaul and rework.

Version History

As a result of these changes, some blocks have been removed from their libraries or moved to other sub-libraries. It is no longer possible to export models that contain these blocks to previous releases.

Asynchronous Machine Blocks Enhancement: Specify the turn ratio

You can now specify the voltage turn ratio of the Asynchronous Machine blocks.

To specify the voltage turn ratio, set the value of the **Voltage turn ratio** parameter.

PV Array Block Enhancement: Updated default value for modules parameter

For the PV Array block, the default value of the **Module** parameter is now User-defined. If your model contains a PV Array block with the **Module** parameter set to the previous default value, Simscape throws the following error:

```
Warning: In instantiating linked block '<blockPath>/PV Array'  
: Invalid setting in PV array block (mask) 'PV Array' for parameter  
'ModuleName'
```

Measurements and States Analyzer App: Compute initial and steady-state values of voltages and currents of model

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.

FFT Analyzer App: Perform Fourier analysis of simulation data signals

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.

Load Flow Analyzer App: Perform positive-sequence load flow or unbalanced load flow and initialize models containing load flow blocks

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. It offers most of the functionality of other load flow software available in the power utility industry

Featured Examples

New examples in this release include:

- Speed Regulation of Brushless DC Motor Drive Using a Hysteresis-Based Current Controller — This example shows the speed regulation of a brushless DC motor (BLDC) drive using a hysteresis-based current controller.
- Speed Regulation of a Brushless DC Motor Drive Using a Variable DC Link Six-Step Inverter — This example demonstrates the speed regulation of a brushless DC (BLDC) motor drive using a variable DC link six-step inverter.
- Direct Torque Control of an Induction Motor Drive — This example demonstrates the speed regulation of a variable-frequency AC drive using a hysteresis-based direct torque control (DTC) technique.
- Direct Torque Control with Space Vector Modulation of an Induction Motor Drive — This example demonstrates the speed regulation of a variable-frequency AC drive using the direct torque control (DTC) technique with space vector modulation.
- Field-Oriented Control of an Induction Motor Drive Used in a Ground Transportation System — This example demonstrates the operation of a variable-speed AC drive in a trolley bus traction system.
- Field-Oriented Control of an Interior Permanent Magnet Synchronous Motor — This example shows the wide-speed operation of an interior permanent magnet synchronous motor (IPMSM) drive. The drive uses a field-oriented control system with maximum torque per ampere (MTPA) and field-weakening control strategies.
- Field-Oriented Control of a Surface Mounted Permanent Magnet Synchronous Motor — This example shows the speed regulation of a surface-mounted permanent magnet synchronous motor (PMSM) using field-oriented control (FOC).
- Speed Regulation of a 6/4 Switched Reluctance Motor — This example shows the speed regulation of a three-phase 6/4 switched reluctance motor (SRM).

R2020b

Version: 7.4

New Features

Bug Fixes

Version History

Simscape Electrical

Modular Multilevel Converter Blocks: Model modular multilevel converters as a number of series-connected power submodules

These blocks implement modular multilevel converters with series-connected power submodules:

- Modular Multilevel Converter Arm
- Modular Multilevel Converter Leg
- Modular Multilevel Converter (Three-Phase)

Battery and Battery (Table-Based) Block Enhancements: Model the dependency of the terminal resistance on current directionality

The Battery and Battery (Table-Based) blocks now have the option to model the internal terminal resistance based on the direction of the current. To enable this dependency, set the **Current directionality** parameter to Enabled.

PMSM (Six-Phase) Block: Model a six-phase permanent synchronous motor

You can use the PMSM (Six-Phase) block to model a six-phase permanent magnet synchronous motor with sinusoidal flux distribution.

Synchronous Machine (Six-Phase) Block: Model a six-phase synchronous machine

The Synchronous Machine (Six-Phase) block models a six-phase synchronous machine, also known as a dual-star machine.

There are two groups of stator windings in a six-phase synchronous machine: the ABC group and the XYZ group. These two groups have a 30-degree phase shift.

Ideal Switching Device Block Enhancements: Improve switching loss implementation and thermal modeling options

These blocks now have improved losses and thermal modelling options:

- GTO
- IGBT (Ideal, Switching)
- MOSFET (Ideal, Switching)
- Thyristor (Piecewise Linear)

Version History

As a result of these changes, for the MOSFET (Ideal, Switching) block:

- Electrical and thermal on-state losses are now always identical. The **Thermal loss dependent on** parameter and its options, **Voltage and current** and **Voltage, current, and temperature**, have been renamed to **On-state behavior and switching losses**, **Specify constant values**, and **Tabulate with temperature and current**:
 - If you selected **Voltage and current** for **Thermal loss dependent on**, then the electrical on-state losses are unchanged and their values are determined using the on-state resistance. However, the thermal on-state losses are now also determined by the on-state resistance. Prior to R2020b, the thermal on-state losses were defined by the product of the **On-state voltage** and **Output current, Iout** parameters.
 - If you selected **Voltage, current, and temperature** for **Thermal loss dependent on**, then the thermal on-state losses are unchanged and the **On-state voltage, Vds(Tj,Ids)** parameter sets their values and the value of the electrical voltage drop. Consequently, the electrical on-state losses are now equal to the thermal on-state losses. Prior to R2020b, the electrical voltage drop and on-state losses were defined by the value of the on-state resistance.
- The **On-state voltage** parameter is no longer used.

Similar changes apply to the GTO, IGBT (Ideal, Switching), and Thyristor (Piecewise Linear) blocks.

Diode Block Enhancement: Model a tabulated I-V diode

The Diode block now supports a tabulated I-V curve parameterization. To model a tabulated diode, set the **Diode model** parameter to **Tabulated I-V curve**.

SPICE-Imported MOSFET Block: Model a predefined MOSFET parameterized by an external SPICE subcircuit

The SPICE-Imported MOSFET block models a metal-oxide-semiconductor field-effect transistor (MOSFET) that has been parameterized using an external SPICE subcircuit.

You can use this block when the MOSFET that you want to model is one of the available parameterizations.

Battery (Table-Based) Block Enhancement: Model a specific lithium-ion battery from multiple built-in parameterizations

There are now multiple available built-in parameterizations for the Battery (Table-Based) block.

This pre-parameterization data allows you to set up the block to represent components by specific suppliers. The parameterizations of these batteries match the manufacturer data sheets. To load a predefined parameterization, click on the **Select a predefined parameterization** hyperlink in the Battery (Table-Based) block mask and select the part you want to use from the list of available components.

ee_convertedmosfetvalidation Function: Validate MOSFET conversions by generating characteristics and comparing results with LTspice

To validate a SPICE conversion into an ssc file, the `ee_convertedmosfetvalidation` function allows you to generate standard MOSFET characteristics and compare the simulation results with results from a SPICE netlist file simulated in LTspice.

SPICE Conversion Assistant: Use table SPICE function

The SPICE Conversion Assistant now supports the table SPICE function, `table`. For more information, see [Converting a SPICE Netlist to Simscape Blocks](#).

SPICE NMOS and SPICE PMOS Block Enhancements: Model the transistor capacitances with Meyer gate or charge conservation models

The SPICE NMOS and SPICE PMOS blocks now support two additional options for capacitance modeling. To model capacitance using a Meyer gate model, set the **Model gate capacitance (CGS, CGD, CGB)** parameter to `Meyer gate capacitances`. To model capacitance using a charge conservation model, set the **Model gate capacitance (CGS, CGD, CGB)** parameter to `Charge conservation capacitances`.

Current-Controlled Switch and Voltage-Controlled Switch Blocks Enhancements: Model a variable-resistance or a short-transition switch

The Current-Controlled Switch and Voltage-Controlled Switch blocks now model either a variable-resistance or a short-transition switch. For a variable-resistance switch, set the **Switch model** parameter to `Smooth transition between Ion and Ioff` for the Current-Controlled Switch block and `Smooth transition between Von and Voff` for the Voltage-Controlled switch. For a short-transition switch, set **Switch model** to `Abrupt transition after delay`.

Motor & Drive (System Level) Block: Represent a generic motor and drive with closed-loop torque control

The Motor & Drive (System Level) block represents a generic motor and drive with closed-loop torque control and intermittent over-torquing operation. Use this block to model a brushless motor (such as a PMSM) and drive, or traction and actuation systems implemented using a wide range of motor types.

To enable fast simulation at a system level, the block abstracts the motor, drive electronics, and control.

Version History

The Simplified PMSM Drive block has been renamed the Motor & Drive (System Level) block and moved from the **Simscape > Electrical > Electromechanical > Permanent Magnet** library to the **Simscape > Electrical > Electromechanical > System-Level Modeling** library.

If you have a script that depends on the library path or the Simscape component path of this block, you need to update it to reflect these changes.

DC Motor Block Enhancement: Model armature winding or field winding faults

The DC Motor block now supports fault modeling. You can model an armature winding fault, a field winding fault, or both at the same time. To enable an armature winding fault, set the **Enable**

armature winding open-circuit fault parameter to Yes. To enable a field winding fault, set the **Enable field winding open-circuit fault** parameter to Yes.

Unipolar Stepper Motor Block: Model mutual inductance

The Unipolar Stepper Motor block now supports mutual inductance modeling. To model the mutual inductance, specify a value in the **Coefficient of coupling** parameter.

Simplified Synchronous Machine Block: Model a simplified synchronous machine with electromotive force (EMF)

The Simplified Synchronous Machine block models a simplified synchronous machine with a voltage source representing electromotive force. Set the **Parameterization unit** parameter to SI or Per unit to specify the internal resistance and inductance with SI or per-unit parameters.

Simplified Synchronous Machine Measurement Block: Output a per-unit measurement associated with a connected Simplified Synchronous Machine block

The Simplified Synchronous Machine Measurement block outputs a per-unit measurement associated with a connected Simplified Synchronous Machine block. The input of the Simplified Synchronous Machine Measurement block connects to the pu output port of the simplified synchronous machine block.

Delta-Wound Machines: Support zero sequence dynamics

The following blocks now model zero-sequence dynamics in a delta-wound configuration:

- BLDC
- FEM-Parameterized PMSM (in the 3-D Flux Linkage variant)
- PMLSM
- PMSM
- PMSM (Five-Phase) (only when **Winding Type** is set to **Pentacle-Wound** or **Pentagon-Wound**)

Power Sensor (Three-Phase) Block: Model an ideal sensor for active and reactive power measurement

The Power Sensor (Three-Phase) block implements an ideal sensor for active and reactive power measurement in balanced or unbalanced three-phase branches.

