Simscape™ Battery™ Release Notes

MATLAB®&SIMULINK®

MathWorks®

How to Contact MathWorks

Latest news:	`www.mathworks.com`
Sales and services:	`www.mathworks.com/sales_and_services`
User community:	`www.mathworks.com/matlabcentral`
Technical support:	`www.mathworks.com/support/contact_us`
Phone:	508-647-7000

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

Simscape™ Battery™ Release Notes

Trademarks

Patents

Contents

R2023a

R2022b

1

R2023a

Battery Builder App: Design battery systems interactively and generate dynamic models

Use the **Battery Builder** app to create Simscape™ battery models by interactively creating, modifying, and visualizing MATLAB battery objects.

With the **Battery Builder** app, you can:

- Import existing battery objects from your workspace or a MAT file.
- Create new `Cell`, `ParallelAssembly`, `Module`, `ModuleAssembly`, and `Pack` objects.
- Generate a 3-D plot of the battery object, edit the plotting options, and export the plot to a FIG file.
- Inspect the hierarchy of a battery object and visualize its subcomponents.
- Edit the properties of a battery object, such as geometrical data and thermal boundary conditions.
- Export the objects you create in the app to your workspace or a MAT file.
- Create a Simscape battery model from an object.

Cell and CellModelBlock Objects Enhancement: Specify a custom block for the cell model block component

The `Cell` object now supports custom blocks for the cell model block component other than the Battery (Table-Based) block. To use your own custom block, set the `CellModelBlockPath` property of the associated `CellModelBlock` object to the library path of your custom block.

To use a custom block, it must meet all these requirements:

- The custom block must provide exactly two electrical conserving ports named **n** and **p**.
- All variables must be scalar-valued.
- The custom block cannot contain a variable or intermediate variable called *power_dissipated*.

You can connect the custom block to any number of thermal nodes. The software connects all nodes in parallel.

Cell Object Enhancement: Model a battery cell from built-in parameterizations

You can now parameterize the `Cell` object with built-in parameterizations. Use this pre-parameterization data to set up the object to represent components from specific suppliers. The parameterizations of these batteries match the discharge curves in the manufacturer data sheets.

To parameterize the `Cell` object with pre-parameterized data, specify the `ParameterizationManufacturer` and `ParameterizationPartNumber` properties. You can parameterize the `Cell` object only if its cell model block component is the Battery (Table-Based) block.

Cell Object Enhancement: Specify the capacity and energy of a battery cell

Use the `Capacity` and `Energy` properties of the `Cell` object to specify the capacity and energy of a battery cell, respectively.

As a result of these new properties, the `ParallelAssembly`, `Module`, `ModuleAssembly`, and `Pack` objects now contain two new read-only properties, `CumulativeCellCapacity` and `CumulativeCellEnergy`, which represent the cumulative cell capacity and the cumulative cell energy of the battery, respectively.

Battery Pack Modeling Objects and Blocks Enhancement: Model a cooling plate at the ModuleAssembly and Pack levels

You can now model a cooling plate component directly at the `ModuleAssembly` or `Pack` level by setting the `CoolingPlate` property to either `"Top"` or `"Bottom"`.

Moreover, inside the `ParallelAssembly`, `Module`, `ModuleAssembly`, and `Pack` objects, you can now use the new `CoolingPlateBlockPath` property to specify which cooling plate block to assign to the object at the boundary defined by the `CoolingPlate` property.

As a result of these changes, the ModuleAssembly (Generated Block) and Pack (Generated Block) blocks have these new ports:

- **CP_out** — Vector that contains the plate temperature, the fluid pressure drop, and the fluid temperature change of each cooling plate in the model.
- **Fluid_In** — Fluid that comes into each cooling plate.
- **Fluid_Out** — Fluid that comes out of each cooling plate.
- **CPB** — Thermal conserving port for connecting the bottom side of the module assembly with a cooling plate. This port is available only in the ModuleAssembly generated block.
- **CPT** — Thermal conserving port for connecting the top side of the module assembly with a cooling plate. This port is available only in the ModuleAssembly generated block.

