

Simulink for Analog and Mixed-Signal Verification

OEM: MathWorks • Code: MW-SIMULINK-FOR-ANALOG-AND-MIXED-SIGNAL-

COURSE DESCRIPTION

This one-day course, targeted toward knowledgeable users of Simulink, uses modeling techniques and tools to demonstrate how to do Mixed-Signal Verification using Simulink. Using the Simulink interface Linking MATLAB and Cadence Virtuoso ADE Generating SystemVerilog DPI for Mixed-Signal Using C-Code Generation and the DPI-C Interface Converting models to Fixed-Point Solver Generating RTL Code for digital verification Using control design techniques for bode plot analysis and linearization Automating Simulink tasks

AUDIENCE PROFILE

- IT professionals and technical staff seeking hands-on training in this course's subject area.

PREREQUISITES

- MATLAB Fundamentals, Simulink for Analog and Mixed-Signal Design and basic knowledge of digital signal processing and mixed signal design.

COURSE OBJECTIVES

By the end of this course, participants will be able to:

- Convert Simulink models to fixed-step solvers for generating SV DPI-C or RTL code for verification.
- : Use Simulink and Cadence tools to create and verify analog components
- Use Simulink and Cadence tools to generate and verify components using DPI-C and RTL

COURSE OUTLINE

Module 1: Day 1 of 1

Preparing Simulink Models for Mixed-Signal Verification

- Understand Simulink solvers for Analog and Mixed-Signal design
- Compare variable-step and fixed-step solver simulation results
- Determine an appropriate fixed-step sample rate for your model

Analog Cosimulation

- Cosimulation of Simulink and AMS Designer with variable step solvers
- Synchronization and solver configuration
- Case study: cosimulation of a PLL
- Use Cadence Virtuoso and Simulink to co-simulate the components of a PLL and compare the results

Mixed-Signal Verification

- Generate SystemVerilog DPI-C for Analog Mixed-Signal Verification of a Delta-Sigma ADC
- Use C-Code Generation and DPI-C modules for integration with Cadence Xcelium
- Use model references for simulation and RTL code generation
- Generate RTL code using HDL coder for digital verification
- Simulate RTL code and SystemVerilog DPI-C components and compare the result with Simulink