Load-Flow Analyzer App Enhancements: Perform a load-flow analysis through an improved toolstrip and model highlighting

The **Load-Flow Analyzer** app now features a toolstrip that comprises all the functionalities of the previous version, and an improved highlighting functionality that you can use to identify individual blocks or groups of blocks in the model.

Dynamic Load Block: Model a faultable dynamic load for a DC or AC supply

The Dynamic Load block implements a dynamic load for a DC or AC supply. The load behavior changes according to the voltage received from the supply.

You can model an electrical fault as an open circuit by setting the **Enable open circuit fault** parameter to Yes.

Dynamic Load (Three-Phase) Block Enhancement: Model a faultable balanced three-phase dynamic load for a three-phase supply

The Dynamic Load (Three-Phase) block now models a balanced dynamic load for balanced three-phase supplies. The load behavior changes according to the voltage received from the three-phase supply.

You can model an electrical fault as an open circuit by setting the **Enable open circuit fault** parameter to Yes.

Version History

The Dynamic Load (Three-Phase) block has been moved from the **Simscape > Electrical > Passive > RLC Assemblies** library to the **Simscape > Electrical > Passive** library.

If you have a script that depends on the library path or the Simscape component path of this block, you need to update it to reflect these changes.

Constant Power Load Block Enhancement: Model a faultable constant power load for an AC supply

The Constant Power Load block now supports voltages from AC supplies for frequency and time simulations.

To specify the type of the load, set the **Load type** parameter to either DC or AC.

Constant Power Load (Three-Phase) Block: Model a faultable constant power load for a three-phase supply

The Constant Power Load (Three-Phase) block implements a constant power load for a three-phase supply. The load behavior changes according to the voltage received from the three-phase supply.

You can model an electrical fault as an open circuit by setting the **Enable open circuit fault** parameter to Yes.

Delta-Connected Load, Wye-Connected Load, and RLC (Three-Phase) Block Enhancements: Model initial conditions in the Variables tab

The initial conditions parameters of the Delta-Connected Load, RLC (Three-Phase), and Wye-Connected Load blocks have been moved from the **Initial conditions** tab to the **Variables** tab.

Use the **Variables** tab to specify the priority and initial target values for the block variables before simulation. For more information, see [Set Priority and Initial Target for Block Variables](#).

Expanded Support for Power Dissipation Data: View power_dissipated simulation logging data for all three-phase transformer blocks

You can now analyze the data for a power_dissipated logging node for all three-phase transformer blocks in the Transformers library. For more information on logging, viewing, and analyzing power_dissipated data, see:

- [About the Simscape Results Explorer](#)
- [sscexplore](#)
- [ee_getEfficiency](#)
- [ee_getPowerLossSummary](#)
- [ee_getPowerLossTimeSeries](#)

PWM Generator (Multilevel) Block: Generate pulse width modulated signal or waveform for modular multilevel converters

The PWM Generator (Multilevel) block implements a PWM generator for modular multilevel converters. The block calculates gate pulses for a modular multilevel converter based on the number of power submodules and for two converter topologies.

Expanded Pulse Width Modulation Library: Generate gating times and controlling pulses for power converter control

You can use the new pulse width modulation (PWM) blocks to generate PWM waves that can be used to control power converters. These Simulink blocks are modular and decoupled, customizable, and compatible with Embedded Coder. Each block provides open access to the implementation source code.

To access the new blocks, in the Simulink Library Browser, select **Simscape > Electrical > Control > Pulse Width Modulation** to access:

- PWM Gate Signal Generator (Three-phase, Three-level)
- PWM Gate Signal Generator (Three-phase, Two-level)
- PWM Timing and Waveform Generator (Three-phase, Three-level)
- PWM Timing and Waveform Generator (Three-phase, Two-level)

Featured Examples

New examples in this release include:

- [Tolerance Study Using Monte Carlo Simulations in Resonant LLC DC-DC Converter](#) — Perform a Monte Carlo analysis to optimize the design of an LLC resonant DC-DC converter when some of its components have tolerances.
- [BLDC Position Control with Thermal Model](#) — This example shows how to control the rotor angle in a BLDC based electrical drive.

- Three-Phase PMSM Drive with Thermal Model — This example shows a Permanent Magnet Synchronous Machine (PMSM) and an inverter sized for use in a typical hybrid vehicle.
- Three-Phase Grid-Connected Rectifier Control — This example shows how to control the DC-link voltage using a grid-connected rectifier.
- Single-Phase Inverter Current Control — This example shows how to control the current in a single-phase inverter system.
- Three-Phase Grid-Tied Inverter — This example shows how to control the voltage in a grid-tied inverter system.
- Three-Phase Inverter Voltage Control — This example shows how to control the voltage in a three-phase inverter system.
- Six-Phase PMSM Torque Control — This example shows how to control the torque in an electrical-traction drive based on a six-phase PMSM.
- Six-Phase PMSM Velocity Control — This example shows how to control the rotor angular velocity in an electrical-traction drive based on a six-phase PMSM.

Specialized Power Systems

Distributed Parameters Line (Frequency-Dependent) Block: Model a multiphase distributed parameter line with high-fidelity frequency-dependent parameters

The Distributed Parameters Line (Frequency-Dependent) block implements a multiphase distributed parameter line model with high-fidelity, frequency-dependent parameters. It performs accurate transient response simulations for a wide frequency range (typically from DC to 100 kHz or more).

Decoupling Line Block: Model a single-phase distributed parameters line

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.

Decoupling Line (Three-Phase) Block: Model a three-phase distributed parameters line

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.

Power Line Parameters App: Compute RLC parameters of overhead transmission lines from conductor characteristics and tower geometry

The Power Line Parameters 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.

Specify Decoupling Lines App: Replace selected Distributed Parameters Line blocks with Decoupling Line blocks

The Specify Decoupling Lines 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.

PI Section Line Block Enhancement: Implement multi-phase transmission line with lumped parameters

The PI Section Line block now allows you to model an N -phase transmission line with parameters lumped in PI sections.

Featured Examples

New examples in this release include:

- Field-Oriented Control of an Induction Motor Drive Used in a Ground Transportation System — This example demonstrates the operation of a variable-speed AC drive in a trolleybus traction system.
- MMC-STATCOM Connected to a 735-kV Transmission System — This example demonstrates the operation of a 50 MVA, MMC STATCOM (22 power modules per arm) connected to a 735 kV grid.

R2020a

Version: 7.3

New Features

Bug Fixes

Version History

Simscape Electrical

Converters and Choppers Blocks: Select averaged switch option for switching device type

These blocks now support an additional type of switching device:

- Bidirectional DC-DC Converter
- Boost Converter
- Buck Converter
- Buck-Boost Converter
- Converter (Three-Phase)
- Four-Quadrant Chopper
- One-Quadrant Chopper
- Three-Level Converter (Three-Phase)
- Two-Quadrant Chopper

To enable the new switching device, set **Switching devices** parameter to Averaged Switch.

Load-Flow Analyzer App: Generate, visualize, sort, tune, and export steady-state voltage and power data for three-phase AC power transmission systems

The **Load-Flow Analyzer** app performs a power-flow, or load-flow, analysis for a three-phase AC electrical power transmission system that you model and configure for analysis using Simscape Electrical. The app generates two tables. One of the tables contains data for the network nodes, as represented by Busbar, Load Flow Source, synchronous machine, induction machine, and three-phase load blocks. The other table contains data for the network connections, as represented by transmission lines, cable, and transformer blocks. When you open the tool, the tables are preloaded with the specified parameter values for the relevant blocks in the current or specified model. After you run the power-flow analysis, the tables also display the steady-state voltage magnitudes, voltage phase angles, active power, and reactive power for the node and connection blocks.

The **Load-Flow Analyzer** app allows you to:

- Run a power-flow analysis.
- Highlight and update inputs for parameters in the Nodes table for busbar, load flow source, synchronous machine, induction machine, and three-phase load blocks.
- Change the load flow source, synchronous machine, and induction machine blocks' bus type.
- Select and highlight node and connection blocks in the model.
- Sort columns in the tables by increasing or decreasing values.
- Export the data to a spreadsheet, a MAT-file, or a comma-separated variable (CSV) file.

PMSM and BLDC Blocks Enhancements: Model thermal variants with empirical losses

The PMSM and BLDC blocks now support temperature dependency and iron losses. To enable these parameters, right-click the block in your model, select **Simscape > Block choices**, and then select the desired block variant with thermal ports: **Composite three-phase ports | Show thermal port** or **Expanded three-phase ports | Show thermal port**.

SPICE NIGBT block: Model a SPICE-compatible N-Channel insulated gate bipolar transistor

The SPICE NIGBT block models a SPICE n-type insulated gate bipolar transistor (IGBT).

SPICE, or Simulation Program with Integrated Circuit Emphasis, is a simulation tool for electronic circuits. You can convert some SPICE subcircuits into equivalent Simscape Electrical models by using the Environment Parameters block and SPICE-compatible blocks from the Additional Components library. For more information, see `subcircuit2ssc`.

Expanded SM Control Library: Speed up modeling by using prebuilt and documented automatic voltage regulators and power system stabilizers blocks

Save time deriving, implementing, testing, debugging, and documenting algorithm models by using blocks from the expanded SM Control library. These Simulink blocks are modular and decoupled, customizable, and compatible with Embedded Coder. Each block provides open access to the implementation source code.

To access the new blocks, in the Simulink Library Browser, select **Simscape > Electrical > Control > SM Control** to access:

- Synchronous-machine-type AC blocks — SM AC2C, SM AC3C, SM AC4C, SM AC5C, SM AC6C, SM AC7C, SM AC8C.
- Synchronous-machine-type DC blocks — SM DC1C, SM DC2C.
- Synchronous-machine-type ST blocks — SM ST1C, SM ST2C, SM ST3C, SM ST4C.
- Power system stabilizer (PSS) blocks — SM PSS1A, SM PSS2C, SM PSS7C.

Stepper Motor Block: Select a predefined parameterization

There are now multiple available built-in parameterizations for the Stepper Motor block.

This pre-parameterization data allows you to set up the block to represent a specific supplier component. To load a predefined parameterization, click the "Select a predefined parameterization" hyperlink in the Stepper Motor block mask and select the part you want to upload from the list of available components.

DC Motor Block Enhancement: Generate magnetic fields with new motor field option

You can now specify how to generate the magnetic field of a DC Motor block. Set **Field type** to **Permanent magnet** to generate the magnetic field using a permanent magnet, or **Wound** to generate it from the current flowing through the windings.

PMSM (Five-Phase) Block: Model a five-phase permanent synchronous motor

You can use the PMSM (Five-Phase) block to model a five-phase permanent magnet synchronous motor with sinusoidal flux distribution.

To specify the stator configuration, set the **Winding Type** parameter to either **Star-wound**, **Pentacle-wound**, or **Pentagon-wound**.

