MathWorks™ Products
for Control Design

MATLAB® & SIMULINK®

Accelerating the pace of engineering and science

The MathWorks

MATLAB® & SIMULINK®

The MATLAB® and Simulink® product families provide a complete set of tools for applying Model-Based Design to control system development.

Simulink® offers an interactive graphical environment and customizable block libraries for designing, simulating, implementing, and testing dynamic and embedded systems. Add-on products support specific modeling and design tasks, including code generation, algorithm implementation, test, and verification. Simulink is integrated with MATLAB®, giving you access to a rich set of tools for algorithm development and data analysis.

Plant Modeling

Simulink enables you to combine modeling techniques based on mathematics, physical principles, and measured data in a single environment.

Predefined continuous dynamics, algorithmic, and nonlinear function blocks provide a foundation for modeling any dynamic system mathematically.

Physics-based modeling tools let you build plant models using elements that correspond to the physical system. You can model mechanical, electrical, and hydraulic systems, as well as common aerospace and automotive vehicle components, and employ standard reference models and specifications.

You can use test data to refine your Simulink model by reconciling differences between the test data and the model outputs, or to create a stand-alone model for use in Simulink.

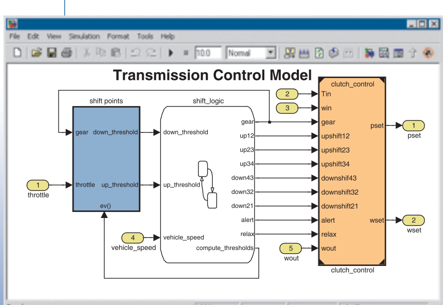
Control Design

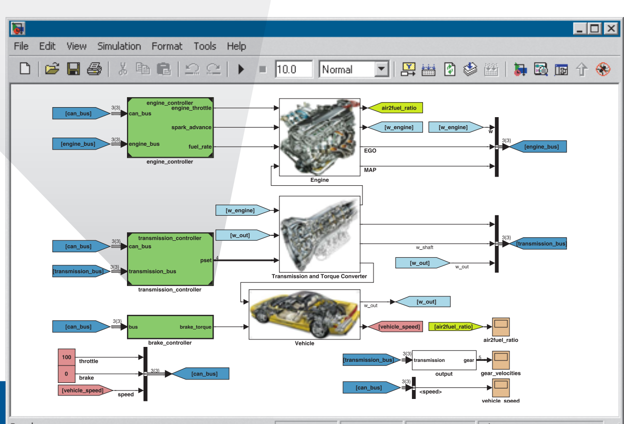
The MATLAB and Simulink product families offer proven control design and optimization algorithms. You can create models of your embedded software that combine supervisory control logic with feedback control algorithms and include floating-point and fixed-point design implementations.

You can apply classical and modern control design methods to design and analyze single and multivariable controllers. Robust control design techniques let you factor model uncertainty into your design to reduce controller sensitivity to parameter variations and modeling errors.

Specialized control tools, such as fuzzy logic, model predictive control, and constraint-based optimization, are also available.

You can design complex control, supervisory, or mode logic behavior using graphical language elements. Based on finite state machine theory, these elements provide control flow, truth tables, temporal operators, and directed-event broadcasting.





Simulink® model of a powertrain control system incorporating a transmission control component model.

"Model-Based Design with MATLAB and Simulink is fully proven and indispensable to our engineering process, and gives us an advantage over our competition."

– Shigeaki Kakizaki, Nissan Motor Co., Ltd.

Simulation

With Simulink you can simulate the dynamic behavior of your plant, controller, and environment together. Fixed-step and variable-step solvers, a graphical debugger, and other tools ensure simulation efficiency and accuracy.

You can accelerate model execution and use model profiling to identify performance bottlenecks. Scopes, display blocks, and other tools let you visualize and analyze simulation results. You can interact with your simulations in a 3-D virtual reality world, represent your system with customized displays, and interface to other simulation environments through an open application program interface.

Rapid Control Prototyping and Hardware-in-the-Loop Testing

The Simulink product family provides a host-target prototyping environment that includes a real-time kernel, I/O device drivers, and software for creating a rapid control prototyping or hardware-in-the-loop (HIL) system.

When you create a real-time application from the model, the control algorithm, device drivers, and real-time kernel are automatically generated, linked, and downloaded to the real-time target system. Using commercial I/O hardware you can connect the target system to the sensors and actuators of a physical plant for real-time software testing and validation. For your target system you can use any form factor

continued on next page

About Model-Based Design

In Model-Based Design, a system model is at the center of the development process—from requirements capture and design to implementation and test.

The model is an executable specification that is continually elaborated throughout the development process. Simulation shows whether the model works correctly. When software and hardware implementation requirements are included, such as fixed-point and timing behavior, you can automatically generate code for embedded deployment and create test benches for system verification, saving time and avoiding the introduction of hand-coding errors.

The MATLAB and Simulink product families support every facet of Model-Based Design.

