MATLAB EXPO 2016 KOREA

4월 28일 (목)

등록 하기 matlabexpo.co.kr
Product Code Generation and Real-Time Testing

김종헌 차장
Senior Application Engineer
MathWorks Korea
Agenda

- Production Code Generation
  - MathWorks’ Code Generation Products
    - Embedded Coder
    - Equivalence Test with SIL and PIL

- Integration Test
  - What’s Simulink Real-Time
  - Automation of Real-Time Testing
Code Generation Products

**MATLAB Coder**
Generate C and C++ code from MATLAB code

**Simulink Coder**
Generate C and C++ code from Simulink and Stateflow models

**Embedded Coder**
Generate C and C++ code optimized for embedded systems
Code Generation Products: Simulink Coder and Embedded Coder

**Simulink Coder**
- Generates code for use in simulation and prototyping applications
- Comes with Generic Real-Time (GRT) based targets

**Embedded Coder**
- Generates efficient code that can be customized to look like hand code for production
- Comes with Embedded Real-Time (ERT) based targets
Rapid Prototyping
Simulink Coder with Simulink Real-Time

Generate, deploy, and tune code for a component (algorithm or controller) on a real-time simulator connected to system hardware.
Run the generated code in real time, tune parameters, and monitor real-time data on the same processor you plan to use in mass production, or a close equivalent to it.
Production Code Generation
Embedded Coder

Select Target

Choose Optimizations and File Packaging

Select Target File Browser: pl.cdr

- System Target File: Description
  - asap2.tlc: ASAM-AP2 Data Definition Target
  - autosar.tlc: AUTOSAR
  - ertl.tlc: Embedded Coder
  - ertl.<lib>: Embedded Coder (host-based shared lib)
  - gtl.tlc: Generic Real-Time Target
  - gtl.<lib>: Generic Real-Time Target (host-based shared lib)
  - grt.tlc: Create Visual C/C++ Solution File for
  - grt.<lib>: Create Visual C/C++ Solution File for
  - ide:ide.cfg: Create IDE Link ERT
  - ide:ide.cfg: Create IDE Link GRT
  - realtime.tlc: Real-Time Toolbox
  - rtm.m: Rapid Simulation Target
  - reali.<lib>: Real-Time Windows Target
  - RTE: Real-Time Component Generator
  - xpc.<lib>: xPC Target

Full Name: C:\MATLAB\R2012b\rtw\plert/ert.tlc
Template Makefile: ert_default.tmf
Make Command: make

Choose Optimizations and File Packaging

- Software environment
  - Code replacement library: C99/ANSI
  - Shared code placement: C99 (ISO)
  - Support:
    - floating-point: T1 C80x (ISO)
    - absolute time: T1 C67x
    - variable-size s
  - Multitarget type definitions:
    - T1 C67x
    - T1 C65x
    - T1 C62x
    - T1 C55x
  - Code interface:
    - T1 C64x
    - T1 C67x
    - T1 C672x
    - T1 C674x
  - Generate preprocessor com

- Suppress error status in real-time model data structure
- Configure Model Functions
- Data exchange
  - MAT-file logging
- Interface: None
Software-in-the-Loop Testing
Embedded Coder

Test generation production code with your environment or plant model to verify a successful conversion of the model to code.
Processor-in-the-Loop Testing
Embedded Coder

Use processor-in-the-loop PIL to evaluate the behavior of a candidate algorithm on the target processor.
Hardware-in-the-Loop Testing
Embedded Coder and Simulink Coder with Simulink Real-time

Final test before integration using simulated plant executing in real time.