PMLSM Block: Model a three-phase permanent magnet linear synchronous motor

You can use the PMLSM block to model a three-phase permanent magnet linear synchronous motor with sinusoidal flux distribution.

To specify the stator configuration, set the **Winding Type** parameter to either **Wye-wound** or **Delta-wound**.

Electromechanical Library Enhancement: Model some machines with different fidelity levels

To allow for a new level of modeling fidelity, these blocks have been updated:

- BLDC
- Hybrid Excitation PMSM
- PMSM
- Synchronous Reluctance Machine
- Synchronous Machine Model 1.0

To enable the new parameters, set the **Modeling fidelity** parameter to either **Constant Ld, Lq, and PM** or **Tabulated Ld, Lq, and PM**.

Induction Machine Squirrel Cage and Induction Machine Wound Rotor Blocks Enhancements: Update Saturation Characteristic

For the saturation characteristic, the Induction Machine Squirrel Cage and Induction Machine Wound Rotor blocks now take into account the magnetic flux due to the leakage inductance.

The default values for these parameters have changed:

- **Stator leakage reactance, Xls**

- **Stator zero-sequence reactance, X_0**
- **Stator leakage inductance, L_{ls} (pu)**
- **Stator zero-sequence inductance, L_0 (pu)**
- **Terminal voltage saturation data, v (phase-phase, rms)**
- **Per-unit terminal voltage saturation data, v**

Version History

As a result of these changes, inside a model saved in an earlier release, if you have used the Induction Machine Squirrel Cage or Induction Machine Wound Rotor blocks with saturation enabled, review the saturation data.

Inductive Rotor Position Sensor Block: Obtain the rotor position

To obtain the rotor position, the Inductive Rotor Position Sensor block comprises four planar coils and a target trace, and decodes the rotational angle by using the theory of eddy current losses.

Power Transformers Blocks Enhancements: Represent magnetic saturation

The magnetic circuit and equations of these blocks have been redesigned to allow for an accurate representation of a saturable transformer:

- Two-Winding Transformer (Three-Phase)
- Three-Winding Transformer (Three-Phase)
- Zigzag-Delta-Wye Transformer

To represent the magnetic saturation of these blocks, set the **Magnetic saturation representation** parameter to `Lookup table (phi versus i)`.

Distributed Parameter Line Model for Transmission Line Block: Obtain more accurate simulations at a particular frequency

You can now model the Transmission Line block as a distributed parameter line. A distributed parameter line gives more accurate simulations at specified frequencies.

However, this is not a frequency-dependent model and its accuracy will drop outside the frequency point you specify in the **Frequency used for rlcg specification** parameter.

SPICE Diode Block Enhancements: Calculate high-injection knee current, IKF, and recombination current, ISR

The SPICE Diode block now comprises several new parameters: **High-injection knee current, IKF** and **Recombination current, ISR**.

N-Channel MOSFET and P-Channel MOSFET Block Enhancements: Parameterize four terminals

In order to access bulk, the N-Channel MOSFET and P-Channel MOSFET blocks now support four electrical nodes parameterization.

To enable the new port and parameters, in the block dialog box, in the **Main** settings, set **Parameterization** to `Four terminals`.

SPMT Switch Block: Model a single-pole multiple-throw switch

The SPMT Switch block models a single-pole multiple-throw switch. You can specify three to eight throw connections by setting the **Number of throw ports** to the desired amount.

Open connections are modeled by a resistor with a value equal to the reciprocal of the **Open conductance** parameter value. Closed connections are modeled by a resistor with a value equal to the **Closed resistance** parameter value.

Average-Value Chopper and Average-Value DC-DC Converter Blocks Enhancement: Model conduction loss dependencies by using tabulated efficiency parameterization

To model conduction loss dependencies on the provided output current, the Average-Value Chopper and Average-Value DC-DC Converter blocks now support tabulated efficiency parameterization.

Set the **Converter efficiency** parameter to `Tabulated` and specify the desired values for the **Output current vector** and **Efficiency vector (%)** parameters.

Average-Value DC-DC Converter Block Enhancement: Use voltage and current reference as control inputs

To convert the electrical energy between the connected components on either side of the converter, the Average-Value DC-DC Converter block now supports voltage and current references as control inputs.

If you set the **Control input** parameter to `Current reference`, the converter sets the output current and it computes the voltage.

Similarly, if you set the **Control input** parameter to `Voltage reference`, the converter sets the output voltage and it computes the current.

DC-DC Converter Block Enhancement: Model the voltage reference externally

You can now model the voltage reference inside the DC-DC Converter block as an external input.

To expose the voltage reference port, set the **Voltage reference** parameter to `External`.

Battery (Table-Based) Block Enhancement: Select additional extrapolation and interpolation options for table-based parameters

The Battery (Table-Based) block can now use nearest interpolation and extrapolation methods for all table-based parameters, or no extrapolation at all.

You can set the **Extrapolation method for all tables** parameter to:

- **Linear** — Estimates values beyond the dataset by creating a tangent line at the end of the known data and extending it beyond that limit.
- **Nearest** — Extrapolates a value at query point that is the value at the nearest sample grid point.
- **Error** — Returns an error if the value goes beyond the known dataset. If you select this option, the block does not use extrapolation.

Turbine-Governors Blocks: Implement speed-governing steam or hydropower turbine models and load controllers

A new Turbine-Governors category has been added to the Control library. You can use the blocks from this library to implement speed-governing steam or hydropower turbine models and load controllers:

- Controller LCFB1
- Governor Type 1
- Governor Type 3

Phasor Measurement Unit (PMU, Three-Phase) Block: Measure the positive sequence component of the frequency, phase, and amplitude of the input signal

The Phasor Measurement Unit (PMU, Three-Phase) block measures the frequency, phase, and amplitude of the positive sequence component of the input electrical signal in a three-phase system.

Programmable Signal Generator block: Generate a three-phase sinusoidal signal with programmable magnitude, phase, and frequency over time

The Programmable Signal Generator (Three-Phase) block generates a three-phase sinusoidal signal with programmable magnitude, phase, and frequency over time. You can specify one or two harmonics that can be overlapped on the signal fundamental frequency.

Moving Average (Variable Frequency) Block: Output the moving average value of an input signal of variable frequency

To filter higher-frequency signal components and smooth noisy signals, the Moving Average (Variable Frequency) block computes the moving average value of an input signal of variable frequency.

Sine Wave Generator (Three-Phase) Block: Generate a balanced three-phase sine wave

The Sine Wave Generator (Three-Phase) block uses the magnitude, phase shift, and frequency inputs to generate a balanced three-phase sine wave.

Stair Generator Block: Generate a stair pattern signal based on specified values

The Stair Generator block generates a custom stair signal based on the values specified in the **Time (s)** and **Amplitude** parameters.

Constant Power Load Block: Model a faultable constant power load for a DC supply

The Constant Power Load block implements a constant power load for a DC supply. The load behavior changes according to the voltage received from the DC supply.

You can model an electrical fault as an open circuit by setting the **Enable open circuit fault** parameter to Yes.

General Control Blocks Enhancement: Specify initialization options

You can now specify the initial conditions for these blocks:

- Filtered Derivative (Discrete or Continuous)
- Lead-Lag (Discrete or Continuous)
- Low-Pass Filter (Discrete or Continuous)
- Second-Order Low-Pass Filter (Discrete or Continuous)
- Washout (Discrete or Continuous)

To enable the **Initial state** parameter, set **Initialization** to Specify as parameter.

SPICE Conversion Assistant: Use derivative SPICE function

The SPICE Conversion Assistant now supports the derivative SPICE function, ddt. For more information, see [Converting a SPICE Netlist to Simscape Blocks](#).

SPICE Current Sources Blocks: Specify behavior of parallel conductance

You can now fully specify the behavior of the parallel conductance, *GMIN*, for these blocks:

- DC Current Source
- Exponential Current Source
- PCCCS
- PCCCS2

- Piecewise Linear Current Source
- Pulse Current Source
- PVCCS
- PVCCS2
- SFFM Current Source
- Sinusoidal Current Source

The value of the parallel conductance, $GMIN$, depends on whether your model contains an Environment Parameters block.

This table shows the possible values of $GMIN$:

Values of parallel conductance, $GMIN$

Include parallel conductance, $GMIN$		
Yes		No
Model contains Environment Parameters block	Model does not contain Environment Parameters block	Parallel conductance, $GMIN$, is not included and it does not affect the current source.
$GMIN$ is equal to the value of the $GMIN$ parameter in the Environment Parameters block.	$GMIN = 1e-12$ Ohm	

P-Channel MOSFET Block: Changes on order and location of s and d ports

The ports order and location of the surface-potential-based variant of the P-Channel MOSFET block are now consistent with those of its threshold-based variant.

Featured Examples

New examples in this release include:

- Solar PV System with MPPT Using Boost Converter — Design a boost converter to control the power output of a solar PV system by using maximum power point tracking technique.
- Stand-Alone Solar PV AC Power System with Battery Backup — Design a standalone PV AC power system with battery backup.
- Stand-Alone Solar PV DC Power System with Battery Backup — Design a standalone PV DC power system with battery backup.
- Single-Phase Grid-Connected Solar Photovoltaic System
- Three-Phase Grid-Connected Solar Photovoltaic System
- SPICE Conversion of a CMOS Voltage Comparator — Convert a SPICE implementation of a CMOS voltage comparator to a Simscape component and compare the results.

- Troubleshoot a SPICE Conversion of an IGBT Subcircuit — Troubleshoot the conversion of a SPICE insulated gate bipolar transistor (IGBT) subcircuit and convert it into an equivalent Simscape component.
- Five-Phase PMSM Velocity Control — Control the rotor angular velocity in an electrical-traction drive based on a five-phase permanent magnet synchronous machine (PMSM).
- Five-Phase PMSM Torque Control — Control the torque in an electrical-traction drive based on a five-phase permanent magnet synchronous machine (PMSM).
- Three-Phase PMLSM Drive — Control the position in a three-phase permanent magnet linear synchronous machine (PMLSM) drive.
- DC-DC Converter Model Fidelity Comparison — Compare simulation results using different levels of fidelity in DC-DC converters.
- Chopper Model Fidelity Comparison — Compare simulation results using different levels of fidelity in chopper converters.
- Calculate Performance Curves of BLDC — Calculate the peak torque, power, current, and efficiency values of a brushless DC (BLDC) motor for a given speed.
- Single-Phase PMSM Control — Control the rotor speed of an electrical drive based on a single-phase permanent magnet synchronous machine (PMSM).

