Automotive | Automation
Green Energy
From Requirement to Execution

Implementing a PLCnext-Based Turbine Control System in Simulink

Jakoba Reimann
Key Takeaways

1. Model-Based Design offers you many benefits

2. Matlab Simulink can help us to get one step closer to a world with 100% renewable energies

3. We show you the way from your first idea to the final product on hardware
Sokratel GmbH - Departments

Kiel
Hamburg
Bochum
Augsburg

Information Security: TISAX Level 3 (Based on ISO 27001)
Content of this presentation

What is model-based design (MBD) and why do we use it?

The process of a requirement – from the idea to the prototype
What is model-based design (MBD) and why do we use it?

Continuous Test and Verification

- Requirement specification
- Design Controller & Plant
- Hardware Target

Automatic Code Generation
What is model-based design (MBD) and why do we use it?

Controller and plant design
- Create a model of the plant
- Verify controller directly in closed loop and configure controller model

Automatic code generation
- No reimplementation in C, C++, ST or similar needed
- Hardware independent
- Applicable for different targets

Continuous verification
- Quick fault detection and fixing

Continuous integration
- Automatic test execution
- Static model & code analysis for design verification and Guideline alignment
What is model-based design (MBD) and why do we use it?

Traditional development

- Several tools throughout the development process
- Manual translation into executable code
- Testing after finalization of implementation

MBD

- Reduce inefficiencies
- Reduce defects and time
- Early bug discovery
- One tool environment
- Automatic code generation
- Testing throughout the whole process
Introduction of the process

The process of a requirement – from the idea to the prototype

Example Requirement:

During the night, the turbine shall activate a green light on top of the nacelle.
Introduction of the process

Continuous Test and Verification

Requirement specification

Design Controller & Plant

HW Target

Automatic Code Generation
- PLCNext Target for Simulink

Jira

Simulink Requirements

Simulink Test

Simulink Test

Simulink Coder

www.sokratel.de
Model in the Loop testing (MiL)

Test closed loop behaviour with a representative model of the plant

- Purely simulative
- Testing of scenarios
- Integration of model parts to a whole system
Hardware in the Loop testing with PLCnext (HiL)

- Real-time plant simulation
- Testing of communication and application interaction
- Time and cost efficient
- Model Viewer PLCnext Engineer

Test physical hardware with a representative model of the plant
Hardware in the Loop testing with PLCnext (HiL)
How we automate our daily work

1. Check Out
2. Static Code Analysis
3. Test Execution
4. Code Generation
5. Clean Up

Developer
Commit
Model

Simulink
Model Advisor

www.sokratel.de
From theory to reality
Thank you for your attention!

www.sokratel.de