Diagram:
- Host
- Simulink
  - Embedded algorithm
  - Model of physical system
- Code Generation
- Harness
  - Embedded microprocessor
  - Real-time simulator
Usage of Code Generation Products

- **Host**
  - Simulink
  - Model of physical system
  - Code Generation
  - Tuning and logging

- **Rapid prototyping hardware (real-time simulator)**
  - Physical system

- **Embedded microprocessor**
  - Tuning and logging

- **Host**
  - Simulink
  - Model of physical system
  - Code Generation
  - Tuning and logging

- **Physical system**

- **Host**
  - Simulink
  - Model of physical system
  - Code Generation
  - PIL Testing on Embedded Hardware

- **Host**
  - Simulink
  - Model of physical system
  - Code Generation

- **Host**
  - Simulink
  - Model of physical system
  - Code Generation

- **Real-time simulator**
Agenda

- Production Code Generation
  - MathWorks’ Code Generation Products
  - Embedded Coder
    - Equivalence Test with SIL and PIL

- Integration Test
  - What’s Simulink Real-Time
  - Automation of Real-Time Testing
Example) Controller of Landing Gear
Simple Software Architecture

Most Development is on Core Software Algorithms

Communication Interfaces

Input Drivers

Core Software Algorithms and Logic

Output Drivers

Special Device Drivers

Scheduler/Operating System And Support Utilities

Actuators

Sensors

Special Interfaces

CCP

ASAP2
Basic Code Generation Workflow for Embedded Target

- Select solver to be fixed step with discrete
- Select ert.tlc target
- Select generate code only
- Select HTML Report for easy review
- Generate code for model
- Review code in browser
Demo: Code Generation Workflow
Code Generation Reports

- Subsystem Report
- Code Interface Report
- Traceability Report
- Static Code Metrics Report
- Code Replacements Report
Demo: Code Generation Reports

Code Generation Report for 'pi_ctrl'

Summary

Code generation for model 'pi_ctrl'

Model version: 1.10
Simulink Coder version: 8.9 (R2015b) 18-Aug-2015
C source code generated on: Mon Feb 01 13:36:49 2016
C source code generated at: C:\Applications\Simulink\PCG_Seminar\work\pi_ctrl_ert_rtw\}

Configuration settings at the time of code generation: click to open
Code generation objectives: Unspecified
Validation result: Not run
Embedded Coder Quick Start

Easily configure Simulink Model to generate production code

- Ask questions about code generation goals
- Auto configure and validate model against the selections
- Show recommended configuration changes
- Apply configuration changes and generate code
Demo: Embedded Coder Quick Start
Optimization Considerations
- Models have a lot of possible settings
- Code Generation Objectives gives a starting point
Demo: Optimizing Generated Code
Legacy Code Tool

- Legacy Code Tool is a utility that generates an S-function automatically from existing C code
- It can also insert an appropriate call to generated code
Demo: Code Generation with Legacy C Code
Agenda

- Production Code Generation
  - MathWorks’ Code Generation Products
  - Embedded Coder
  - Equivalence Test with SIL and PIL

- Integration Test
  - What’s Simulink Real-Time
  - Automation of Real-Time Testing
**Equivalence Test**

SIL (Software-In-the-Loop) and PIL (Processor-In-the-Loop)

- **SW Design**
  - Design Model
    - simulate
    - generate
  - Source Code
    - generate

- **Object Code**
  - Model Test Harness for 100% model coverage
    - Test (with and without instrumentation)
    - verify numerical equivalence with simulation
    - verify 100% code coverage (for traceability)

- **Object Code**
  - Generate Design Model
  - Source Code
  - Model Test Harness for 100% model coverage
  - Test (with and without instrumentation)
  - Verify numerical equivalence with simulation
  - Verify 100% code coverage (for traceability)
Software-in-the-Loop (SIL) Testing:
Verify Production Controller with Software-in-the-loop

Execution
• Host/Host
• Nonreal-time

Compiled C Code
S-Function
(Windows DLL)
Processor-in-the-Loop Testing:
Verify Production Controller with Processor-in-the-loop

Execution
- Host/Target
- Nonreal-time
**Processor-in-the-Loop (PIL) API**

**Problem**
- Embedded IDE Link does not support PIL for an arbitrary combination of
  - Processor
  - Compiler
  - Debugger or download utility
  - Communications channel

**Solution**
- Provide an API that allows integration of third-party or customer tools for
  - Building the PIL application
  - Downloading and running the application
  - Communicating with the application

**Benefit**
- The power of PIL verification is easily adaptable for any target environment
- A fully documented API is stable across MathWorks releases
PIL Testing Example
Infineon Tricore with Trace32 Debugger
Key Benefits of SIL and PIL