Specialized Power Systems

Powergui Block: Use the discrete phasor solver type to discretize and solve electrical networks in phasor mode

The discrete phasor solver type uses simplified machine models where differential equations on the stator side are replaced with algebraic equations. These lower-order machine models produce simulation results similar to commercial stability software. The solver uses a robust solution method that allows you to eliminate machine parasitic loads and use Simulink Coder to generate code and simulate your model in real time.

PV Array Block: Use the robust discrete model to discretize PV Array blocks

The robust solver iterates to resolve the algebraic loop in internal diode models at a specified cell temperature. The robust solver is the recommended method for discretizing the PV Array blocks.

Power_flicker Function: Analyze the short-term and long-term flicker severity of an instantaneous flicker level signal shorter than 10 minutes

The new function allows you to specify a short-term observation interval that is shorter than the 10 minutes (600 seconds) default setting defined by the IEC standard. The short-term flicker severity and long-term flicker severity are computed for the specified observation interval period.

Featured Examples

New examples in this release include:

- 2-MW PV Farm Connected to a 25-kV Distribution System — Show how to connect PV Arrays to a distribution system using DC boost converters driven by Maximum Power Point Tracker (MPPT) controllers.
- PV Home On-Grid Solar System — Show the operation of a photovoltaic residential system connected to the electrical grid.

R2019b

Version: 7.2

New Features

Bug Fixes

Version History

Simscape Electrical

Load-Flow Analysis: Determine nodal voltages and real and reactive power flows in Simscape Electrical networks

You can now perform a load-flow analysis on models built using Simscape components by using new load-flow blocks and updated machine blocks. You can use the load-flow results to initialize dynamic AC simulation.

The new load-flow blocks are:

- Load Flow Source block — Model a swing, PV, or PQ bus, or a voltage with source impedance.
- Busbar block — Model a load-flow bus with up to four connections.

These machine blocks now have load-flow compatible AC steady-state initialization options:

- Induction Machine Squirrel Cage
- Induction Machine Wound Rotor
- Synchronous Machine Model 2.1
- Synchronous Machine Salient Pole
- Synchronous Machine Round Rotor

For examples that show how to perform a load-flow analysis for a Simscape electrical network, see:

- 2-Bus Load Flow
- IEEE 9-Bus Load Flow
- Induction Motor Initialization with Load Flow
- Synchronous Machine Initialization with Load Flow

Isothermal and Fade Models for Battery Blocks: Specify fade data using temperature-dependent or temperature-independent parameterization

The Battery and Battery (Table-Based) blocks now support advanced battery fade modeling through both temperature-independent and temperature-dependent tabulated parameterization. The blocks can also calculate the no-load voltage as a function of temperature.

Thermal Model Improvements for Semiconductors: Model external thermal dynamics using Cauer and Foster representations

You can now model a fully external thermal network by removing the internal thermal network of certain semiconductor blocks. You can also model thermal dynamics using both Cauer and Foster thermal representations.

Expanded Support for Power Dissipation Data: View power_dissipated simulation logging data for all machine blocks

You can now analyze data for a power_dissipated logging node for all induction, permanent magnet, reluctance, and synchronous machines in the Electromechanical library. For more information on logging, viewing, and analyzing power_dissipated data, see:

- About the Simscape Results Explorer (Simscape)
- Data Logging (Simscape)
- sscexplore
- ee_getEfficiency
- ee_getPowerLossSummary
- ee_getPowerLossTimeSeries

Average-Value PWM for Three-Phase Converter Block: Use averaged firing pulses for PWM-controlled power conversion

The Converter (Three-Phase) block now allows you to model an ideal three-phase converter that is controlled by either averaged PWM signals or modulation waveforms. To model an ideal three-phase converter that can accept averaged gate signals, in the **Switching Devices** settings for the block, set the **switching device** parameter to Averaged switch.

Japanese Language Localization Support: Block parameters, variables, and error messages are available in Japanese

Simscape Electrical is available in Japanese for Japanese-localized systems. For more information, see Internationalization.

Expanded Control Library: Speed up modeling by using prebuilt and documented algorithm components

Save time deriving, implementing, testing, debugging, and documenting algorithm models by using blocks from the expanded Control library. The blocks are Simulink blocks that are modular and decoupled, customizable, and compatible with Embedded Coder. Each block provides open access to the implementation source code.

To access the new blocks, in the Simulink Library Browser, select **Simscape > Electrical > Control**, and then:

- **Measurements** — To access the Total Harmonic Distortion block.
- **Observers** — To access the Resolver-to-Digital Converter and Quadrature Shaft Decoder blocks.
- **Pulse Width Modulation** — To access the PWM Generator (Vienna Rectifier) block.

Hall-Effect Rotary Encoder Block: Convert Hall-effects into voltage and phase angle or angular position

To model a 360° rotary position sensor, the Hall-Effect Rotary Encoder block uses four equally spaced Hall elements and a rotating magnet to generate four sinusoidal waveforms.

Passive Harmonic Filter (Three-Phase) Block: Improve power quality

To suppress system harmonic currents and decrease voltage distortion, the Passive Harmonic Filter (Three-Phase) block provides low-impedance paths for specified harmonics. At rated frequency, the passive shunt filters are capacitive. The capacitive filters provide reactive power that improves power factor.

Sequence Analyzer Block: Calculate magnitude and phase of the sequence of a three-phase signal

The Sequence Analyzer block outputs the magnitude and phase of the positive-sequence, negative-sequence, or zero-sequence components of a three-phase input signal.

Single-Phase and Three-Phase Fault Block Improvements: Limit fault duration time and specify both event-based and temporal faults for both fault blocks

The Fault block, a single-phase block, now allows you to limit the duration time for a temporal fault. To limit the fault duration, in the **Temporal Trigger** settings, change the value of the **Fault duration** parameter from `inf` to a nonnegative number.

The Fault (Three-Phase) block, replaces both the Enabled Fault and the Time-Based Fault. The merged block allows you to:

- Set externally triggered faults using either a specified threshold value or a physical input signal.
- Set temporally triggered faults.
- Reenter the unfaulted state for both externally and temporally triggered faults.
- View the fault state.

Version History

For information on issues related to block name changes, see “Version History” on page 6-9.

SPST and SPDT Relay Blocks: Model a controlled single-pole single-throw or double-throw relay with faults and delays

The SPST Relay and SPDT Relay blocks allow you to:

- Control the relay state using a physical signal input port or electrical conserving ports
- Output the relay state
- Delay the breaking and making of connections
- Introduce a behavioral or temporal fault that results in a stuck connection, an open circuit, or degraded contact resistance

The SPDT Relay block replaces the Relay.

Version History

For information on issues related to block name changes, see “Version History” on page 6-9.

Nonlinear Reluctance with Saturation: Include different levels of linearity and saturation in magnetic reluctance

The Nonlinear Reluctance block replaces the Reluctance with Hysteresis block. The Nonlinear Reluctance block provides these parameterization models:

- Linear reluctance.
- Reluctance with single saturation point.
- Reluctance (B-H curve).
- Nonlinear reluctance with hysteresis. For this implementation, the block uses the Jiles-Atherton model, as was provided by the Reluctance with Hysteresis block, with the new nonlinearity capability.

Version History

For information on issues related to block name changes, see “Version History” on page 6-9.

Three-Winding Nonlinear Transformer: Model varying levels of nonlinearity for a transformer with two secondary windings

The Three-Winding Nonlinear Transformer block allows you to model varying levels of nonlinearity, set initial current or flux for each winding, and include parasitic conductance. The Three-Winding Nonlinear Transformer block allows you to simulate circuit topologies that you cannot model using multiple Nonlinear Transformer blocks.

Zigzag-Delta-Wye Transformer block: Model a linear, nonideal zigzag-delta-wye transformer with a zigzag interconnect port

The Zigzag-Delta-Wye Transformer block replaces the Zigzag-Delta1-Wye Transformer and Zigzag-Delta11-Wye Transformer blocks. The new block provides these configuration options for the delta winding:

- Delta 1 o'clock
- Delta 11 o'clock

To access the winding interconnect port, in the **Main** settings, set the **Zigzag configuration** parameter to Accessible winding interconnects.

Version History

For information on issues related to block name changes, see “Version History” on page 6-9.

Delta-Wound PMSM and BLDC Motor Models: Increase motor speed range by applying the full voltage across all three phases

The BLDC, PMSM, and FEM-Parameterized PMSM blocks now support a delta-wound configuration. Due to the full utilization of terminal voltage, delta windings allow you to increase the constant power speed range and provide higher efficiency than wye windings.

To model a delta winding, in the block settings, set the **Winding type** parameter to Delta-wound.

Dielectric Relaxation Model for Capacitor Block: Model both resistive and dielectric losses

To represent both resistive and dielectric capacitance losses, the Capacitor block uses the Debye dielectric relaxation model. The Debye model considers a collection of noninteracting dipoles in the frequency domain. To use this frequency-dependent, lossy capacitor model, in the **Main** settings of the Capacitor block, set the **Capacitance** parameter to Dielectric relaxation (Debye).

H-Bridge Block Voltage Supply Ports: Model external DC power for a PWM-controlled H-bridge motor drive

You can now model DC voltage demand by exposing the internal supply rail connections of an H-Bridge block that simulates in PWM mode. To expose the power supply ports on an H-Bridge block, in the **Simulation Mode & Load Assumptions** settings, set the **Power supply** parameter to External.

The H-Bridge block also has a new parameter that determines the circumstances for applying the regenerative braking model when the **Simulation mode** parameter for the block is set to Averaged. You can now apply regenerative braking regardless of the voltage applied at the **REV** port for controlling block output polarity reversal, the H-Bridge. Alternatively, you can make the application of regenerative breaking dependent on the state of the **REV** port and the direction of current.

Harmonic Generation Time Limits for Voltage Source Blocks: Set start and stop times for generating source harmonics in the single and three-phase programmable voltage source blocks

For both the Programmable Voltage Source and Programmable Voltage Source (Three-Phase) blocks, you can now start generating source harmonics after the start of simulation and stop generating the harmonics before the end of simulation. To set time limits for generating source harmonics, in the block **Harmonics** settings, set **Source harmonics** to Generate harmonics, and then, specify simulation times for one or both of these parameters:

- **Start time**
- **Stop time**

Solenoid Block Orientation: Set the direction of force between the solenoid plunger and case

You can now specify the direction of action for the Solenoid block. Depending on the setting that you choose for the **Motion direction under excitation** parameter, a positive current from the electrical

+ port to the electrical - port can yield either a pulling or a pushing force from the mechanical **C** port to the mechanical **R** port.

When you change the direction of the force between the mechanical ports, the relative positions of the case and plunger on the icon of the Solenoid block change accordingly.