Battery Pack Modeling Objects and Blocks Enhancement: Model intercell heat transfer

You can now model direct cell-to-cell heat exchange. Direct cell-to-cell heat exchange is important when you want to simulate more detailed thermal management strategies or thermal propagation scenarios where the intercell heat transfer happens at faster rates than the ambient or coolant rates.

This enhancement comprises:

- Two new properties in the `ParallelAssembly`, `Module`, `ModuleAssembly`, and `Pack` objects:
  - `InterCellThermalPath` — Option to use thermal resistance blocks to represent a linear cell-to-cell heat transfer path between adjacent battery cells
  - `InterCellRadiativeThermalPath` — Option to use thermal resistance blocks to represent a radiative cell-to-cell heat transfer path between adjacent battery cells
- New read-only properties in the `ParallelAssembly` and `Module` objects that show the internal thermal connections and mapping of cells and parallel assemblies:

  - `NumInterCellThermalConnections` — Number of internal thermal connections between the cells.
  - `NumInterParallelAssemblyThermalConnections` — Number of internal thermal connections between the parallel assemblies. This property is available only in the `Module` object.
  - `InterCellConnectionsMapping` — 2-D map of the internal connections between the cells.
  - `InterParallelAssemblyConnectionsMapping` — 2-D map of the internal connections between the parallel assemblies. This property is available only in the `Module` object.
  - `InterParallelAssemblyCellConnectionsMapping` — 2-D map of the internal connections between the battery cell models inside each parallel assembly of the module and the intercell heat transfer resistors. This property is available only in the `Module` object.
  - `Layout` — 2-D distribution of subcomponents or cells within the parent.
- New thermal parameters in the ParallelAssembly and Module generated blocks:
  - **Inter-cell thermal path resistance** — Resistance of the thermal path between the cells inside the parallel assembly or module.
  - **Inter-parallel assembly thermal path resistance** — Resistance of the thermal path between the parallel assemblies inside the module. This parameter is available only in the Module generated block.
  - **Inter-cell radiation heat transfer area** — Area of the radiation heat transfer between the cells inside the parallel assembly or module.
  - **Inter-cell radiation heat transfer coefficient** — Coefficient of the radiation heat transfer between the cells inside the parallel assembly or module.
  - **Inter-parallel assembly area for radiation heat transfer** — Area of the radiation heat transfer between the parallel assemblies inside the module. This parameter is available only in the Module generated block.
  - **Inter-parallel assembly coefficient for radiation heat transfer** — Coefficient of the radiation heat transfer between the parallel assemblies inside the module. This parameter is available only in the Module generated block.

Battery Management System Blocks Enhancement: Specify single data type for all inputs and parameters

These blocks in the **Battery Management System** library now support single-precision floating-point simulation in addition to the existing double-precision floating-point simulation support:

- Passive Cell Balancing
- Battery CC-CV
- Battery Charging Current Limit
- Battery Discharging Current Limit
- SOC Estimator (Adaptive Kalman Filter)
- SOC Estimator (Coulomb Counting)
- SOC Estimator (Coulomb Counting, Variable Capacity)
- SOC Estimator (Kalman Filter)
- SOH Estimator

- Fault Qualification
- Battery Coolant Control
- Battery Heater Control

Note To enable single-precision-floating point simulation, the data type of all inputs and parameters, except for the **Sample time (-1 for inherited)** parameter, must be `single`.

Note Continuous-time implementation of these blocks works only in double-precision floating-point simulation. If you provide single-precision floating-point parameters and inputs, these blocks cast them to double-precision floating-point values to prevent errors.

SOC Estimator (Coulomb Counting, Variable Capacity) Block: Estimate state of charge by using Coulomb counting and variable capacity

Use the SOC Estimator (Coulomb Counting, Variable Capacity) block to estimate the state of charge (SOC) of a battery by using the Coulomb counting method and the cell capacity of the battery. The block calculates the SOC by dividing the accumulated charge by the value of the cell capacity. The block then calculates the accumulated charge by integrating the battery current.

Battery CC-CV Block Enhancement: Specify the tracking coefficient of a proportional integral (PI) controller

You can now specify the tracking coefficient K_t of the Battery CC-CV block.

