sciengineer

Design Verification with Simulink

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.

Design Verification with Simulink

This one-day course focuses on using Simulink Design Verifier to ensure that a design is devoid of possible design errors, is fully tested, and satisfies necessary requirements. Themes of detecting design errors, automatically generating tests, property proving and managing model complexity are explored throughout the course.

Prerequisites

MATLAB Fundamentals and Simulink Fundamentals

DURATION	LEVEL
1 Day	Advanced
●●●	▬▬▬

TOPICS

Day 1

- Understanding the Verification Workflow
- Detecting Design Errors
- Automatically Generating Tests
- Property Proving
- Managing Model Complexity

TRAINING CONTENT - DAY 1

Understanding the Verification Workflow

OBJECTIVE: Become familiar with the verification workflow and learn to prepare a model and perform a simple analysis with Simulink Design Verifier.

- Introducing formal verification methods
- Exploring the Simulink Design Verifier workflow
- Preparing models and ensuring compatibility
- Configuring analysis options
- Generating results

Detecting Design Errors

OBJECTIVE: Learn how to use design error detection to verify a model.

- Introducing types of design errors
- Detecting design errors
- Interpreting results and reporting
- Fixing design errors
- Using parameter tables

Automatically Generating Tests

OBJECTIVE: Review coverage types and learn how to use automatic test generation to complete missing model coverage.

- Understanding types of coverage
- Measuring model coverage
- Increasing coverage with automatic test generation
- Interpreting results and reporting
- Applying custom conditions and objectives

TRAINING CONTENT - DAY 1

Property Proving

OBJECTIVE: Formalize requirements into properties and learn to use property proving to verify that a design meets its requirements.

- Formalizing requirements
- Proving properties
- Debugging falsified properties
- Using temporal operators

Managing Model Complexity

OBJECTIVE: Identify sources of model complexity and learn techniques to improve analysis performance.

- Identifying sources of model complexity
- Implementing block replacements
- Handling nonlinearities
- Managing lookup table complexity
- Working with counters and timers

sciengineer

Expand your knowledge