Designing and Deploying Embedded Algorithms on PLCs and other Industrial Controllers
Why System Control Design more than Ever

- **Reliability:** Many industrial applications require high maintenance costs. Control Design helps to reduce wearing and lower the risk of malfunctions.
Why System Control Design more than Ever

- **Flexibility**: The same system can be used to answer specific market requests or tailored solutions.
Digital Transformation and flexible production lead to increasingly complex equipment that involves **multidisciplinary fields** (software, mechanics, hydraulics, pneumatics, electronics, etc.)

Design and commissioning of industrial equipment require simulation, virtual commissioning and code generation for industrial controllers (e.g. PLCs and industrial PCs)
Growing complexity

Growing complexity of mechatronic systems based on industrial controllers

...requires new design methods.

Model Based Design
Metso Develops Controller for Energy-Saving Digital Hydraulic System for Papermaking Equipment Using Model-Based Design

Challenge
Precisely control the speed, position, and pressure of calendar rolls in paper finishing equipment

Solution
Simulate, prototype, and implement advanced controls for a digital hydraulic system using Model-Based Design

Results
- Months of design time saved
- Weeks of customer startup time eliminated
- System reliability increased

“Using Model-Based Design with MATLAB and Simulink, we achieved multiple goals simultaneously. We developed a sophisticated controller for digital hydraulics that is more reliable, accurate, and efficient than previous systems, and we accelerated development, which gives us a competitive advantage.”

Kari Leminen
Metso
Challenge
Accelerate the development of control systems for injection molding machinery

Solution
Use Model-Based Design with MATLAB and Simulink to model controllers and plants, run closed-loop simulations to minimize hardware testing, and generate PLC Structured Text

Results
• Control algorithms developed and debugged without hardware
• Controller quality improved
• Test data analysis accelerated

Link to user story
“Model-Based Design reduces the time needed to produce quality control algorithms. Simulations help us understand the system, and code generation enables us to maintain a single source for the design. The results are faster development and higher-quality systems.”

Hannes Bernhard
ENGEL
Model-Based Design for Industrial System Control Design

What if you were able to verify your system’s behavior through the entire design process?

What if you could use your models not only for design simulation but also as a Digital Twin during lifetime of your system?

Model Based Design
Model-Based Design for Industrial System Control Design

Step 1: Desktop Simulation

- Prototype new functionality and combine with existing code
Model-Based Design for Industrial System Control Design

Step 1: Desktop Simulation

- Prototype new functionality and combine with existing code
- Perform (automated) system tests that would not be feasible outside of simulation
Step 1: Desktop Simulation

- Prototype new functionality and combine with existing code
- Perform (automated) system tests that would not be feasible outside of simulation
- Optimize parameters (software, mechanics, hydraulics, etc.)
Model-Based Design for Industrial System Control Design

Step 2: Code Generation

- Design and test hardware independent functionality (C/C++, IEC 61131-3, HDL)

Implementation

- Environment Models
- Physical Plant Models
- Control / Supervisory Logic Models

MathWorks Simulation Environment

Hardware Abstraction Layer

IDE (vendor specific)

Automation Hardware (vendor specific)
Model-Based Design for Industrial System Control Design

Step 2: Code Generation

<table>
<thead>
<tr>
<th>Vendor</th>
<th>IDE</th>
<th>IEC 61131-3</th>
<th>C/C++</th>
</tr>
</thead>
<tbody>
<tr>
<td>3S - Smart Software Solutions</td>
<td>CODESYS</td>
<td></td>
<td>✔</td>
</tr>
<tr>
<td>ABB / B&amp;R Industrial Automation</td>
<td>Automation Studio</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Bachmann Electronic</td>
<td>SolutionCenter</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Beckhoff Automation</td>
<td>TwinCAT</td>
<td></td>
<td>✔</td>
</tr>
<tr>
<td>Bosch Rexroth</td>
<td>IndraWorks</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Mitsubishi Electric</td>
<td>CW Workbench</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Ingeteam</td>
<td>Ingesys I3C</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Omron</td>
<td>Sysmac Studio</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Phoenix Contact</td>
<td>PC WORX</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Rockwell Automation</td>
<td>RSLogix / Studio 5000</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Schneider Electric</td>
<td>Control Expert / Unity Pro</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Selectron</td>
<td>CAP1131</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Siemens</td>
<td>TIA Portal / STEP 7</td>
<td>✔</td>
<td>✔</td>
</tr>
</tbody>
</table>
Step 3: Hardware in the Loop

