MATLAB and Simulink in the World: Connecting MATLAB and Simulink to Hardware

Engineers and scientists connect MATLAB® and Simulink® to FPGAs, microprocessors, cameras, instruments, and other hardware to design, test, and verify systems that combine hardware components and software algorithms. This approach supports project-based learning with low-cost hardware at schools and universities, as well as high-performance production system development at automotive, aerospace, communications, and other commercial organizations.

Project-Based Learning: Ohio State University
Teaching programming with MATLAB and Arduino hardware

For their final project in ENGR 1181: Fundamentals of Engineering I, first-year OSU students develop a MATLAB program to control a model train as it travels around a circular track. They implement their programs on an Arduino Uno using the MATLAB Support Package for Arduino Hardware. To optimize limited lab time and hardware, instructors developed a virtual train simulator in MATLAB. All 1700 students can use the simulator to design, test, and debug their algorithm code. They can then use the same code on the actual train set in the lab.

» Read the article

FPGA Design: Siglead
Accelerating development of signal processing components for mass storage devices

Storage devices, including solid state drives (SSDs) and hard disk drives (HDDs), require advanced signal processing subsystems for high-speed data encryption and error correction. Algorithms for these subsystems are often developed in C or C++. Translating the C algorithms into HDL is time-consuming and error-prone, as the sequential behavior of C must be mapped to the parallel behavior of hardware. Siglead engineers use MATLAB, Simulink, and HDL Coder™ to model, simulate, verify, and generate HDL code for the signal processing components in their SSD and HDD systems. With HDL code automatically generated from a verified, cycle-accurate Simulink model, modifications that used to take days are now completed in hours.

» Read the story
Implementing embedded software for certification of an emergency transport ventilator

The MEDUMAT Transport ventilator is the most advanced—and the most complex—ventilator that Weinmann has ever developed. Model-Based Design with MATLAB and Simulink enabled Weinmann to handle the increased complexity and achieve compliance certification. Engineers developed a Simulink plant model, which included hardware components and a mechanical model of human lungs, and a Simulink and Stateflow model of the controller. After running closed-loop simulations of the controller and plant, they generated production code for deployment to a 16-bit Infineon® microcontroller using Embedded Coder®. This approach helped Weinmann certify MEDUMAT Transport to ISO/IEC 62304, ISO 10651-3, DIN EN ISO 13485, and DIN EN ISO 14971 standards.

» Read the story
Real-Time Testing: Korea Institute of Machinery and Materials

Creating a real-time prototype of an antirolling control system to stabilize mobile harbors

KIMM researchers needed to evaluate an antirolling system for a catamaran-based mobile harbor platform. An aggressive schedule allowed time for only one prototype. The researchers converted a SolidWorks® assembly of the mobile harbor platform into a SimMechanics™ model, and developed Simulink models of the controllers. They performed closed-loop simulations using the catamaran model to verify the control algorithms. C code was automatically generated from the controller models and executed in real time using xPC Target running on a PC/104 computer with an I/O board interface to the catamaran hardware.

» Read the story

Communication Systems: Semtech

Generating production VHDL code for FPGA and ASIC implementation of wireless RF receivers

Semtech engineers needed to accelerate development of optimized digital receiver chains for wireless RF devices. They developed a Simulink model of the complete receiver chain. After converting the model from floating point to fixed point, they conducted bit-true simulations. The team then generated VHDL® from the model, verifying the code by cosimulating their Simulink design with the Mentor Graphics® Questa® simulator. By automatically generating VHDL from the Simulink model, they eliminated the tedious hand-coding of each block, reducing prototype development time from months to weeks.

» Read the story
FPGAs in thermal imaging infrared cameras filter and process signals generated by sensors. Turning a new signal processing concept into an algorithm that runs in real time on a production camera can be time-consuming, because hardware engineers must translate algorithms developed by algorithm engineers into HDL. At FLIR Systems, engineers develop and simulate advanced algorithms in MATLAB. Synthesizable HDL code automatically generated by HDL Coder from these MATLAB algorithms is implemented and tested on the FPGA. As a result, prototyping time is reduced by up to 60%, enabling the FLIR team to explore new designs and enhance existing ones.

» Read the story

Products Used
- MATLAB
- Simulink
- Embedded Coder
- HDL Coder
- SimMechanics
- Simscape
- Stateflow
- xPC Target

Learn More
- Hardware Support
- User Stories

See more articles and subscribe at mathworks.com/newsletters.