

HIL Simulation Checklist

Scope:

- Would it be beneficial to have open-loop capabilities as well?
- Will there be more than 1 ECU configuration tested with this system (i.e. 4-cyl & 8-cyl engine simulation)?
- Will there be any environmental fixtures (temperature, electro-magnetic, vibration ...)?
 - What connectivity impact does this have on the connection between the test system and the ECU?
 - Does the chamber require control and/or command signals from the test system?
- Will there be multiple ECUs involved?

Enterprise:

- Requirements Traceability – Are there requirements documents that should be linked to this test system?
 - Is there an existing requirements management utility?
- Source Code Control – Do the models, test scripts/sequences, or other software need source code control?
 - Is there an existing source code control utility?
- Data Management – How will test data be managed?
 - What format will the data be stored in?
 - Is there a need for data mining (searchable data)?
 - Is there an existing data management system?
- Alerts & Notifications – Is there a need for alerts to be automatically sent based on events in the test system via email, web, or text message?
- User Management – Does there need to be access restrictions for different users?
 - What access groups will be necessary?

Test Specification:

- Profiles – What are the test profile requirements?
 - Is streaming a data file to the model required (data playback)?
 - What file format(s) need to be supported?
 - Does the test system need to be able to generate profiles (data generation)?
 - What types of profiles need to be generated (sinusoid, noise ...)?
 - Are there any closed-loop control requirements that are not included in the model (i.e. vehicle speed profile controlled via throttle input to model)?
- Analysis – What are the analysis requirements?
 - Does analysis need to be completed at run-time (in-line as opposed to post-processing)?
 - What types of analysis will be performed?
 - Are there existing analysis routines?
 - Are external packages required?
- Reporting – What are the reporting requirements?
 - What information must be included in reports (raw data, processed data, test criteria, test results)?
 - What format are the reports to be delivered in (pdf, html ...)?
 - Are example reports available?
- Automation – What automation is necessary?
 - Test configuration (model connection to UI and IO)?
 - Analysis & reporting?
 - Test sequences?
 - Are there existing sequences?
 - What format are they in (TestStand, Python, custom ...)?
 - Is there conditional branching (flow control) in the sequences?
 - Do the sequences need to be executed deterministically (in a real-time environment)?
 - Does the test system need to perform a special action when a fault occurs (abort, modify logging, re-test ...)?

Data Logging:

- How many channels must be logged?
- Are multiple data log files required?
- What rate(s) do the files need to be logged at?
- Is data log triggering required (conditional logging)?

User-Interface:

- What types of controls and indicators are required for the display (numeric indicators, meters, graphs, instrument cluster, tables, 3D plots, *.vrml models ...)?
- Are custom GUI objects necessary?
- What functionality is necessary from the user interface?
 - Manual Fault Insertion?
 - Access to external interfaces (calibration/diagnostic tools)?
 - Ability to add new controls/indicators at run-time?
 - Test sequence tracing?
 - Configuration capabilities?

Simulation:

- What format is the model (programming language, modeling language, state chart)?
- How fast will the model be required to execute (loop rate or step size)?
- What is the model size?
 - How many parameters must be accessed?
 - How many I/O ports are there?
 - What is the raw execution time of the model?
 - Is the model multi-rate?
- Can the model be run in real-time (is it deterministic)?
- Does the system require the ability to execute multiple models at the same time (driver model, environmental model, additional plant models ...)?
- Is there an example of the model available?
- Are there other nodes in the ECU network that must be simulated (*Restbus* simulation)?

I/O Interfaces:

- Is there existing hardware interfaces that need to be utilized?
 - What type of interface do they use (PCI, PXI, USB, VME, VXI, GPIB, Ethernet ...)?
- Is sensor simulation necessary (LVDT/resolver, RTD/TC, Force/Flow ...)?
- Is actuator pre-processing necessary (filtering, linearization, scaling ...)?
- Are custom digital protocols necessary?
- Channel Details – How many channels are needed including simulation I/O and stimulus/logging I/O (example table below)?
 - Type of channel (AIO, DIO, counter, frequency, PWM ...)?
 - Sample/Update Rate?
 - Range (AC or DC, max/min voltage or current,)?
 - Resolution (1mV, 1kHz, 1mA ...)?
 - Is isolation necessary?
 - How much current needs to be sourced/sinked?
 - Is loading necessary?
 - Does the load need to be simulated or will the actual component be used?
 - What wattage is necessary?
 - Does the loading need to be dynamic?
- What communication bus interfaces are necessary (CAN, LIN, K-line, FlexRay, MOST, ARINC-429, MIL-STD-1553, SCRAMNET+, AFDX, PROFIBUS, Serial, DeviceNet ...)?
- How many messages will there be?
 - What messages are required (including configuration information specifying arbitration, scaling, and decoding)?
 - What rate(s) do the messages need to be sent at?

Other Interfaces:

- What other interfaces are necessary?
 - Calibration tools
 - Diagnostics tools
 - Vision tools for inspection
 - Custom Instrumentation
- How do these interfaces connect to the HIL test system
 - Hardware (GPIB, USB, Serial, digital protocol ...)
 - Software (ActiveX, driver ...)

Fault-Insertion:

- What type of faults are necessary (short to ground/battery/channel, open, short through resistance)?
- What fault coverage is necessary?
- What are the voltage and current requirements for the faults?
- Are the faults simulated (in software) or physical?
- Does the fault insertion need to be deterministic (driven from the simulator)?

Connectivity:

- How many physical connectors are there and how will the UUT interfaced with these connectors?
- Is a custom connector needed?
- Are there any cable length requirements from the UUT to the HIL system?
- How much current must the connection be able to handle?
- Is a break-out panel needed?

Is Your HIL Platform Open?

With the National Instruments platform you can create powerful hardware-in-the-loop (HIL) simulators for your validation and testing needs. The combination of open software and modular hardware empowers you to easily adapt your HIL system to new requirements or configurations, providing a flexible, cost-effective alternative to proprietary systems.

Open

- Deterministic execution of any plant model with LabVIEW Real-Time
- Integrate any signal or protocol with support for multi-vendor I/O on the PXI standard (www.pxisa.org)

Flexible

- Easily adapt to new configurations with modular PXI hardware
- Simulate any sensor or protocol with graphical FPGA programming and NI Reconfigurable I/O (RIO) hardware

Cost-Effective

- Take advantage of the availability and value of COTS hardware
- Reuse your software and hardware investment for future applications

