Signal Processing with Simulink

Prerequisites

MATLAB Fundamentals and basic knowledge of digital signal processing.

Day 1 of 3	
What is Simulink?	**Objective:** Get an introduction to Simulink. What is Simulink? Benefits of using Simulink Simulink add-ons A look at a Simulink model
Creating and Simulating a Model	**Objective:** Explore the Simulink interface and block libraries. Build a simple model and analyze the simulation results. Creating and editing a Simulink model Defining system inputs and outputs Simulating the model and analyzing results
Modeling Discrete Dynamic Systems	**Objective:** Model discrete dynamic systems, and visualize frame-based signals and multichannel signals using a scope. Modeling a discrete system with basic blocks Finding sample times of block outputs Using frames in your model Using buffers Frames vs. multichannel signals Viewing frame-based signals Behavior of delay blocks with frame-based signals Multichannel frame-based signals
Modeling Logical Constructs	**Objective:** Model logical expressions. See how zero-crossing detection is used in Simulink and model simple logic in Simulink using MATLAB code. Modeling logical expressions Modeling conditional signal routing Understanding zero-crossing detection Modeling with the MATLAB Function block
From Algorithm to Model	**Objective:** Create a model from an algorithm specification. Modeling from algorithmic specifications Iterative algorithm development through modeling and simulation Verifying models against specified algorithms

Day 2 of 3

Mixed-Signal Models	**Objective:** Model mixed-signal systems. What is a mixed-signal model? Modeling an ADC with aperture jitter and nonlinearity Case study: Modeling TI's ADS62P29 ADC
Simulink Solvers	**Objective:** Choose the right solver for a Simulink model. Understanding the Simulink solver Solving simple models Solving models with discrete and continuous states Solving models with multiple rates Fixed-step and variable-step solvers Choosing a continuous-state system solver Handling zero crossings Handling algebraic loops
Subsystems and Libraries	**Objective:** Create custom blocks in Simulink, apply masks, and develop custom libraries. Creating subsystems Understanding virtual and atomic subsystems Using a subsystem as a model component Masking subsystems Creating custom block libraries Working with and modifying library blocks Adding custom libraries to the Simulink Library Browser Creating configurable subsystems
Conditional Subsystems	**Objective:** Model systems with parts that are executed conditionally. Conditionally executed subsystems Modeling condition-driven systems with enabled subsystems Modeling condition-driven systems with triggered subsystems Working with an example using the AGC model
Spectral Analysis	**Objective:** Perform spectral analysis in the Simulink environment, and use spectrum computation in an algorithm. Performing spectral analysis with the Spectrum Scope block Choosing spectral analysis parameters Analyzing power spectrum of a motor noise Building a spectral classifier of speech Determining the frequency response of a discrete system

Day 3 of 3	
Designing and Applying Filters	**Objective:** Incorporate filters in a model, and explore different ways filters can be designed and implemented in a Simulink model. Designing filters in Simulink Converting filters to fixed point

Multirate Systems	**Objective:** Model multirate systems. Resample data and explore multirate filter blocks. Modeling multirate systems Exploring blocks for multirate signal processing Resampling oversampled data Designing and implementing anti-imaging and anti-aliasing filters Using multirate filter blocks Case study: Converting professional audio to CD format Converting the design to fixed point
Incorporating External Code	**Objective:** Import or incorporate custom or external MATLAB and C code into a Simulink model. Working with custom and external code considerations Incorporating MATLAB code and C code with the MATLAB Function block
Combining Models into Diagrams	**Objective:** Explore model integration, an important topic for large-scale projects in which several developers are developing different portions of a large system. Exploring model referencing and subsystems Setting up a model reference Setting up model reference arguments Exploring model reference simulation modes Viewing signals in referenced models Browsing the model reference dependency graph
Automating Modeling Tasks	**Objective:** Control and run Simulink models from the MATLAB command line. Automating test runs Checking and modifying parameter settings Finding blocks with specific parameter values Constructing and modifying block diagrams