Programming Xilinx Zynq SoCs with MATLAB and Simulink

Prerequisites

Simulink for System and Algorithm Modeling (or *Simulink for Automotive System Design* or *Simulink for Aerospace System Design*). Knowledge of C and HDL programming languages.

Day 1 of 2	
Zynq Platform Overview and Environment Setup	**Objective:** Configure Zynq-7000 platform and MATLAB environment. Zynq-7000 overview Setting up Zynq platform and software Configuring MATLAB environment Testing connectivity to Zynq hardware
Introduction to Embedded Coder and HDL Coder	**Objective:** Configure Simulink models for embedded code generation and effectively interpret the generated code. Architecture of an embedded application Generating ERT code Code modules Data structures in generated code Configuring a Simulink model for HDL code generation Using HDL Workflow Advisor
IP Core Generation and Deployment	**Objective:** Use HDL Workflow Advisor to configure a Simulink model, generate and build both HDL and C code, and deploy to Zynq platform. Configuring a subsystem for programmable logic Configuring the target interface and peripherals Generating the IP core and integrating with SDK Building and deploying the FPGA bitstream Generating and deploying a software interface model Tuning parameters with External Mode
Using AXI4 Interface	**Objective:** Use various AXI interfaces for data communication between processing system and programmable logic. AXI interface overview AXI4-Lite applications Using AXI4-Stream AXI4 performance considerations
Processor-in-the-Loop Verification	**Objective:** Use processor-in-the-loop to verify the algorithm running on Zynq platform and profile the execution times in your production algorithm. Processor-in-the-loop (PIL) workflow on Zynq PIL verification with model reference Code execution profiling with PIL PIL considerations