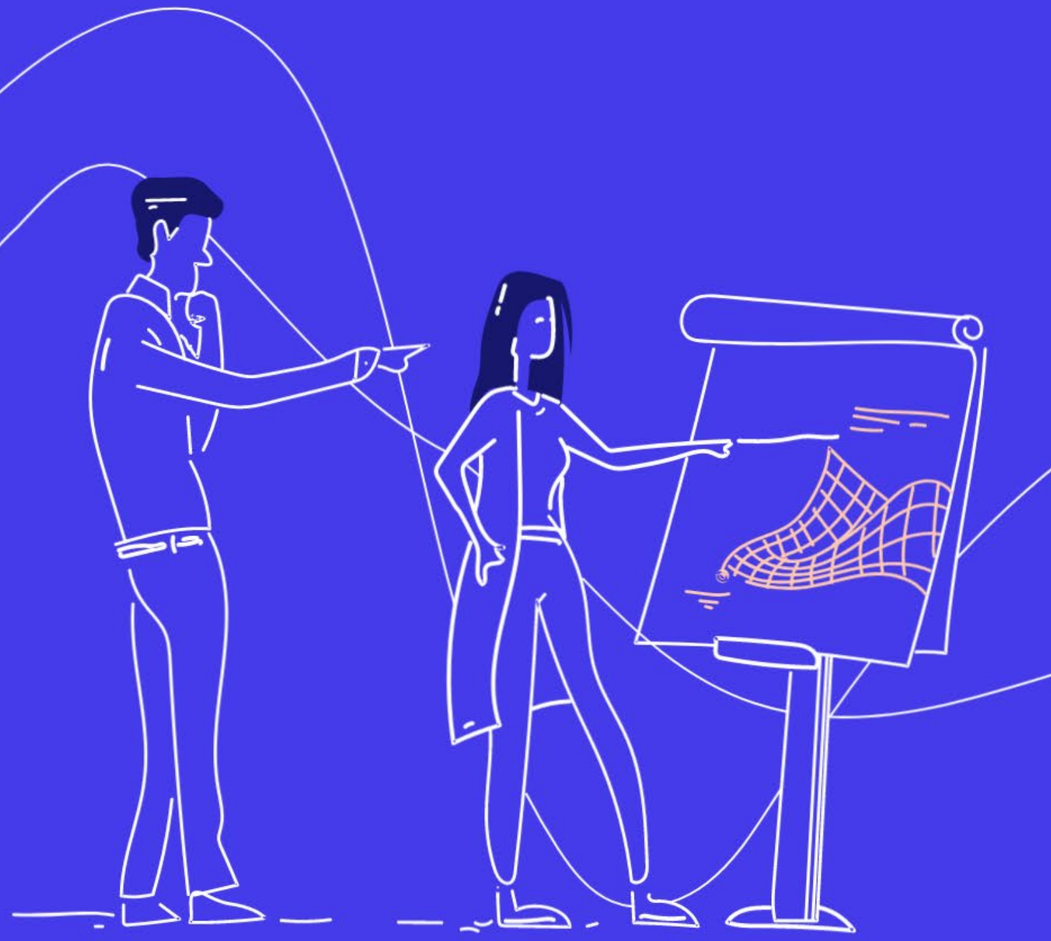
sciengineer

Simulink Fundamentals for Automotive Applications



SciEngineer's training courses are designed to help organizations and individuals close skills gaps, keep up-to-date with the industry-accepted best practices and achieve the greatest value from MathWorks® and COMSOL® Products.

Simulink Fundamentals for Automotive Applications

This two-day course provides a comprehensive introduction to the Simulink environment for automotive engineers. It demonstrates how to create, modify Simulink models and improve simulation accuracy and speed and create reusable model components using subsystems, model references and libraries.

Prerequisites

MATLAB Fundamentals for Automotive Applications

TOPICS

Day 1

- Creating and Simulating a Model
- Modeling Programming Constructs
- Modeling Discrete Systems
- Modeling Continuous Systems

Day 2

- Solver Selection
- Developing Model Hierarchy
- Modeling Conditionally Executed Algorithms
- Combining Models into Diagrams
- Creating Libraries

TRAINING CONTENT - DAY 1

Creating and Simulating a Model

OBJECTIVE: Create a simple Simulink model, simulate it, and analyze the results.

- Introduction to the Simulink interface
- Potentiometer system
- System inputs and outputs
- Simulation and analysis

Modeling Programming Constructs

OBJECTIVE: Model and simulate basic programming constructs in Simulink.

- Comparisons and decision statements
- Vector signals
- PWM conversion system
- Zero crossings
- MATLAB Function block

Modeling Discrete Systems

OBJECTIVE: Model and simulate discrete systems in Simulink.

- Discrete signals and states
- PI controller system
- Discrete transfer functions and state-space systems
- Multirate discrete systems

Modeling Continuous Systems

OBJECTIVE: Model and simulate continuous systems in Simulink.

- Continuous states
- Throttle system
- Continuous transfer functions and state-space systems
- Physical boundaries

Solver Selection

OBJECTIVE: Select a solver that is appropriate for a given Simulink model.

- Solver behavior
- System dynamics
- Discontinuities
- Algebraic loops

Developing Model Hierarchy

OBJECTIVE: Use subsystems to combine smaller systems into larger systems.

- Subsystems
- Bus signals
- Masks

Modeling Conditionally Executed Algorithms

OBJECTIVE: Create subsystems that are executed based on a control signal input.

- Conditionally executed subsystems
- Enabled subsystems
- Triggered subsystems
- Input validation model

Combining Models into Diagrams

OBJECTIVE: Use model referencing to combine models.

- Subsystems and model referencing
- Model referencing workflow
- Model reference simulation modes
- Model workspaces
- Model dependencies

Creating Libraries

OBJECTIVE: Use libraries to create and distribute custom blocks.

- Creating and populating libraries
- Managing library links
- Adding a library to the Simulink Library Browser

sciengineer

Expand your knowledge