- Emulate the behavior of the physical system (plant model) in real-time
- Connect the virtual plant to your PLC or industrial PC (e.g. over an industrial fieldbus)
Conclusion

- Model-Based Design helps to **embrace the growing complexity** on equipment based on industrial controllers.
Conclusion

- Model-Based Design helps to **embrace the growing complexity** on equipment based on industrial controllers.

- Code generation from MATLAB, Simulink, Stateflow and Simscape is available for **all major industrial controls platforms**.

  Approach us if you don’t find your vendor in this table.

<table>
<thead>
<tr>
<th>Vendor</th>
<th>IDE</th>
<th>IEC 61131-3</th>
<th>C/C++</th>
</tr>
</thead>
<tbody>
<tr>
<td>3S - Smart Software Solutions</td>
<td>CODESYS</td>
<td>✔️</td>
<td></td>
</tr>
<tr>
<td>ABB / B&amp;R Industrial Automation</td>
<td>Automation Studio</td>
<td>✔️ ✔️ ✔️</td>
<td></td>
</tr>
<tr>
<td>Bachmann Electronic</td>
<td>SolutionCenter</td>
<td>✔️</td>
<td></td>
</tr>
<tr>
<td>Beckhoff Automation</td>
<td>TwinCAT</td>
<td>✔️</td>
<td>✔️ ✔️</td>
</tr>
<tr>
<td>Bosch Rexroth</td>
<td>IndraWorks</td>
<td>✔️</td>
<td>✔️ ✔️</td>
</tr>
<tr>
<td>Mitsubishi Electric</td>
<td>CW Workbench</td>
<td>✔️</td>
<td>✔️ ✔️</td>
</tr>
<tr>
<td>Ingeteam</td>
<td>Ingesys IC3</td>
<td>✔️</td>
<td></td>
</tr>
<tr>
<td>Omron</td>
<td>Sysmac Studio</td>
<td>✔️</td>
<td></td>
</tr>
<tr>
<td>Phoenix Contact</td>
<td>PC WORX</td>
<td>✔️</td>
<td>✔️ ✔️</td>
</tr>
<tr>
<td>Rockwell Automation</td>
<td>RSLogix / Studio 5000</td>
<td>✔️ ✔️ ✔️</td>
<td></td>
</tr>
<tr>
<td>Schneider Electric</td>
<td>Control Expert / Unity Pro</td>
<td>✔️ ✔️</td>
<td></td>
</tr>
<tr>
<td>Selectron</td>
<td>CAP1131</td>
<td>✔️</td>
<td></td>
</tr>
<tr>
<td>Siemens</td>
<td>TIA Portal / STEP 7</td>
<td>✔️ ✔️</td>
<td>✔️</td>
</tr>
</tbody>
</table>
Conclusion

- Model-Based Design helps to **embrace the growing complexity** on equipment based on industrial controllers.

- Code generation from MATLAB, Simulink, Stateflow and Simscape is available for **all major industrial controls platforms**.

- Real-time testing using connectivity to all relevant **industrial fieldbus and ethernet protocols**.
Conclusion

- Model-Based Design helps to embrace the growing complexity on equipment based on industrial controllers

- Code generation from MATLAB, Simulink, Stateflow and Simscape is available for all major industrial controls platforms

- Real-Time testing using connectivity to all relevant industrial fieldbus and ethernet protocols
Learn More

- **White Paper**
  - Virtual Commissioning with Model-Based Design

- **Webinars**
  - Virtual Commissioning with Simulink
  - Virtual Commissioning of Production Machines

- **User Stories**
  - Metso Develops Controller for Energy-Saving Digital Hydraulic System for Papermaking Equipment Using Model-Based Design
  - ENGEL Speeds Development of Injection Molding Machine Controllers