Hardware-in-the-Loop (HIL) Test System Architectures

Publish Date: Nov 13, 2009

Overview

HIL simulation is a powerful test method you can use to test embedded control systems more efficiently. When testing embedded control systems, safety, availability, or cost considerations can make it impractical to perform all of the necessary testing using the complete system. You can use HIL simulation to simulate the parts of the system that pose these challenges, which gives you the power to thoroughly test the embedded control device in a virtual environment before proceeding to real-world tests of the complete system. With this capability, you can maintain reliability and time-to-market requirements in a cost-effective manner even as the systems you are testing become more complex. To learn more about how HIL testing improves control system validation, watch the What Is HIL Testing webcast. This tutorial discusses a variety of HIL test system architectures and how to implement them.

Table of Contents

1. Components of an HIL Test System

An HIL test system consists of three primary components: a real-time processor, I/O interfaces, and an operator interface. The **real-time processor** is the core of the HIL test system. It provides deterministic execution of most of the HIL test system components such as hardware I/O communication, data logging, stimulus generation, and model execution. A real-time system is typically necessary to provide an accurate simulation of the parts of the system that are not physically present as part of the test. The **I/O interfaces** are analog, digital, and bus signals that interact with the unit under test. You can use them to produce stimulus signals, acquire data for logging and analysis, and provide the sensor/actuator interactions between the electronic control unit (ECU) being tested and the virtual environment being simulated by the model. The **operator interface** communicates with the real-time processor to provide test commands and visualization. Often, this component also provides configuration management, test automation, analysis, and reporting tasks.

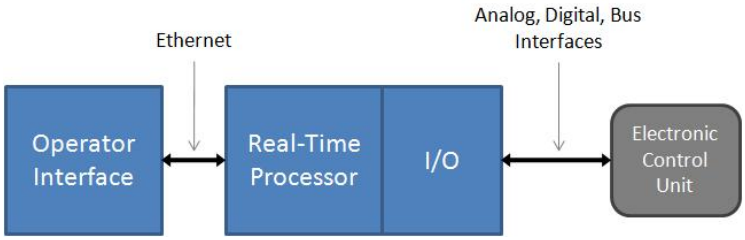


Figure 1. An HIL test system consists of three primary components: an operator interface, a real-time processor, and I/O interfaces.

2. Hardware Fault Insertion

Many HIL test systems use hardware fault insertion to create signal faults between the ECU and the rest of the system to test, characterize, or validate the behavior of the device under these conditions. To accomplish this, you can insert fault insertion units (FIUs) between the I/O interfaces and the ECU to allow the HIL test system to switch the interface signals between normal operation and fault conditions such as a short-to-ground or open circuit.

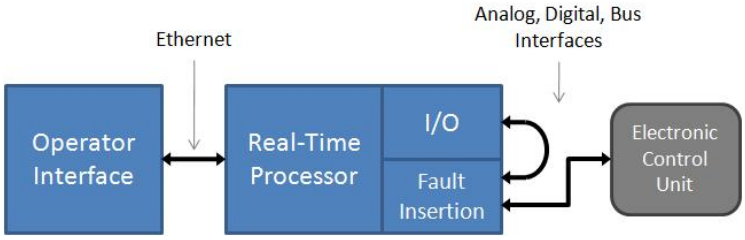


Figure 2. You can use hardware fault insertion to test the behavior of the ECU during signal faults.

3. Testing Multi-ECU Systems

Some embedded control systems, such as an automobile, aircraft, or wind farm, use multiple ECUs that are often networked together to function cohesively. Although each of these ECUs may initially be tested independently, a system's integration HIL test system, such as a full vehicle simulator or iron bird simulator, is often used to provide more complete virtual testing.

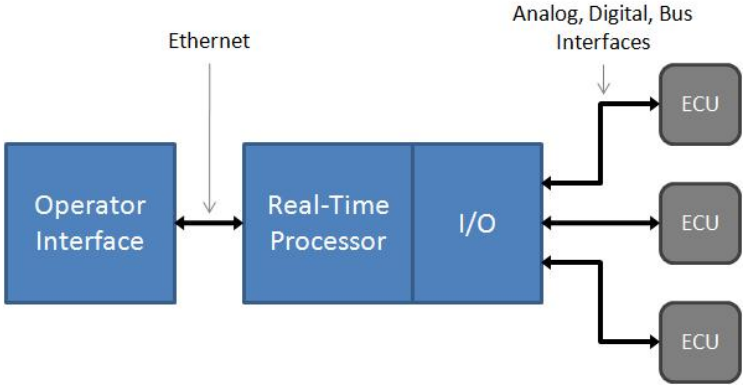


Figure 3. Automobiles, aircraft, and wind farms use multiple ECUs.

When testing a multi-ECU control system (and even some single ECU control systems), two needs often arise: **additional processing power** and **simplified wiring**.

4. Additional Processing Power – Distributed Processing

Even with the latest multicore processing power, some systems require more processing power than what is available in a single chassis. To address this challenge, you can use distributed processing techniques to meet the performance requirements of these systems. In very high-channel-count systems, the need is more than simply additional processing power, additional I/O is also necessary. In contrast, systems using large, processor-hungry models often use additional chassis only for the extra processing power, allowing those processors to remain dedicated to a single task for greater efficiency. Depending on how the simulator tasks are distributed, it may be necessary to provide shared trigger and timing signals between the chassis as well as deterministic data mirroring to allow them to operate cohesively.