- Reuse test vectors for simulation, SIL and PIL
  - Verify correct execution behaviour of compiled code (including on production hardware)
  - Collect metrics for the generated code
    - Code coverage
    - Execution profiling
    - Stack profiling
- Evaluate hardware specific optimizations
- Generate artifacts for IEC-61508, IEC-62304, ISO-26262, EN-50128, and DO-178 certification
- Early verification and fixing of defects reduces cost
Agenda

- Production Code Generation
  - MathWorks’ Code Generation Products
  - Embedded Coder
  - Equivalence Test with SIL and PIL

- Integration Test
  - What’s Simulink Real-Time
  - Automation of Real-Time Testing
Real-Time Simulation and Testing Tasks: Rapid Controls Prototyping
Real-Time Simulation and Testing Tasks:

Hardware-in-the-loop (HIL) Simulation

1. System Model
   - Controller Model
   - Plant Model
   - Verification
   - Code Generation and Download

2. Embedded Controller Hardware
   - Wiring and Signal Conditioning

3. Target Computer Hardware
Today’s Configuration

![Diagram showing the process of code generation and download for Controller Model and Plant Model connected through CAN interface to Target Computer Hardware.]
What is Simulink Real-Time?

*From desktop simulation to real time*

Creation of real-time applications from Simulink models and loading them onto dedicated target computer hardware in 3 automated steps:

1. **Code Generation**
2. **Compile and Link**
3. **Download and Ready to Run**
What is Simulink Real-Time?

Connect to your physical system

- Support for a broad range of I/O types and communication protocols
- Easy drag and drop and configuration within a Simulink model
What is Simulink Real-Time?

Extendable, integrated, and interactive

1. Live parameter tuning, signal monitoring, and execution control
2. Data logging for offline analysis in MATLAB
3. UI/HMI connectivity
4. Extensibility with other software tools (e.g. virtual reality)
Streaming to the Simulation Data Inspector

1. Select the signals to stream
2. Connect to the running target computer
3. Visualize in the Simulation Data Inspector
What Hardware is used with Simulink Real-Time? 

*Development computer + target computer*
Current Performance Level

- 25 microsecond minimum sample time
- < 1 microsecond sample time with FPGA’s
- High performance quad core Intel processors
- Expandable, low latency I/O
Performance Advisor for Real-Time Execution

- Encodes best practices for transitioning to real-time
- Adds testing on the target computer.

Before:
4000 Sec

After:
300 Sec
Configuring Landing Gear Model for HIL Testing

Model:

Problem: Configure solvers to minimize computations so the model can simulate in real time

Solution: Use local solvers on stiff physical networks and explicit solvers elsewhere
Agenda

- Production Code Generation
  - MathWorks’ Code Generation Products
  - Embedded Coder
  - Equivalence Test with SIL and PIL

- Integration Test
  - What’s Simulink Real-Time
  - Automation of Real-Time Testing
Let’s remind Simulink Test….

Tool for authoring, managing, and executing simulation-based tests

Test Harness
Embedded canvas for isolation testing of components

Test Sequence
Easily express logic-based tests

Test Manager
Systematic authoring and management of test cases
Test Automation with Simulink Test

- Available to verify algorithm in real-time
  - Reusing Test Harness and Test Sequence in Simulink Test
  - Avoiding multiple build/download to target
  - Avoiding additional programming to access test results
Key Design Principles

- Use Test Sequence Block for Assessments
  - Evaluated in real-time on target
  - Non-fatal verification language
    - Failure does not stop execution
  - Language constructs for fault recovery
    - Prevent hardware damage

- Runtime variants
  - Avoid multiple build/download to target

- Rapid Iterations
  - Over runtime variants on target hardware
Summary

- Production Code Generation
  - You can get a code for production without human error.
  - Embedded Coder provides “Quick Start” for beginner to try code generation easily.

- Real-Time Testing
  - You can get many benefits with real-time testing
    - Reduce hardware testing
    - Avoid breaking expensive equipment
    - Improve product quality
  - You can do real-time testing in one environment with Simulink.
  - For real-time testing, you can reuse all test cases developed to verify models.