Voltage Plots for CMOS Logic Blocks: Visualize voltage data for integrated circuit models

These complementary metal-oxide-semiconductor (CMOS) logic blocks now create plots of the voltage at each electrical port:

- CMOS AND
- CMOS NAND
- CMOS OR
- CMOS NOT
- CMOS NOR
- CMOS XOR

To create voltage plots for a CMOS block, right-click the block and, from the context menu, select **Electrical > Basic Characteristics**.

Run-Time Tolerance Parameters: Modify fractional and random tolerance parameters for capacitance, inductance, and resistance without recompiling

You can now designate fractional and random tolerance parameters for these blocks as run-time tunable:

- Capacitor
- Diffusion Resistor
- Inductor
- Mutual Inductor
- Resistor

If you designate a parameter as run-time tunable and use a variable to define the parameter value, you can apply a value change between simulation runs without recompiling. For more information about run-time parameters for Simscape blocks, see Run-Time Parameters.

Block Name Changes and Mergers: Easily identify and configure modeling components

Block names were updated to better indicate the functionality and the electrical domain of the component. Merged blocks combine several model variants in a single block.

Additional functionality is available, as indicated, for some of the renamed and merged blocks.

Block Name Changes and Mergers

R2019a Block Name	R2019b Block Name	Merger	Additional Functionality
Enabled Fault	Fault (Three Phase)	Time-Based Fault	See “Single-Phase and Three-Phase Fault Block Improvements: Limit fault duration time and specify both event-based and temporal faults for both fault blocks” on page 6-4
Relay	SPDT Relay	Not applicable	See “SPST and SPDT Relay Blocks: Model a controlled single-pole single-throw or double-throw relay with faults and delays” on page 6-4
Reluctance with Hysteresis	Nonlinear Reluctance	Not applicable	See “Nonlinear Reluctance with Saturation: Include different levels of linearity and saturation in magnetic reluctance” on page 6-5
Time-Based Fault	Fault (Three Phase)	Enabled Fault	See “Single-Phase and Three-Phase Fault Block Improvements: Limit fault duration time and specify both event-based and temporal faults for both fault blocks” on page 6-4
Zigzag-Delta1-Wye Transformer	Zigzag-Delta-Wye Transformer	Zigzag-Delta11-Wye Transformer	See “Zigzag-Delta-Wye Transformer block: Model a linear, nonideal zigzag-delta-wye transformer with a zigzag interconnect port” on page 6-5
Zigzag-Delta11-Wye Transformer	Zigzag-Delta-Wye Transformer	Zigzag-Delta1-Wye Transformer	See “Zigzag-Delta-Wye Transformer block: Model a linear, nonideal zigzag-delta-wye transformer with a zigzag interconnect port” on page 6-5

Version History

If you reference a block within a library by an obsolete block name in MATLAB code, the code generates an error.

For example, the Relay block has been renamed the SPDT Relay block and moved from the **Simscape > Electrical > Switches & Breakers** library to the **Simscape > Electrical > Switches & Breakers > Relays** library.

Suppose an existing model named **modelName** contains an SPDT Relay block. When you open the model using the R2019a version of Simscape Electrical, the block is still named **Relay**. A function that calls the block in the model as an argument does not generate an error if it uses this syntax:

```
functionName('modelName/Relay','argument2',...,'argumentX')
```

However, the code generates an error if it uses a function to add an SPDT Relay from the Switches & Breakers library, or in any other way changes the SPDT Relay block that is in the library by using this syntax:

```
functionName('ee_lib/Switches & Breakers/Relay','argument2',...,'argumentX')
```

To prevent this type of error, use this syntax in your code:

```
functionName('ee_lib/Switches & Breakers/Relays/SPDT Relay','argument2',...,'argumentX')
```

If your code generates an error, it could contain an obsolete name as an argument in a function for adding or changing a block that is in a library. Some functions that are commonly used to manipulate blocks and block properties are:

- `add_block`
- `set_param`
- `get_param`

Featured Examples

New examples in this release include:

- 2-Bus Loadflow — Model a three-phase power system network that includes a swing bus and a PV bus.
- IEEE 9-Bus Loadflow — Model a 9-bus three-phase power system network based on an IEEE benchmark test case.
- Induction Motor Initialization with Loadflow — Initialize a three-phase induction motor using data from a load-flow analysis.
- Synchronous Machine Initialization with Loadflow — Initialize a synchronous machine using data from a load-flow analysis.
- Battery Parameter Extraction from Data — Specify Battery block parameters to fit data defined over different temperatures.
- Power Converter Model Fidelity Comparison — Model a three-phase matrix converter that drives a static load and draws unity power factor at the source.
- Three-Phase Matrix Converter — Model a three-phase matrix converter that drives a static load and draws unity power factor at the source.
- Photovoltaic Thermal (PV/T) Hybrid Solar Panel — Model a cogeneration system that contains electrical, thermal, and thermal-liquid components.

- Stepper Motor Pull-In Characteristics — Validate a model using manufacturer datasheets and pre-parameterized models.
- Clocked Reset-Dominant SR-Latch — Model a reset-dominant SR-Latch by using logic components.
- Clocked Set-Dominant SR-Latch — Model a set-dominant SR-Latch by using logic components.
- Earthing Effects with Unbalanced Load — See how various reference blocks affect the voltage and current of a connected network.

Specialized Power Systems

Expanded Automatic Advanced Settings Handling: Simplify discrete simulation settings for machine models by using the powergui block discrete solver

For some machine models, the powergui block discrete solver determines values that you would otherwise have to specify in the advanced simulation settings of the connected block. The simplified simulation settings can now be used with the Simplified Synchronous Machine block, the Permanent Magnet Synchronous Machine blocks, and the double squirrel-cage rotor-type model for the per-unit and SI Asynchronous Machine blocks.

To use automatic handling for advanced settings of machine blocks:

- 1 Open the settings for the attached powergui block.
- 2 In the **Solver** settings, for the **Simulation** parameter, select **Discrete**.
- 3 In **Preferences**, select the **Automatically handle the Discrete solver and Advanced tab solver settings of blocks** option.

Discrete Load and Transformer Blocks: Build models for discrete-time simulation using nonlinear and variable blocks

You can use the discrete solver capability of the powergui block to calculate results from models that contain one of these new blocks from the Elements library:

- Nonlinear Inductor
- Nonlinear Resistor
- Variable Capacitor
- Variable Inductor
- Variable-Ratio Transformer
- Variable Resistor

Pulse-Averaging and Interpolating PWM Blocks: Use advanced pulse generation techniques for high-precision motor control

The PWM Generator (Interpolation) and PWM Generator (Pulse Averaging) blocks control electronic converters using carrier-based unipolar pulse width modulation. Both blocks generate pulses for these converter types:

- Two-quadrant
- Full-bridge, single-phase
- Two-level, three-phase
- Three-level, three-phase

Featured Example

The On Load Tap Changer (OLTC) Regulating Transformer Using Variable-Ratio Transformer Blocks example shows how to regulate a positive-sequence voltage at a transmission network 25 kV bus by using an on-line tap changer (OLTC) transformer.

R2019a

Version: 7.1

New Features

Bug Fixes

Version History

Simscape Electrical

Dynamic Load (Three-Phase) Block: Model a four-quadrant dynamic load

The Dynamic Load (Three-Phase) block models a four-quadrant load that you can vary over the course of simulation. You specify the real and reactive power demand of the block using a physical signal input port.

Controlled Voltage Source (Three-Phase) Block: Apply an ideal controlled three-phase voltage to your network

The Controlled Voltage Source (Three-Phase) block maintains the specified voltage regardless of the load. You specify the reference voltage using a physical signal input port.

Current and Voltage Sensor (Three-Phase) Block: Measure current and voltage at three-phase electrical nodes

The Current and Voltage Sensor (Three-Phase) block represents an ideal three-phase voltage and current sensor. It measures voltage and current of a three-phase electrical node and outputs two three-element physical signal vectors.

PS Output Port for Fuse and Relay Blocks: Output block state as physical signal

Optional PS output ports on Fuse and Relay blocks let you output the block state as a physical signal. The port visibility is controlled by a new parameter in the block dialog box:

- Fuse block — The parameter name is **PS output for fuse state**. The default value is `Hidden`. When you set it to `Visible`, a PS output port **x** appears on the block icon. The port outputs 0 if the fuse is intact, and 1 if the fuse is blown.
- Relay block — The parameter name is **PS output for relay state**. The default value is `Hidden`. When you set it to `Visible`, a PS output port **x** appears on the block icon. The port outputs a vector of length two, the first element corresponding to the C-S1 connection and the second to the C-S2 connection. The elements are 1 if the corresponding connection is closed, and 0 otherwise.

Pressure Transducer Block Measurement Port: Select domain type for pressure measurement

The Pressure Transducer block turns a pressure measurement into voltage. The block now lets you measure pressure in a variety of domains. The new **Fluid port type** parameter has these options:

- Hydraulic
- Thermal Liquid
- Gas (default)
- Moist Air

- Two-Phase Fluid

Version History

In previous releases, the Pressure Transducer block had a pneumatic measurement port. Pneumatic blocks are no longer part of the Foundation library, but they are included in the Simscape product installation as an example custom library. The legacy Pressure Transducer block, with a pneumatic port, is now part of this custom library. From 2019a, use the Gas library for modeling pneumatic systems, and use the latest version of the Pressure Transducer block with a gas port. For more information, see Gas Domain and Block Library: Model gas systems with various levels of idealization (Simscape).

Earthing Transformer Block: Create a neutral point for your three-phase power network

The Earthing Transformer block is a three-phase coupler that provides your power system with a **neutral** connection for earthing.

Frequency-Dependent Overhead Line (Three-Phase) Block: Model a three-phase line over a wide range of frequencies

The Frequency-Dependent Overhead Line (Three-Phase) block generates frequency-dependent parameters that consider skin effect and earth return based on the simplified parameters that you specify. The block is suitable for power-system transient and harmonic analyses.

Programmable Voltage Source Blocks: Provide a single-phase or three-phase voltage with programmable magnitude, frequency, and phase shift to your network

The Programmable Voltage Source block models a single-phase AC voltage source with programmable magnitude, frequency, phase shift, and DC voltage offset. You can also include harmonics and noise in the voltage source.

The Programmable Voltage Source (Three-Phase) block models a three-phase voltage source with programmable AC magnitude, frequency, and phase shift. You can also include impedance and harmonics in the voltage source.

For both blocks, you can parameterize each programmable property individually using:

- A constant value
- A ramp
- A step
- Modulation

Alternatively, you can input a property using an external signal connected to a physical signal input port on the single-phase or three-phase programmable voltage source block.

