Simulink Real-Time HDL Workflow with Speedgoat Hardware

OEM: MathWorks • Code: MW-SIMULINK-REAL-TIME-HDL-WORKFLOW-WITH-

COURSE DESCRIPTION

This two-day course focuses on partitioning Simulink models intended for real-time execution on Speedgoat target machines to execute on the CPU and FPGA. A distinction is made here between the algorithm itself and any I/O functionality which may be necessary to communicate with the outside world. Both rapid Control prototyping (RCP) and hardware-in-the-loop Testing (HIL) are discussed. The course consists of various modules which can be combined according to customer needs.

AUDIENCE PROFILE

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

PREREQUISITES

- Simulink Fundamentals (or Simulink Fundamentals for Aerospace Applications or Simulink Fundamentals for Automotive Applications). Knowledge of Simscape™ preferred.

COURSE OBJECTIVES

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

- Understand the concepts of RCP versus HIL. Know the deployment options: CPU versus FPGA.
- Be able to set up the communication between target PC and development computer. Be able to run ready-made applications on real-time target machine.
- Become familiar with the example used during the course. Understand different levels of modeling accuracy. Be able to transform a desktop simulation model to a deployable real-time model.
- Become familiar with the basics of HDL Workflow Advisor for programming an FPGA within the Speedgoat target machine. Be able to deploy a very simple model which uses just digital I/O and does not need any special optimization for deployment.
- Be able to convert a Simulink model using floating point data types to a model using fixed-point data types.
- Be able to configure a Simulink model to make use of functionality that is already available as HDL code.
- Be able to configure a Simulink model to make use of I/O functionality provided by Speedgoat HDL I/O blocksets.
- Be able to combine application algorithm and I/O functionality on a FPGA. Be able to understand and fix timing issues which may occur when generating HDL code from Simulink models.
- Be able to convert Simscape-based models into models only using Simulink blocks which may be deployed to an FPGA.

COURSE OUTLINE

Module 1: Day 1 of 2

Overview on Workflows

- Real-time testing workflows
- Levels of model accuracy
- Deployment options on CPUs and FPGAs

Setting Up Development and Target Computers

- Setting up development computer and target PC
- Starting and stopping the application
- Viewing signals
- Changing parameters at run time

From Desktop to Real-Time Simulation

- Course example: servo motor control
- Different levels of model accuracy
- Simulation with average values
- Simulation with PWM
- From desktop to real-time simulation

Basic HDL Workflow

- HDL workflow overview
- Preparing models for HDL code generation
- HDL Workflow Advisor
- Oversampling

Fixed-Point Conversion

- Converting from floating to fixed-point
- Using internal rules
- Fixed-point scaling and inheritance
- Using the Fixed-Point Tool

Module 2: Day 2 of 2

Integrating External Code – Black Boxing

- Existing external HDL code
- Configuration of model for code generation
- Subsystem for including the external code
- Subsystem for analog input
- Generation of interface model
- Deployment and running of the application

Speedgoat HDL Coder™ I/O Blocksets

- FPGA library blocks - PWM
- CPU library blocks - PWM
- Including the library blocks into course example model
- Finalizing the model

Implementing Algorithms Together with External HDL Code

- Combining I/O functionality and controller algorithm for FPGA deployment
- Understanding timing on a FPGA
- Using the generic ASIC/FPGA workflow within the HDL Workflow Advisor (HDLWA)
- HDLWA – Timing optimization using clock-rate pipelining
- HDLWA – Timing optimization using enable-based constraints

Simscape™ Hardware-in-the-Loop Workflow

- Simscape HIL workflow overview
- Using the Simscape HDL Workflow advisor for converting a Simscape model into a Simulink implementation model
- Validating the implementation model
- Preparing the implementation model for HDL code generation
- Generating HDL code
- Running the HIL application