Model-Based Design improves development efficiency by enabling engineers to:

- Use a common design environment across project teams
- Integrate testing with design to identify and correct errors
- Refine algorithms through multidomain simulation
- Automatically generate embedded software
- Develop and reuse test suites
- Automatically generate documentation
- Reuse designs across multiple hardware targets

CUSTOMER SUCCESSES

Streamlining development with MathWorks products and Model-Based Design

ABB accelerated development of a controller for power electronic applications by using automatic code generation.

BAE SYSTEMS Controls reduced design and rework costs by reusing models from other projects on the baseline design of a UAV autopilot.

Chess Embedded Technology cut development time by as much as 70% for a hydrogen generator controller.

DaimlerChrysler automatically generated cruise controller code that required about 16% less RAM than handwritten code and met all efficiency and structure requirements.

Honeywell Commercial Aviation Systems updated and certified flight-control system software while reducing development time by 60%.

Nissan cut development time by 50% on an emission reduction system that improves fuel economy and reduces ozone-depleting hydrofluorocarbons.

Xerox reduced paper-path controls development time by up to 80% with a streamlined workflow based on integrated tools and rapid prototyping.

continued from previous page

PC system. Also available is a custom-configured industrial PC system for rapid control prototyping.

You can automatically generate code from the plant model for HIL testing. The HIL system simulates the dynamics and behavior of the physical system to enable real-time testing of embedded control software on the designated target processor.

Embedded Code Generation

You can automatically generate production-quality ANSI/ISO C code (production code) from your algorithm model. Customizable and ready for deployment on virtually any embedded microprocessor, the code can be traced back to the model and design requirements and executed with or without a real-time operating system.

You have full access to the generated code, which can be incorporated into your real-time execution environment. You can include legacy code and tailor the embedded software for specific hardware or software target platforms.

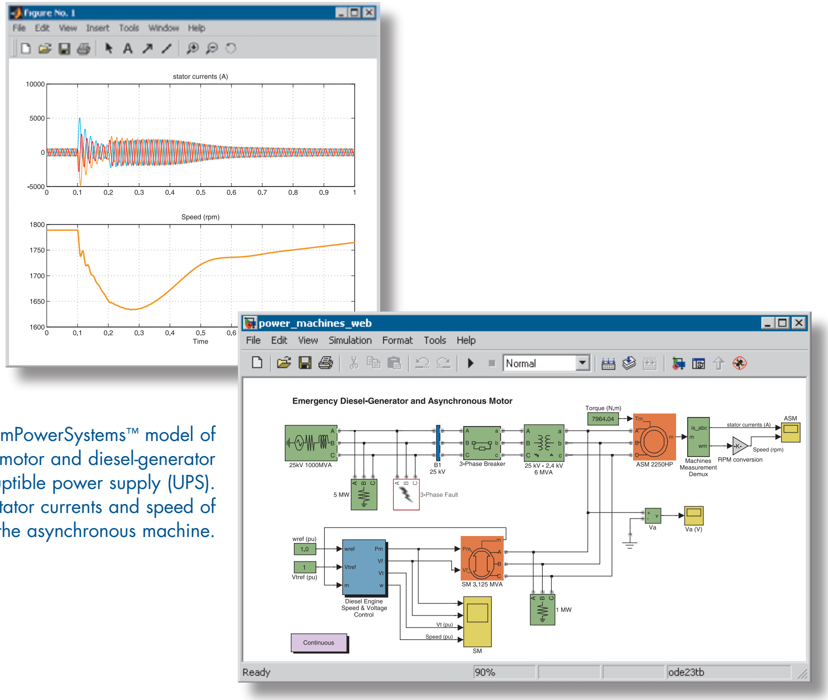
Verification and Validation

With the Simulink product family you can test and verify your design throughout the development workflow.

System requirements can be associated with design models, letting you trace requirements through to implementation, and with specific test cases that are then applied to the design model.

You can measure the level of testing by collecting structural coverage metrics at the model level, including condition, decision, and modified decision/condition coverage (used in safety-critical systems developed under RTCA/DO-178B).

As you implement the design, you can reuse your test cases to perform software-in-the-loop, processor-in-the-loop, or HIL testing to validate the automatically generated code running as software, in the target processor, or in the production embedded system, respectively.



SimPowerSystems™ model of an asynchronous motor and diesel-generator uninterruptible power supply (UPS). The figure shows stator currents and speed of the asynchronous machine.

Resources and Support

ONLINE USER COMMUNITY
www.mathworks.com/matlabcentral

DEMOS
www.mathworks.com/cdademos

PARTNER PRODUCTS
www.mathworks.com/connections

TECHNICAL SUPPORT
www.mathworks.com/support

TRAINING SERVICES
www.mathworks.com/training

CONSULTING SERVICES
www.mathworks.com/consulting

BOOKS BASED ON MATLAB® AND SIMULINK®
www.mathworks.com/books

WORLDWIDE CONTACTS
www.mathworks.com/contact

For more information on MathWorks products and services: **www.mathworks.com**

91258v01 03/08

The MathWorks