The block feeds the difference between the tracking signal and the controller output back to the integrator input with a gain value K_t. Use this parameter to specify the gain in that feedback loop.

Signal tracking has several applications, including bumpless control transfer and avoiding windup in multiloop control structures.

Featured Examples

New examples in this release include:

- “Get Started with Battery Builder App” — This example shows how to use the Battery Builder app to interactively create a battery pack with thermal effects and build a Simscape™ model that you can use as a starting point for your simulations.
- “Battery Cell Characterization for Electric Vehicles” — This example uses test methods to characterize a battery cell for electric vehicle applications.
- “Analyze Battery Spatial Temperature Variation During Fast Charge” — This example shows how the temperature gradient over the cell surface varies during the fast charging of a battery.
- “Build Model of Battery Module with Inter-Cell Heat Exchange” — This example shows how to create and build a Simscape system model of a battery module with inter-cell heat exchange in Simscape Battery.
- “Build Model of Battery Module Assembly with Multi-Module Cooling Plate” — This example shows how to create and build a Simscape system model of a module assembly with a multi-module cooling plate by using Simscape Battery.

Functionality being removed or changed

The Modeling Option parameter of Battery block is replaced

Behavior change

The **Modeling option** parameter of the Battery block is removed and replaced with the **Expose charge measurement port** and **Thermal port** parameters. To expose the internal charge output port **q**, set the **Expose charge measurement port** parameter to `Yes`. To expose the thermal port **H** and simulate thermal effects, set the **Thermal port** parameter to `Model`.

As a result of this change, if you created a custom component by using the Battery block in an earlier release, you must update the custom component accordingly.

```
For batteryecm.battery_thermal:
charge_measurement_port=simscape.enum.thermaleffects.model
```

```
For batteryecm.battery_instrumented:
charge_measurement_port=simscape.enum.battery.enable.yes
```

```
For batteryecm.battery_thermal_instrumented:
charge_measurement_port=simscape.enum.thermaleffects.model
charge_measurement_port=simscape.enum.battery.enable.yes
```

R2022b

Version: 1.0

New Features

Introducing Simscape Battery: Design and simulate battery and energy storage systems

Simscape™ Battery™ provides design tools and parameterized models for designing battery systems. You can create digital twins, run virtual tests of battery pack architectures, design battery management systems, and evaluate battery system behavior across normal and fault conditions. Battery Pack Model Builder is a design tool that lets you interactively evaluate different battery pack architectures. The tool automates the creation of simulation models that match the desired pack topology and includes cooling plate connections so electrical and thermal responses can be evaluated.

Parameterized models of battery packs and battery management systems demonstrate operations, including cell balancing and state of charge estimation. You can use these examples to determine cell requirements, perform trade-off analyses and hardware-in-the-loop (HIL) testing, and generate readable and efficient C/C++ code.

To learn how to build your own battery models with Simscape Battery, see the Get Started with Simscape Battery category.

Battery Pack Model Builder: Create and evaluate different pack architectures

The Battery Pack Model Builder is a design tool that lets you interactively evaluate different battery pack architectures. The tool automates the creation of simulation models that match the desired pack topology and includes cooling plate connections so that you can evaluate electrical and thermal responses. The MATLAB objects of the Battery Pack Model Builder allow you to define your own battery design specifications, visualize your battery in a 3-D space, customize the modeling resolution during simulation, and generate a Simulink® library that contains your custom generated battery blocks. You can use these blocks to assist with virtual battery design and verification, help develop battery control algorithms using Simulink, explore design sensitivities, and design thermal management strategies.

The **Build a Simple Model of a Battery Pack in MATLAB and Simscape** and **Build Detailed Model of Battery Pack From Cylindrical Cells** examples show how to build simple and detailed models of a battery pack by using the objects and methods in Simscape Battery.