Restructured Libraries: Easily discover blocks for building networks that have the right mix of performance and fidelity for your model-based design goals

Electronics and Mechatronics and **Power Systems**, two of the three top-level Simscape Electrical block libraries have been integrated. Some library names are changed to reflect the commonality of blocks that they now contain.

To open the top-level block library, at the MATLAB command prompt, enter `simscapeelectrical` or `ee_lib`.

Library Restructure

R2018b	R2019a
Electronics and Mechatronics <ul style="list-style-type: none"> • Actuators and Drivers • Integrated Circuits • Logic • Passive Devices • Switches • Semiconductor Devices • Sensors • Sources 	Connectors and References Control Electromechanical Integrated Circuits Semiconductors and Converters Passive Sensors and Transducers Sources
Power Systems <ul style="list-style-type: none"> • Connections • Control • Machines • Passive Devices • Semiconductors • Sensors • Sources • Switches and Breakers 	Switches and Breakers Utilities Additional Components
Specialized Power Systems	Specialized Power Systems

Version History

MATLAB code generates an error if a function argument includes:

- The previous name of a library
- The incorrect 'libraryName/block Name' combination

Block Name Changes and Mergers: Updated block names and merged blocks allow you to easily identify and configure model components

Block names have been updated to better indicate the functionality and the electrical domain, in terms of the number of phases, of the component.

Merged machine blocks allow you to easily switch between SI and per-unit parameterizations. Merged diode, transformer, switch, and reluctance machine blocks combine several model variants in a single block.

Block Name Changes and Mergers

Block Name Prior to R2019a	Block Name from R2019a	Merger
AC Cable	AC Cable (Three-Phase)	N/A
ASM Current Controller	Induction Machine Current Controller	N/A
ASM Direct Torque Control	Induction Machine Direct Torque Control	N/A
ASM Direct Torque Control with Space Vector Modulator	Induction Machine Direct Torque Control with Space Vector Modulator	N/A
ASM Field-Oriented Control	Induction Machine Field-Oriented Control	N/A
ASM Flux Observer	Induction Machine Flux Observer	N/A
ASM Scalar Control	Induction Machine Scalar Control	N/A
Asynchronous Machine Measurement	Induction Machine Measurement	N/A
Asynchronous Machine Squirrel Cage (fundamental)	Induction Machine Squirrel Cage	Asynchronous Machine Squirrel Cage (fundamental, SI)
Asynchronous Machine Squirrel Cage (fundamental, SI)	Induction Machine Squirrel Cage	Asynchronous Machine Squirrel Cage (fundamental)
Asynchronous Machine Wound Rotor (fundamental)	Induction Machine Wound Rotor	Asynchronous Machine Wound Rotor (fundamental, SI)
Asynchronous Machine Wound Rotor (fundamental, SI)	Induction Machine Wound Rotor	Asynchronous Machine Wound Rotor (fundamental)
Average-Value Inverter	Average-Value Inverter (Three-Phase)	N/A
Average-Value Rectifier	Average-Value Rectifier (Three-Phase)	N/A
Average-Value Voltage Source Converter	Average-Value Voltage Source Converter (Three-Phase)	N/A
Brushless DC Motor	BLDC	N/A
Commutation Diode	Diode	Diode Diode (Piecewise Linear)
Converter	Converter (Three-Phase)	N/A
Coupled Lines	Coupled Lines (Three-Phase)	N/A
Current Sensor	Current Sensor (Three-Phase)	N/A
Delta Reference	Delta Reference (Three-Phase)	N/A

Block Name Prior to R2019a	Block Name from R2019a	Merger
Delta11-Delta11-Wye Transformer	Three-Winding Transformer (Three-Phase)	Delta1-Delta1-Wye Transformer Wye-Delta11-Wye Transformer Wye-Delta1-Wye Transformer
Delta1-Delta1-Wye Transformer	Three-Winding Transformer (Three-Phase)	Delta11-Delta11-Wye Transformer Wye-Delta11-Wye Transformer Wye-Delta1-Wye Transformer
Delta-Delta Transformer	Two-Winding Transformer (Three-Phase)	Wye-Delta1 Transformer Wye-Delta11 Transformer Wye-Wye Transformer
Diode	Diode	Commutation Diode Diode (Piecewise Linear)
Five-Phase Switched Reluctance Machine	Switched Reluctance Machine (Multi-Phase)	Four-Phase Switched Reluctance Machine
Four-Phase Switched Reluctance Machine	Switched Reluctance Machine (Multi-Phase)	Five-Phase Switched Reluctance Machine
Diode (Piecewise Linear)	Diode	Commutation Diode Diode
Floating Neutral	Floating Neutral (Three-Phase)	N/A
Grounded Neutral	Grounded Neutral (Three-Phase)	N/A
Hybrid Excitation Synchronous Machine	Hybrid Excitation PMSM	N/A
IGBT	IGBT (Ideal, Switching)	N/A
Induction Motor	Simplified Induction Motor	N/A
Line Voltage Sensor	Line Voltage Sensor (Three-Phase)	N/A
Machine Mechanical Power (pu)	Machine Mechanical Power	Machine Mechanical Power (SI)
Machine Mechanical Power (SI)	Machine Mechanical Power	Machine Mechanical Power (pu)
MOSFET	MOSFET (Ideal, Switching)	N/A
Neutral Port	Neutral Port (Three-Phase)	N/A
Open Circuit	Open Circuit (Three-Phase)	N/A
Permanent Magnet Synchronous Motor	PMSM	N/A
Phase Voltage Sensor	Phase Voltage Sensor (Three-Phase)	N/A

Block Name Prior to R2019a	Block Name from R2019a	Merger
Piezo Linear Motor	Piezo Linear Actuator	N/A
Piezo Rotary Motor	Piezo Rotary Actuator	N/A
PWL Current Source	Piecewise Linear Current Source	N/A
PWL Voltage Source	Piecewise Linear Voltage Source	N/A
Rectifier	Rectifier (Three-Phase)	N/A
RLC	RLC (Three-Phase)	N/A
Servomotor	Simplified PMSM Drive	N/A
Single-Phase ASM Direct Torque Control	Induction Machine Direct Torque Control (Single-Phase)	N/A
Single-Phase ASM Field-Oriented Control	Induction Machine Field-Oriented Control (Single-Phase)	N/A
Single-Phase Asynchronous Machine (fundamental)	Induction Machine (Single-Phase)	Single-Phase Asynchronous Machine (fundamental, SI)
Single-Phase Asynchronous Machine (fundamental, SI)	Induction Machine (Single-Phase)	Single-Phase Asynchronous Machine (fundamental)
Single-Phase Circuit Breaker	Circuit Breaker	N/A
Single-Phase Circuit Breaker (with arc)	Circuit Breaker (with arc)	N/A
Single-Phase Permanent Magnet Synchronous Motor	PMSM (Single-Phase)	N/A
Single-Phase Switch	SPST Switch	SPST Switch
Single-Phase Two-Way Switch	SPDT Switch	SPDT Switch
SPDT Switch	SPDT Switch	Single-Phase Two-Way Switch
SPICE Environment Parameters	Environment Parameters	N/A
SPST Switch	SPST Switch	Single-Phase Switch
Switch	SPST Switch (Three-Phase)	N/A
Switched Reluctance Machine	Switched Reluctance Machine	Switched Reluctance Motor
Switched Reluctance Motor	Switched Reluctance Machine	Switched Reluctance Machine
Synchronous Machine Field Circuit (pu)	Synchronous Machine Field Circuit	Synchronous Machine Field Circuit (SI)
Synchronous Machine Field Circuit (SI)	Synchronous Machine Field Circuit	Synchronous Machine Field Circuit (pu)
Synchronous Machine Model 1.0 (SI parameters)	Synchronous Machine Model 1.0	N/A
Synchronous Machine Model 2.1 (fundamental)	Synchronous Machine Model 2.1	Synchronous Machine Model 2.1 (standard)
Synchronous Machine Model 2.1 (standard)	Synchronous Machine Model 2.1	Synchronous Machine Model 2.1 (fundamental)

Block Name Prior to R2019a	Block Name from R2019a	Merger
Synchronous Machine Round Rotor (fundamental)	Synchronous Machine Round Rotor	Synchronous Machine Round Rotor (standard)
Synchronous Machine Round Rotor (standard)	Synchronous Machine Round Rotor	Synchronous Machine Round Rotor (fundamental)
Synchronous Machine Salient Pole (fundamental)	Synchronous Machine Salient Pole	Synchronous Machine Salient Pole (standard)
Synchronous Machine Salient Pole (standard)	Synchronous Machine Salient Pole	Synchronous Machine Salient Pole (fundamental)
Three-Level Converter	Three-Level Converter (Three-Phase)	N/A
Three-Phase Bridge Cycloconverter Voltage Controller	Bridge Cycloconverter Voltage Controller (Three-Phase)	N/A
Three-Phase Hysteresis Current Controller	Hysteresis Current Controller (Three-Phase)	N/A
Three-Phase Power Measurement	Power Measurement (Three-Phase)	N/A
Three-Phase Sinusoidal Measurement (PLL)	Sinusoidal Measurement (PLL, Three-Phase)	N/A
Thyristor Rectifier Voltage Controller	Thyristor Rectifier Voltage Controller (Three-Phase)	N/A
Two-Way Switch	SPDT Switch (Three-Phase)	N/A
Wye-Delta1 Transformer	Two-Winding Transformer (Three-Phase)	Delta-Delta Transformer Wye-Delta11 Transformer Wye-Wye Transformer
Wye-Delta11 Transformer	Two-Winding Transformer (Three-Phase)	Delta-Delta Transformer Wye-Delta1 Transformer Wye-Wye Transformer
Wye-Delta11-Wye Transformer	Three-Winding Transformer (Three-Phase)	Delta11-Delta11-Wye Transformer Delta1-Delta1-Wye Transformer Wye-Delta1-Wye Transformer
Wye-Delta1-Wye Transformer	Three-Winding Transformer (Three-Phase)	Delta11-Delta11-Wye Transformer Delta1-Delta1-Wye Transformer Wye-Delta11-Wye Transformer

Block Name Prior to R2019a	Block Name from R2019a	Merger
Wye-Wye Transformer	Two-Winding Transformer (Three-Phase)	Delta-Delta Transformer Wye-Delta1 Transformer Wye-Delta11 Transformer

Version History

MATLAB code generates an error if it refers to a block within a library by an obsolete block name.

For example, the Single-Phase Switch block has been renamed as the SPST Switch block. The Single-Phase Switch block remains in the Switches & Breakers library.

