Q&A: Why System-Level Design Is the Future of Engineering

National Instruments Cofounder and Business and Technology Fellow Jeff Kodosky provides an overview of system-level design, explains its benefits and discusses how innovations in NI LabVIEW make it ideal for this approach.

Q: What is system-level design?

System-level design is a methodology where the engineer accounts for all the components of a system when designing the solution. For the embedded control system of a jet engine, for example, the system designer considers the dynamics of the engine, the sensors and actuators that measure and interact with the engine, the I/O devices that acquire sensor signals and generate actuator commands, the communication buses that connect the I/O to the processor or processors and, finally, the control logic that executes on the processor or processors.

Q: How is system-level design different from traditional design?

In a traditional design method, you may have a hardware designer working on a device to acquire a voltage from an accelerometer at a frequency of 102.4 kHz, filter the signal using a field-programmable gate array (FPGA) and communicate to a processor through an SPI bus. The software designer is asked to implement a transfer function in embedded software running on a processor that will use input data from the SPI bus. The hardware designer knows little about what happens downstream of the design, while the software designer knows little about the accelerometer. Moreover, neither the hardware or software designer knows much about the jet engine vibrations that will be measured with the accelerometer or the nature of the transfer function. The control engineer, who knows the jet engine dynamics and the transfer function has little insight into the hardware and software design that is critical for making the control system successful. This situation can lead to long and frustrating design iterations to achieve the right trade-offs for a successful design. The more complex systems are, the higher the likelihood that traditional design will cause errors and delays in bringing the design to market.

Q: What benefit do system-level design tools offer?

A good system-level design tool provides multiple levels of abstraction and intuitive integration between hardware and software. This makes it so that a designer can operate at a high level that takes into account the various components of the system and still produce a design that can do something in the real world. The system-level tool must have the ability to express mathematical equations, execute them in a real-time embedded multicore processor or FPGA and interface with a myriad of sensors and actuators through intuitive models. In our fictional case, the control engineer could use such a system-level design tool to create a prototype of a jet engine control system that would run on real embedded hardware with real-time performance.

This sounds too good to be true, and it is – if you only have a stand-alone software tool. The key is having a system-level design tool that is tightly integrated with a flexible hardware architecture to provide the system-aware experience needed. At National Instruments, we have invested heavily to offer this combination with LabVIEW software and NI reconfigurable I/O (RIO) architecture-based hardware.

Engineers have long been using flowcharts and block diagrams to describe systems, so the LabVIEW graphical dataflow model aligns naturally with system descriptions. The graphical nature of LabVIEW is well-suited to represent asynchronous and parallel tasks that are often part of complex systems. LabVIEW has always sported rich interfaces to hardware devices, sensors and actuators. LabVIEW also includes hundreds of math, analysis and signal processing functions as well as the ability to import code and IP written in standard software and hardware formats.

LabVIEW has the ability to target real-time multicore processors and high-performance FPGAs, making the RIO architecture, which consists of a real-time processor and FPGA connected to I/O, a perfect hardware platform for system-level design. The designer using LabVIEW doesn't have to be aware of the various cores on a multicore processor, the intricacies of a real-time scheduler or the details of the parallel paths on FPGAs to take advantage of them. Additionally, LabVIEW designs are easily portable between form factors that share the same architecture such as NI CompactRIO and NI FlexRIO on PXI.

Q: I thought LabVIEW was a test and measurement tool. Is it also being used for design?

LabVIEW has been used as a design tool since it was invented in the 1980s.The original graphical dataflow semantics of LabVIEW turned out to be a great way to design a wide range of systems from large telescope control to analytical instruments. The vast LabVIEW IP for advanced math, analysis and signal processing including specific tools like the LabVIEW Digital Filter Design Toolkit drastically reduce the development time of these systems.

In the last 10 years, the design capability of LabVIEW has grown to deliver the power of the software to embedded real-time hardware. We have introduced technologies such as the LabVIEW FPGA Module that enables users to design hardware logic with graphical programming. These capabilities are leading engineers to choose LabVIEW for designing systems in a variety of disciplines including medical devices, robotics, energy technology and communications.

NATIONAL INSTRUMENTS™

11500 N Mopac Expwy • Austin, TX 78759-3504 USA • Tel: (800) 433 3488 • Fax: (800) 683 9300 • info@ni.com • ni.com