- `Cell` — Create a single electrochemical battery cell. The `Cell` object is the foundational element for the creation of a battery pack system model.
- `CylindricalGeometry` — Define a cylindrical geometry for a battery `Cell` object.
- `PouchGeometry` — Define a pouch geometry for a battery `Cell` object.
- `PrismaticGeometry` — Define a prismatic geometry for a battery `Cell` object.
- `CellModelBlock` — Specify set of conditional parameters for a battery `Cell` object.
- `ParallelAssembly` — Create a battery parallel assembly which represents a number of cells connected electrically in parallel under a specific topological configuration or geometrical arrangement.
- `Module` — Create a battery module which represents a number of battery parallel assemblies connected electrically in series.
- `ModuleAssembly` — Create a battery module assembly which represents a number of Module objects connected electrically in series or in parallel.

- `Pack` — Create a battery pack which represents a number of ModuleAssembly objects connected electrically in series or in parallel.
- `BatteryChart` — Construct a battery chart to visualize a Simscape Battery object.
- `buildBattery` — Generate custom library blocks from battery objects

Battery Management System: Implement control strategies using controllers, estimators, monitors, and balancers

Simscape Battery includes Simulink blocks that perform typical battery management system (BMS) functions, such as state estimation, battery protection, cell balancing, thermal management, and current management.

Use these blocks to implement estimation algorithms for battery cell state-of-charge and battery cell state-of-health, simulate battery cell balancing algorithms, size the balancing resistors, get and apply battery charging and discharging limits based on the internal states, simulate battery protection algorithms, and much more. These libraries are fully compatible with the custom battery models that you can generate by using the `buildBattery` function.

These examples perform typical battery management system (BMS) functions:

- Battery Monitoring
- Battery Charging and Discharging
- Battery State-of-Health Estimation
- Battery State-of-Charge Estimation
- Battery Passive Cell Balancing
- **Size Resistor for Battery Passive Cell Balancing**
- Peak Shaving with Battery Energy Storage System
- Protect Battery During Charge and Discharge for Electric Vehicle

Cell Balancing

- Passive Cell Balancing — Implement a passive battery cell balancing algorithm that keeps a similar state-of-charge value in all cells by dissipating the excess charge in a bleed resistor.

Current Management

- Battery CC-CV — Implement a constant current (CC), constant voltage (CV) charging algorithm.
- Battery Charging Current Limit — Calculate the maximum charging current of a battery.
- Battery Discharging Current Limit — Calculate the maximum discharging current of a battery.

Estimators

- SOC Estimator (Adaptive Kalman Filter) — Implement an estimator which estimates the state-of-charge and the terminal resistance of a battery by using the Kalman filter algorithms.
- SOC Estimator (Coulomb Counting) — Implement an estimator which estimates the state-of-charge of a battery by using the Coulomb counting method.
- SOC Estimator (Kalman Filter) — Implement an estimator which estimates the state-of-charge of a battery by using the Kalman filter algorithms.

- SOH Estimator — Implement an estimator that estimates the state-of-health (SOH) of a battery.

Protection

- Battery Current Monitoring — Implement battery overcurrent monitoring.
- Battery Temperature Monitoring — Implement battery temperature monitoring.
- Battery Voltage Monitoring — Implement battery voltage monitoring.
- Fault Qualification — Implement a fault qualification algorithm.

Thermal Management

- Battery Coolant Control — Implement an algorithm for battery coolant control.
- Battery Heater Control — Implement an algorithm for battery heater control.

Cyclers

- Charger — Implement an ideal battery charger.
- Discharger — Implement an ideal battery discharger.
- Cycler — Implement an ideal galvanostat, potentiostat, or a cycler.

Thermal Management: Simulate battery cooling systems for modules or packs

Simscape Battery includes blocks and models of battery cooling systems for simulations of battery thermal management. Use these blocks to add detailed thermal boundary conditions and thermal interfaces to the battery Module or ParallelAssembly blocks:

- Edge Cooling — Model a battery cooling plate with edge cooling.
- Parallel Channels — Model a battery cooling plate with multiple channels and a pair of distributor channels for inlet and outlet flow.
- U-shaped Channels — Model a battery cooling plate with flat channels.