Suppose an existing model named **modelName** contains a Single-Phase Switch block. When you open the model using the R2019a version of Simscape Electrical, the block is still named **Single-Phase Switch**. A function that calls the block in the model as an argument does not generate an error if it uses this syntax:

```
functionName('modelName/Single-Phase Switch','argument2',...,'argumentX')
```

However, code generates an error if it uses a function to add a Single-Phase Switch from the Switches & Breakers library, or in any other way changes the Single-Phase Switch block that is in the library by using this syntax:

```
functionName('Switches & Breakers/Single-Phase Switch','argument2',...,'argumentX')
```

If your code generates an error, it could contain an obsolete name as an argument in a function for adding or changing a block that is in a library. Some functions that are commonly used to manipulate blocks and block properties are:

- add_block
- set_param
- get_param

Function Name Changes: Functions renamed to ee_functionName

Simscape Electrical function names are updated from pe_functionName and elec_functionName to ee_functionName. The pe and elec functions are still available, but are set for removal in a future release.

Function Name Changes

Prior to R2019a	From R2019a
<code>elec_calculateFluxPartialDerivatives</code>	<code>ee_calculateFluxPartialDerivatives</code>
<code>elec_generateIdealPMSMfluxData</code>	<code>ee_generateIdealPMSMfluxData</code>
<code>elec_getEfficiency</code>	<code>ee_getEfficiency</code>
<code>elec_getNodeDvDtSummary</code>	<code>ee_getNodeDvDtSummary</code>
<code>elec_getNodeDvDtTimeSeries</code>	<code>ee_getNodeDvDtTimeSeries</code>
<code>elec_getPowerLossSummary</code>	<code>ee_getPowerLossSummary</code>
<code>elec_getPowerLossTimeSeries</code>	<code>ee_getPowerLossTimeSeries</code>
<code>pe_calculateThdPercent</code>	<code>ee_calculateThdPercent</code>
<code>pe_getEfficiency</code>	<code>ee_getEfficiency</code>
<code>pe_getHarmonics</code>	<code>ee_getHarmonics</code>
<code>pe_getPowerLossSummary</code>	<code>ee_getPowerLossSummary</code>
<code>pe_getPowerLossTimeSeries</code>	<code>ee_getPowerLossTimeSeries</code>
<code>pe_plotHarmonics</code>	<code>ee_plotHarmonics</code>

Example Model Name Changes: Example models renamed to `ee_modelName`

Simscape Electrical example model names are updated from `pe_modelName` and `elec_modelName` to `ee_modelName`. For each model, the simulation data log (simlog) is renamed to agree with the new model name.

Version History

MATLAB code generates an error if it:

- Uses an obsolete model name as an argument for any of function that loads, opens, simulates, or in any way changes the model
- References the obsolete simlog name to extract or explore simulation data.

Three-Phase Electrical Domain Changes: The three-phase electrical domain is renamed and moved to the Simscape foundation package

The three-phase electrical domain is renamed from `pe.electrical.three_phase.electrical` to `foundation.electrical.three_phase`. The domain is moved from Simscape Electrical package to the Simscape package.

Version History

A custom Simscape language file that uses `pe.electrical.three_phase.electrical` can not connect to any built-in three-phase blocks until the node domain is changed to `foundation.electrical.three_phase` and the component is rebuilt or updated.

Featured Examples

New examples in this release are:

- Frequency-Dependent Transmission Line— See how a frequency-dependent transmission line model compares to a classic pi-section transmission line model.
- Import IPMSM Flux Linkage Data from Motor-CAD — Import a motor design from Motor-CAD into a Simscape™ simulation.
- Load-Side Converter Control — Control RMS voltage in a load-side converter.
- PMSM Position Control — Control rotor position in a PMSM-based electrical drive.
- Power Factor Correction for CCM Boost Converter — Correct power factor by using a power factor correction (PFC) preconverter.
- Solar Panel Parameterization Validation — Validate a datasheet-parameterized solar array using current and power output data.
- Three-Phase PMSM Traction Drive — Tune a PID controller for an FEM-Parameterized PMSM.
- Vienna Rectifier Control — Control a Vienna rectifier using space-vector modulation.

Specialized Power Systems

Robust Discrete Solver: New powergui option simplifies advanced simulation settings

The robust discrete solver lets you simulate models that contain certain machine and transformer blocks without adding a parasitic load. The solver calculates and specifies the values for the advanced simulation settings.

To use the robust discrete solver option, select these settings for the powergui block:

- 1 In the **Solver** settings, for the **Simulation** parameter, select **Discrete**.
- 2 In the **Preferences** settings, select the **Automatically handle the Discrete solver and Advanced tab solver settings of blocks** option.

CCCV Battery Charger Block: Model a constant-current constant-voltage battery charger device

The CCCV Battery Charger block implements a generic smart battery charger. The model allows you to specify temperatures that effect the charge rate of the battery.

Autotuning Option for Electric Drive Blocks: AC3, AC4, AC6, and AC7 electric drive blocks can calculate PID control parameters

These electric drive blocks can determine the proportional and integral gain of the speed and flux controller from response time, damping, and bandwidth ratio of the desired loop specifications.

- Field-Oriented Control Induction Motor Drive
- DTC Induction Motor Drive
- PM Synchronous Motor Drive
- Brushless DC Motor Drive

To use the autotuning option, in the electric drive **Controller** settings:

- 1 Click **Show Autotuning Control**.
- 2 Enter the speed loop specifications.
- 3 Click **Calculate PI regulators gains**.

Sensorless Speed Estimation for AC Drive Blocks: AC3 and AC4 electric drive blocks can estimate rotational velocity

Specify a sensorless speed estimation measurement for the Field-Oriented Control Induction Motor Drive block or the DTC Induction Motor Drive block. The drive blocks estimate the motor speed from terminal voltages and currents by applying the model referencing adaptive system technique.

To use speed-estimation option, in the drive block **Controller** settings, select **Set sensorless**.

Independent Simulation Type Selection: powergui solver settings are not inherited from Simulink automatic solver settings

In R2019a, control of the settings for powergui blocks is independent from the Simulink solver settings for the model. When you select **auto (Automatic solver selection)**, Simulink no longer manages or disables the solver settings for the powergui blocks in that model.

Version History

If you open an existing model that has the Simulink **auto (Automatic solver selection)** solver mode selected, the powergui settings do not automatically revert to the values that you had selected before you opted to use the Simulink **auto (Automatic solver selection)** solver mode. However, you can manually change the powergui settings to the values that you selected before you opted to use the Simulink **auto (Automatic solver selection)** solver mode.

R2018b

Version: 7.0

New Features

Bug Fixes

Version History

Simscape Electronics and Simscape Power Systems combined into Simscape Electrical

Simscape Electrical is a new product that includes the modeling, simulation, and analysis technologies previously contained in Simscape Electronics (formerly SimElectronics[®]) and Simscape Power Systems (formerly SimPowerSystems[™]).

Find the top-level Simscape Electrical block library, now named **Electrical**, under **Simscape** in the Simulink Library Browser. It contains three sublibraries: **Electronics and Mechatronics, Power Systems**, and **Specialized Power Systems**. To open the top level-block library, at the MATLAB command prompt, enter `simscapeelectrical`.

To access release notes for prior releases of Simscape Electronics and Simscape Power Systems, see:

- [Simscape Electronics Release Notes](#)
- [Simscape Power Systems Release Notes](#)

Electronics and Mechatronics

SPICE Conversion Assistant: Convert SPICE models to Simscape components

The new SPICE conversion assistant allows you to convert SPICE netlist files into equivalent Simscape component files. For more information, see [Converting a SPICE Netlist to Simscape Blocks](#).

FEM-Parameterized PMSM Flux Parameterization: Specify flux linkage in polar or Cartesian coordinates

The FEM-Parameterized PMSM block now has additional parameterization options:

- You can use polar coordinates, rather than i_d and i_q , to tabulate fluxes
- The block accepts flux tabulation options used by Motor-CAD, as well as tabulated iron losses

Servomotor Block Loss Parameterization: Tabulate losses and efficiency as a function of speed, torque, and DC supply voltage

The Servomotor block now has additional parameterization options that let you tabulate losses and efficiency as a function of speed, torque, and DC supply voltage. Previously, you could tabulate losses and efficiency only as a function of speed and torque. See the block reference page for details.

Battery Characteristics Visualization: Plot voltage-charge characteristics for battery model parameter values

A new quick plot feature lets you visualize the voltage-charge characteristics for the battery model parameter values. To plot the characteristics, right-click a Battery or Battery (Table-Based) block and, from the context menu, select **Electrical > Basic characteristics**. The software automatically computes a set of bias conditions, based on the block parameter values, and opens a figure window containing a plot of no-load voltage versus the state-of-charge (SOC) for the block. For more information, see [Plot Basic Characteristics for Battery Blocks](#).

Peltier Device Block: Model conversion between electrical and thermal energy

The Peltier Device is a new block in the Sources library that lets you convert electrical energy into thermal energy and vice-versa.

elec_getNodeDvDtSummary and elec_getNodeDvDtTimeSeries Functions: Calculate derivatives of terminal voltages with respect to time

Checking terminal voltage derivatives with respect to time (dv/dt) helps ensure that all devices operate within their safe operating areas. Two new functions let you calculate voltage derivatives, based on logged simulation data, for block ports (terminals) that belong to the electrical domain:

- The `elec_getNodeDvDtSummary` function returns a three-column MATLAB table. The first column lists the applicable logging nodes, the second column lists the corresponding electrical terminals, and the third column lists the maximum absolute value of dv/dt for each terminal, in V/s.
- The `elec_getNodeDvDtTimeSeries` function returns a four-column MATLAB table. The first column lists the applicable logging nodes and the second column lists the corresponding electrical terminals. The third and fourth columns list the time-series data of terminal voltages, in V, and corresponding dv/dt values, in V/s, for these electrical terminals.

To use these functions, you have to enable simulation data logging and run the simulation. For more information, see [Data Logging](#).

Featured Examples

New examples in this release are:

- Use of Peltier Device as Thermoelectric Cooler
- Conducted Emission of a Buck Converter

Power Systems

Single-Phase Permanent Magnet Synchronous Motor Block: Model a single-phase PMSM with a squirrel-cage rotor

You can parameterize the Single-Phase Permanent Magnet Synchronous Motor (PMSM) block by specifying the flux linkage or the back electromotive force (BEMF) constant.

Multiphase Switched Reluctance Machines: Model low-current four-phase and five-phase SRMs

The Four-Phase Switched Reluctance Machine and Five-Phase Switched Reluctance Machine blocks represent electric motors that are driven by magnetic-reluctance torque. As is typical for reluctance machines, power is delivered to the stator rather than to the rotor.

You can parameterize either switched reluctance machine (SRM) block by specifying the saturated flux linkage, aligned inductance, and unaligned inductance or by specifying the current and angle vectors and the flux linkage matrix.

