Model-Based Design: Design with Simulation in Simulink

Ruth-Anne Marchant
Application Engineer
MathWorks
Outline

- Model-Based Design Overview
- Modelling and Design in Simulink
  - Modelling
    - Physical Systems
    - Control logic
  - Simulation
    - System-level optimisation
    - Verification of design changes
- Summary
Traditional Development Workflow

RESEARCH

SPECIFICATIONS

REQUIREMENTS

DESIGN AND IMPLEMENTATION

INTEGRATION AND TEST

C/C++
Embedded Software

Algorithm Design
Embeddable Algorithms

MCAD/MCAE
Mechanical Components

EDA
Electrical Components
Problems in Traditional Development Workflow

- Requirement Documents
- Paper Specifications
- Physical Prototypes
- Manual Coding
- Traditional Testing

- Research
- Requirements
- Specifications
- Design and Implementation
- Integration and Test

- C/C++ Embedded Software
- Algorithm Design
- MCAD/ MCAE Mechanical Components
- EDA Electrical Components
Model-Based Design Workflow

Traditional System Development Workflow
1. Research
2. Requirements and Specifications
3. Design
4. Implementation
5. Test and Verification
Model-Based Design: Specifications

**CAPABILITIES**
- Executable specification
- Executable constraints
- Links to requirements

**BENEFITS**
- Early validation and test development
- Clear specification
- Simulate whole system, including environment
- Tight link to requirements
Model-Based Design: Requirements

Formalize requirements as properties and objectives

Model system response bounds

Trace to requirements in DOORS, Word, Excel, etc.

Environment Models
Physical Components
Algorithms
Model-Based Design: Design

**CAPABILITIES**
- Refine model description
- Add fixed point, timing, component interface details

**BENEFITS**
- Fast design exploration
- Design optimization
- Find flaws before implementation
Model-Based Design: Design

**CAPABILITIES**
- Refine model description
- Add fixed point, timing, component interface details

**BENEFITS**
- Fast design exploration
- Design optimization
- Find flaws before implementation
Model-Based Design: Implementation

**CAPABILITIES**
- Rapid Prototyping
- Automatic Code Generation:
  - C/C++
  - HDL
  - PLC

**BENEFITS**
- Eliminate hand-coding
- Eliminate hand-code errors
- Hardware target portability
- Better testability and reuse
- Bridge between domain, software, and hardware knowledge experts
Model-Based Design: Test and Verification

CAPABILITIES
- Model Verification
- Software Verification
- Hardware-in-Loop
- Test and Measurement

BENEFITS
- Detect errors earlier
- Reduce use of physical prototypes
- Implementations that work first time
- Reuse tests throughout development stages
Building a Wave Farm with Model-Based Design

As engineering tools, MATLAB and Simulink provide significant value. They are just as valuable as innovation tools because they enable us to quickly test ideas that we would otherwise never try.

— Jonathan Flévez, Carnegie Wave Energy
Outline

- Model-Based Design Overview
- Modelling and Design in Simulink
  - Modelling
    - Physical Systems
    - Control logic
  - Simulation
    - System-level optimisation
    - Verification of design changes
- Summary
Outline

- Model-Based Design Overview
- Modelling and Design in Simulink
  - Modelling
    - Physical Systems
    - Control logic
  - Simulation
    - System-level optimisation
    - Verification of design changes
- Summary
Mechanical System Modelling

Simscape
Electrical System Modelling

Simscape

Bridge Rectifier (AC to DC)
Electrical System Modelling
Electrical System Modelling
Multi-Domain Modelling of Physical Systems

Simscape

MATLAB & Simulink
Control System Design
Defining System Mode Logic

- Define mode logic using state machine in Stateflow
- Generate production code directly from model
Outline

- Model-Based Design Overview
- Modelling and Design in Simulink
  - Modelling
    - Physical Systems
    - Control logic
  - Simulation
    - System-level optimisation
    - Verification of design changes
- Summary
Optimise Entire System

- Use optimisation algorithms to automatically tune parameter values
  - Match response
  - Meet requirements

- Optimise system performance (controller and physical system)
Distributing Simulations with Parallel Computing

- Simulating in parallel
  - Distribute simulations to multiple cores/processors
  - Dramatic speedup for sets of simulations (parameter sweeps, flight cycles optimisations, and more)

```
for \( \text{parfor} \)
```
**Problem:** Minimize simulation time to run a parameter sweep on the HEV model

**Solution:** Use Parallel Computing Toolbox to speed up the sweep
Outline

- Model-Based Design Overview
- Modelling and Design in Simulink
  - Modelling
    - Physical Systems
    - Control logic
  - Simulation
    - System-level optimisation
    - Verification of design changes
- Summary
Automatically Run Tests And Document Results

**Situation:**

![Diagram showing the cycle of design change, test, evaluate results, and document]

**Problem:** Evaluate test results quickly to make design changes and document the results.

**Solution:** Use Simulink Report Generator to automatically document tests and results.
Outline

- Model-Based Design Overview
- Modelling and Design in Simulink
  - Modelling
    - Physical Systems
    - Control logic
  - Simulation
    - System-level optimisation
    - Verification of design changes
- Summary
Key Points

- Simulink is a multi-domain modelling and simulation environment facilitating Model-Based Design
- Optimise the system-level performance
- Accelerate your development
  - Speed up simulations using Parallel Computing Toolbox
  - Speed-up processes using Simulink Report Generator
Call to Action

Learn more about Model-Based Design with Simulink

- Explore our [website](au.mathworks.com)

- Contact me:
  - Ruth-Anne Marchant
    - [ruth-anne.marchant@mathworks.com.au](mailto:ruth-anne.marchant@mathworks.com.au)
Q & A