The cooling system blocks in this library are flat cooling plates. These blocks support three main flow configurations: parallel channels, U-shaped rectangular channels, and edge cooling. In the edge cooling configuration, the coolant flows at one end of the flat plate and all the heat from the battery cells is transferred via conduction within the cooling plate material. You can discretize these cooling plates into elements to closely capture temperature spreads resulting from the dynamic interaction with the battery and the coolant flow.

For more information on how to connect a cooling plate to a battery module or parallel assembly, see Connect a Cooling Plate to a Battery Module and Parallel Assembly.

Deployment and Hardware-in-the-Loop Simulation: Implement interfaces and workflows for real time simulation

Simscape Battery includes Simscape blocks to connect to real-time hardware inputs and outputs:

- Active Interface — Provide an abstracted active interface between a battery and a cell supervisory circuit.

- Passive Interface — Provide an abstracted passive interface between a battery and a cell supervisory circuit.

When you use these blocks with real-time hardware, you can easily connect your battery simulations to a real-world battery balancing hardware. Use this capability to help design and verify your battery balancing algorithms, to ensure that they work correctly during both regular operation and adequately detect, degraded operation, or ride through faults.

You can develop and test battery balancing strategies by simulating your custom battery blocks with the blocks in the **Battery Management System** library of Simscape Battery. Options on the physical blocks enable you to assume ideal thermal measurements, thermal estimation or state-of-charge estimation algorithms until you have designed and implemented actual thermal measurements or state-of-charge algorithms.

Featured Examples

New examples in this release include:

- **Build a Simple Model of a Battery Module in MATLAB and Simscape** — This example shows how to create and build a Simscape system model of a battery module.
- **Build a Simple Model of a Battery Pack in MATLAB and Simscape** — This example shows how to create and build a Simscape system model of a battery pack.
- **Build Model of Battery Module with Thermal Effects** — This example shows how to create and build a Simscape system model of a battery module with thermal effects.
- **Build Model of Battery Pack with Cell Aging** — This example shows how to create and build a Simscape system model of a battery pack that includes cell aging.
- **Build Model of Battery Pack with Cell Balancing Circuit** — This example shows how to create and build a Simscape system model of a battery pack with cell balancing circuits.
- **Build Model of Battery Pack for Grid Application** — This example shows how to use Simscape Battery to create and build a Simscape system model of a battery pack from prismatic cells for grid applications.
- **Build Model of Hybrid-Cell Battery Pack** — This example shows how to build a Simscape system model of a hybrid-cell battery pack with two sets of cell run-time parameters. The generated battery pack model contains two types of battery modules, each with different battery cell components inside.
- **Build Detailed Model of Battery Pack From Cylindrical Cells** — This example shows how to create and build detailed Simscape system models for various battery designs and configurations based on cylindrical battery cells.
- **Build Detailed Model of Battery Pack From Pouch Cells** — This example shows how to create and build detailed Simscape system models for various battery designs and configurations based on pouch battery cells.
- Battery Monitoring — This example shows how to use battery management system blocks to monitor the current and temperature of a battery.
- Battery Charging and Discharging — This example shows how to use a constant current and constant voltage algorithm to charge and discharge a battery.
- Battery State-of-Health Estimation — This example shows how to estimate the battery internal resistance and state-of-health (SOH) by using an adaptive Kalman filter.
- Battery State-of-Charge Estimation — This example shows how to estimate the battery state-of-charge (SOC) by using a Kalman filter.

- Battery Passive Cell Balancing — This example shows how to balance a battery with two cells connected in series by using a passive cell balancing algorithm.
- **Thermal Analysis for New and Aged Battery Packs** — This example shows how to evaluate a new and end-of-life (EOL) lithium-ion battery pack.
- **Size Resistor for Battery Passive Cell Balancing** — This example shows how to implement a passive cell balancing for a lithium-ion battery pack.
- Peak Shaving with Battery Energy Storage System — This example shows how to model a battery energy storage system (BESS) controller and a battery management system (BMS) with all the necessary functions for the peak shaving.
- Protect Battery During Charge and Discharge for Electric Vehicle — This example shows how to efficiently charge and discharge a battery for an electric vehicle (EV) and handle battery faults.