For examples that include these multiphase switched reluctance machines, see:

- Four-Phase Switched Reluctance Machine Control
- Five-Phase Switched Reluctance Machine Control

One-Quadrant Chopper Block: Convert from fixed to variable DC voltage bidirectionally

The One-Quadrant Chopper block is capable of bidirectional power flow between two connected electrical networks.

For an example that includes the One-Quadrant Chopper block, see One-Quadrant Chopper Control.

Average-Value Converters: Model DC-DC semiconductor converters that are suitable for real-time simulation

The Average-Value Chopper and Average-Value DC-DC Converter blocks allow you to convert DC voltages without switching. The relatively low computational cost of average-value converters makes them more suitable for real-time simulation than high-frequency semiconductor converter models.

For examples that include these average-value converter blocks, see:

- Average-Value Chopper Control
- Average-Value DC-DC Converter Control

Run-Time Parameter Support for All Machine Blocks: Speed up simulation tasks and modify component parameter values without regenerating C code

You can now modify certain block parameters for all electric machine blocks between simulation runs without regenerating C code or triggering the diagram update.

For more information, see Run-Time Parameters.

Single-Phase Asynchronous Machine Blocks: Model a single-phase ASM using fundamental or fundamental SI parameterization

Both the Single-Phase Asynchronous Machine (fundamental) and Single-Phase Asynchronous Machine (fundamental, SI) blocks represent a single-phase asynchronous machine with a squirrel cage rotor that you parameterize using fundamental per-unit or SI parameters, respectively. Choose between these four variants for either block:

- Split-phase
- Capacitor-start
- Capacitor-start-capacitor-run
- Main and auxiliary windings

For examples that include these single-phase ASM blocks, see:

- Single-Phase Asynchronous Machine Direct Torque Control
- Single-Phase Asynchronous Machine Field-Oriented Control

Asynchronous Machine Saturation: Include magnetic saturation in three-phase ASMs using open circuit lookup tables

Model magnetic saturation using lookup tables of per-unit no-load stator current saturation and the per-unit terminal voltage saturation data for these asynchronous machine (ASM) blocks:

- Asynchronous Machine Squirrel Cage (fundamental)
- Asynchronous Machine Squirrel Cage (fundamental, SI)
- Asynchronous Machine Wound Rotor (fundamental)
- Asynchronous Machine Wound Rotor (fundamental, SI)

To plot saturation data for an ASM block in a MATLAB figure window, right-click the block, select the **Electrical** menu on the block context menu, and then choose one of these actions:

- **Plot Open-Circuit Saturation** — Plots terminal voltage versus no-load stator current. The plot contains three traces:
 - Unsaturated
 - Saturated
 - Derived
- **Plot Saturation Factor** — Plots saturation factor applied to magnetic inductance versus magnetic flux linkage.

- **Plot Saturated Inductance** — Plots magnetizing inductance versus per-unit magnetic flux linkage.

If you use the saturation mode, the **Magnetizing inductance, L_m (pu)** parameter (for fundamental blocks) or the **Magnetizing reactance, X_m** parameter (for fundamental, SI blocks) is hidden because the block computes the value based on the saturation curve.

Double-Squirrel Cage Asynchronous Machines: Specify a single or double cage for ASM squirrel cage blocks

You can now specify a single or double cage for these Asynchronous Machine (ASM) Squirrel Cage blocks.

- Asynchronous Machine Squirrel Cage (fundamental)
- Asynchronous Machine Squirrel Cage (fundamental, SI)

When you model a double cage, you can specify separate referred rotor resistance and leakage inductance for each cage.

Synchronous Machine Rotor Angle: Define the rotor axis alignment for synchronous machine blocks

You can now define the reference point for the rotor angle measurement in these synchronous machine blocks:

- Synchronous Machine Model 2.1 (fundamental)
- Synchronous Machine Model 2.1 (standard)
- Synchronous Machine Round Rotor (fundamental)
- Synchronous Machine Round Rotor (standard)
- Synchronous Machine Salient Pole (fundamental)
- Synchronous Machine Salient Pole (standard)

The default value for the **Rotor angle definition** is Angle between the a-phase magnetic axis and the d-axis. When you select this value, the rotor *d*-axis and stator *a*-phase magnetic axis are aligned when the rotor angle is zero. The other value you can choose for this parameter is Angle between the a-phase magnetic axis and the q-axis. When you select this value, the rotor *q*-axis and stator *a*-phase magnetic axis are aligned when the rotor angle is zero.

Synchronous Machine Block Accuracy Improvement: SM blocks that use standard parameters and the SM field circuit block return more accurate results

Due to improved mapping of mask parameters to model coefficients, in certain configurations, the simulation results from models that contain these synchronous machine blocks are more accurate in R2018b:

- Synchronous Machine Model 2.1 (standard)
- Synchronous Machine Round Rotor (standard)

- Synchronous Machine Salient Pole (standard)
- Synchronous Machine Field Circuit (pu)

Machine Plotting and Display Options: Perform plotting and display actions using the Electrical menu

In previous releases, you could perform plotting and display actions for machine blocks using the **Power Systems** menu on the block context menu. Now, use the **Electrical** menu on the block context menu to perform those same actions. For more information, see Plotting and Display Options for Asynchronous and Synchronous Machines.

Controlled Current Source (Three-Phase) Block: Control the current loop of a cascading control system

Use the Controlled Current Source (Three-Phase) block to control the output current of a current loop in a cascading control system or to represent a low-fidelity converter without modeling switches.

Expanded Control Library: Speed up modeling by using prebuilt and documented algorithm components

Save time deriving, implementing, testing, debugging, and documenting algorithm models by using blocks from the expanded Control library. The blocks are Simulink blocks that are modular and decoupled, customizable, and compatible with Embedded Coder. Each block provides open access to the implementation source code.

To access the new blocks in the Simulink Library Browser, select, **Electrical > Power Systems > Control**, and then:

- **ASM Control** — For these single-phase ASM control blocks:
 - Single-Phase ASM Direct Torque Control — For an example that includes this block, see Single-Phase Asynchronous Machine Direct Torque Control.
 - Single-Phase ASM Field-Oriented Control — For an example that includes this block, see Single-Phase Asynchronous Machine Field-Oriented Control.
- **SM Control** — For the SM PSS1A block, which models a power system stabilizer with parameters that can account for the low-frequency effect of high-frequency torsional filters or shape the gain and phase characteristics of the stabilizer.
- **General Control** — For these control blocks:
 - Change Detector — Detect a change in the Boolean input.
 - Counter — Implement a fixed-time-period counter.
 - Fourier Analysis — Obtain the magnitude and angle of the input signal for a specified harmonic.
 - Monostable Flip-Flop — Generate a single pulse of a specified duration in response to an external trigger.
 - Moving Average — Calculate the average value of the input over a moving window.
 - On-Off Delay — Add a time delay in response to a detected transition in the input signal.

- Second-Order Filter — Implement a low-pass, high-pass, band-pass, or band-stop second-order filter.
- Second-Order Low-Pass Filter (Discrete or Continuous) — Implement an IEEE Std 421.5 second-order low-pass filter.
- Serial-In Parallel-Out Shift Register — Obtain a vector of last N samples of the input signals.
- Set-Reset Flip-Flop — Implement a bistable.
- Signal Sample and Hold — Sample and hold the input signal.
- Variable-Frequency Second-Order Filter — Implement a low-pass, high-pass, band-pass or band-stop second-order filter with variable frequency.

Battery Characteristics Visualization: Plot voltage-charge characteristics for battery model parameter values

A new quick-plot feature lets you visualize the voltage-charge characteristics for the battery model parameter values. To plot the characteristics, right-click a Battery or Battery (Table-Based) block and, from the context menu, select **Electrical > Basic characteristics**. The software automatically computes a set of bias conditions based on the block parameter values, and opens a figure window containing a plot of no-load voltage versus the state-of-charge (SOC) for the block.

Block Name Changes: Blocks names disambiguated from identically named blocks in the Simscape Electrical Electronics and Mechatronics library

Blocks names that are identical to the names of blocks in the Simscape Electrical Electronics and Mechatronics libraries are changed in R2018b.

Block Name Changes

Library Path	Name Prior to R2018b	New Name
Simscape > Electrical > Power Systems > Passive Devices > Fundamental Components	Transmission Line	Transmission Line (Three-Phase)
Simscape > Electrical > Power Systems > Semiconductors > Fundamental Components	Diode	Diode (Piecewise Linear)
	Thyristor	Thyristor (Piecewise Linear)
Simscape > Electrical > Power Systems > Sources	Current Source	Current Source (Three-Phase)
	Voltage Source	Voltage Source (Three-Phase)

Version History

To prevent a script that contains the previous name of a block from producing an error, update the block names in your code.

HDL Code Generation from Simscape Electrical Power Systems Models: Convert models to HDL code for simulation on FPGA devices

If you have an HDL Coder™ license, you can use the Simscape HDL Advisor advisor to deploy a linear or switched linear plant model developed using Simscape Electrical blocks to a target FPGA. To generate code using the Simscape HDL Advisor:

- 1** Configure the Power Systems model for conversion.
- 2** Convert the Power Systems model to an HDL code-compatible implementation model.
- 3** Generate HDL code from the implementation model.

For more information, see Real-Time Simulation.

Featured Examples

New examples in this release are:

- Average-Value Chopper Control
- Average-Value DC-DC Converter Control
- DC Motor Control (Lead-Lag)
- DC Motor Control (RST)
- DC Motor Control (Smith Predictor)
- DC Motor Control (State-Feedback and Observer)
- Five-Phase Switched Reluctance Machine Control
- Four-Phase Switched Reluctance Machine Control
- IPMSM Outer Loop Controller Evaluation
- One-Quadrant Chopper Control
- PMSM Parameterization from Datasheet
- PMSM Parameterization from Measurements
- Single-Phase Asynchronous Machine Direct Torque Control
- Single-Phase Asynchronous Machine Field-Oriented Control
- Vienna Rectifier

Specialized Power Systems

SVPWM Generator (3-Level) Block: Generate pulses for three-phase three-level neutral-point-clamped converters

The SVPWM Generator (3-Level) block outputs 12 pulses that it generates using the space vector pulse width modulation (SVPWM) technique. A proportional regulator controls the neutral point voltage deviation of the NPC converter based on the DC voltage references and the DC current flowing in or out of the DC link.

Featured Example

The Three-Level NPC Inverter Using Space-Vector PWM with Neutral-Point Voltage Control example shows how to operate a 2-MVA, 3-Level NPC inverter using the Space Vector Pulse-Width-Modulation (SVPWM) technique with neutral point voltage control.

