Verification and Validation
Introducing Simulink Design Verifier

Goran Begic, Technical Marketing
Goran.Begic@mathworks.com

June 5, 2007
Agenda

- Verification and Validation in Model-Based Design
  - Overview of verification and validation activities and products that support them
- Introducing Simulink Design Verifier
  - What is Simulink Design Verifier?
  - Why is it important?
  - How does it work?
  - Demonstration
- Summary
- Questions
Verification and Validation in Model-Based Design

- Verification and Validation is one of the inherent benefits of Model-Based Design

- Continuous Test and Verification
  - Important design concepts
    - “It should work”
  - Implementation of requirements
    - “It works“
  - Objective evidence
Verification and Validation

Overview

- **Reporting**
  - Automatic collection and archiving of artifacts

- **Model Checking**
  - Static checking of Simulink and Stateflow models

- **Requirements Traceability**
  - Linking and tracing requirements from and to models

- **Model Coverage**
  - Coverage of model structure and logic

- **Functional Testing**
  - Parameter sweeps
  - Monte Carlo Analysis

- **Formal Methods**
  - Test generation
  - Property Proving

NEW IN R2007a

- **Simulink Report Generator**

- **Simulink Verification and Validation**

- **SystemTest**

- **Simulink Design Verifier**
Verification and Validation

Overview

- **Processor In The Loop**
  - Verification of generated object code on the target processor
  - Co-simulation

- **Hardware In The Loop**
  - Real-Time verification of software inside embedded controller against plant simulation

- **Test And Measurement**
  - Data acquisition
  - Instrument control

**Documentation**

**Test cases, artifacts**

**Co-simulation**

**Test systems (T&M, HILS)**

**Embedded Software**
- Link for Code Composer Studio™
- Link for TASKING®
- Target for Freescale™ MPC5xx
- Target for Infineon C166®
- Digital Hardware
- Link for Cadence® Incisive®
- Link for ModelSim®

**xPC Target**
- xPC Target Embedded Option

**Data Acquisition Toolbox**
- Instrument Control Toolbox
Verification and Validation

Overview

- **Formal Methods**
  - Test generation
  - Property Proving


- **Introducing Simulink Design Verifier**
  - What is Simulink Design Verifier?
  - Why is it important?
  - How does it work?
  - Demonstration
Simulink Design Verifier

What is it?

- New model verification and validation product
  - New verification technology for Simulink and Stateflow

- Based on formal analysis engine from Prover Technology

- Key Features
  - Generates tests for Simulink® and Stateflow® models
  - Detects unreachable design elements in models
  - Proves model properties and generates example of violations
  - Includes blocks for definition of properties
  - Produces detailed test-generation and property-proving analysis reports

Formal analysis
Not simulation
Simulink Design Verifier
Why is it important?

- Building exhaustive tests is hard and time consuming
  - Example: Achieving 100% MC/DC coverage

- Some functional requirements are difficult to prove via simulation
  - Example requirement: Reverse thrust operation shall not engage when aircraft in flight

- Particularly relevant for:
  - Safety critical applications
  - Complex Stateflow models
  - Component based development
Working with Simulink Design Verifier

- Test Generation
  - Simulink and Stateflow models
  - Model Harness with test cases

- Property Proving
  - Simulink and Stateflow models augmented with design properties
  - Analysis Report
  - Detailed report and violations
Simulink Design Verifier
Demonstration

- Example model available in product help
- Test generation for model coverage
- Property definition and proving

Demonstration using debounce model
Summary

- **Design Verifier**
  - Uses static analysis to verify model behavior
  - Complete and exhaustive analysis that uncovers problems that are very difficult to detect using simulation only

- Practical implementation of formal methods in control design applications

- Minimizes the risk of unknown and unexpected execution scenarios
Resources

- Product web page
  - Demo recordings
  - Data sheet
  - User’s Guide

- Webinar on June 12

- Exhibit Hall
  - Verification and Validation station
Questions

- Goran Begic
  - gbegic@mathworks.com
  - 508.647.4313