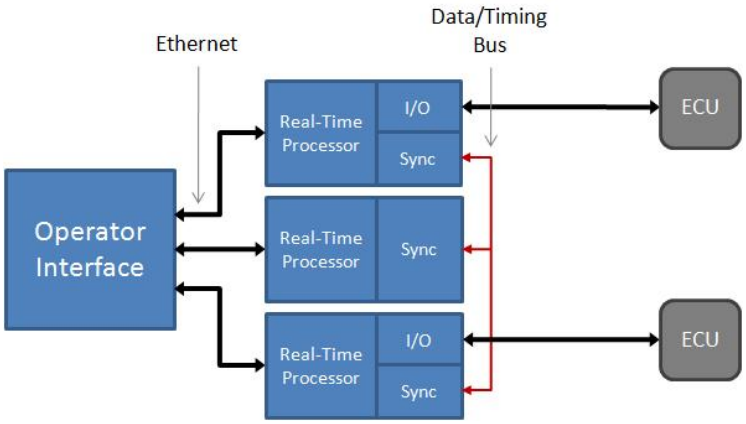


Figure 4. When using multiple chassis for additional processing power, it is often necessary to provide timing and data synchronization interfaces between them.

5. Simplified Wiring – Distributed I/O

Implementing and maintaining wiring for high-channel-count systems can pose costly and time-consuming challenges. These systems can require hundreds to thousands of signals be connected between the ECU and the HIL test system, often spanning many meters to compensate for space requirements.

Fortunately, deterministic distributed I/O technologies can help you tame these wiring complexities and provide modular connectivity to ECUs, which allows for efficient system configuration modifications. Instead of routing all connections back to a single rack containing one or more real-time processing chassis instrumented with I/O interfaces, you can use deterministic distributed I/O to provide modular I/O interfaces located in close proximity to each ECU without sacrificing the high-speed determinism necessary for accurate simulation of the virtual parts of the system.

This approach greatly reduces HIL test system wiring cost and complexity by making it possible for the connections between the ECU and the I/O interfaces to be made locally (spanning less than a meter) while a single bus cable is used to span the additional distance to the real-time processing chassis. Additionally, with the modular nature of this approach, HIL test systems can easily scale, incrementally, from a multi-ECU test system in which all but one of the ECUs are simulated to a complete systems integration HIL test system where none of the ECUs are simulated.

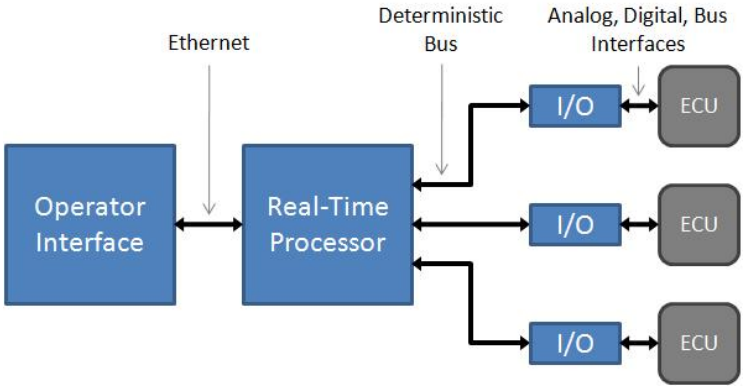


Figure 5. Deterministic distributed I/O interfaces greatly reduce HIL test system wiring cost and complexity because the connections between the ECU and the I/O interfaces can be made locally.

6. Implementing HIL Test Systems

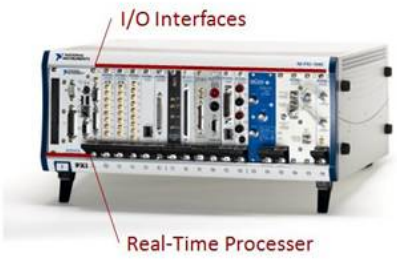
Once you have selected the appropriate architecture for your HIL test system, the first step in creating a HIL test system is to select the real-time processing component(s) that best meets your development requirements. National Instruments provides a wide variety of real-time processing options for implementing HIL test systems. Because they are all based on open industry standards, you can be assured that they always deliver the latest advances in PC technology to your HIL test system and always meet future test system requirements.

PXI is an open, PC-based platform for test, measurement, and control. It offers a wide variety of real-time processor options including several high-performance dual- and quad-core processors. With more than 1,200 products from more than 70 vendors, PXI is the platform of choice for thousands of companies worldwide.

The PXI platform works with many synchronization technologies, including IRIG-B, IEEE 1588, SCRAMNet, and reflective memory for sharing timing, trigger, and data in multichassis HIL test systems.

National Instruments also provides several options for implementing minimal-cost, small-footprint HIL test systems. NI **CompactRIO** is a low-cost reconfigurable control and acquisition system. The system combines small size with an open embedded architecture powered by reconfigurable I/O (RIO) field-programmable gate array (FPGA) technology.

This technology combines a real-time processor with a user-programmable FPGA that you can use to create custom I/O personalities as well as to off-load model execution and signal processing from the real-time processor for increased HIL test system performance.



7. Summary

After determining and selecting the appropriate real-time processing component(s) necessary to meet your HIL test system needs, you need to select the I/O required to interface to your ECU(s). Read Selecting Hardware-in-the-Loop (HIL) Test System I/O Interfaces to learn more about the I/O interfaces available for your HIL test system.

Find additional resources to assist you with your HIL test system development or learn how others have achieved success with the NI HIL platform at ni.com/hil.