The Importance of Test Component Reuse in Testing Safety Compliant Systems

The Importance of Test Component Reuse in Testing Safety Compliant Systems

Whether it is in an automobile, commercial airplane, medical device, or a number of other devices in a variety of industries, there has been a significant recent growth in electronic complexity and the amount of embedded software in products. Often, many of these control systems play a critical role in safety, thus placing an increased importance on the confidence of the embedded software to behave as desired without bugs and defects that could potentially be life-threatening. Several industries are addressing this concern through the use of functional safety standards, which introduce a process-oriented approach to developing electronic systems with safety in mind. These include: DO-178B/C for avionics, IEC 61508 for industrial safety, ISO 26262 for automotive, IEC 62304 for medical devices, and many more. However, the need to follow a process while the amount of software is experiencing exponential growth can be intimidating, especially when trying to test and validate the software and its abilities.

One methodology that can help to ensure efficient and complete testing while also saving time and cost is the reuse of test components throughout the development process. Whether it is in the testing of the actual generated source code for the device, or performing system-level testing of the hardware with the code, the ability to reuse test components throughout the development cycle can save a significant amount of time. Throughout this paper, we will talk about how both National Instruments and the LDRA tool suite can be used to implement test component reuse and how this can benefit development of products for safety critical systems. The National Instruments test platform is ideal for performing system-level testing while the LDRA tool suite® is designed for embedded code testing. Through the use of both toolsets in a complimentary way, test component reuse can be achieved throughout the full development cycle.

Test Component Reuse and System-Level Testing

When developing electronic systems that perform critical operations such as safety-related decisions, there are different types of testing that take place throughout the development lifecycle. For example, let's look at the process for developing an electronic control unit (ECU) for a passenger car. There are a variety of testing stages that take place including:

- Model-in-the-loop (MIL) testing for a fully simulated system,
- Rapid control prototyping when fine-tuning the control algorithm,
- Hardware-in-the-loop (HIL) simulation for simulating the world around the ECU, and
- Physical testing in a real-world environment through the use of elements such as a dynamometer or environmental chamber.

Although showing traceability to requirements is always important in software development, it is especially critical when working on projects concerned with functional safety. A fundamental practice required by these standards is to show how all tests and test results trace back to safety-specific requirements. The graphic below shows how the different stages of testing map to the traditional V cycle view of electronics development.

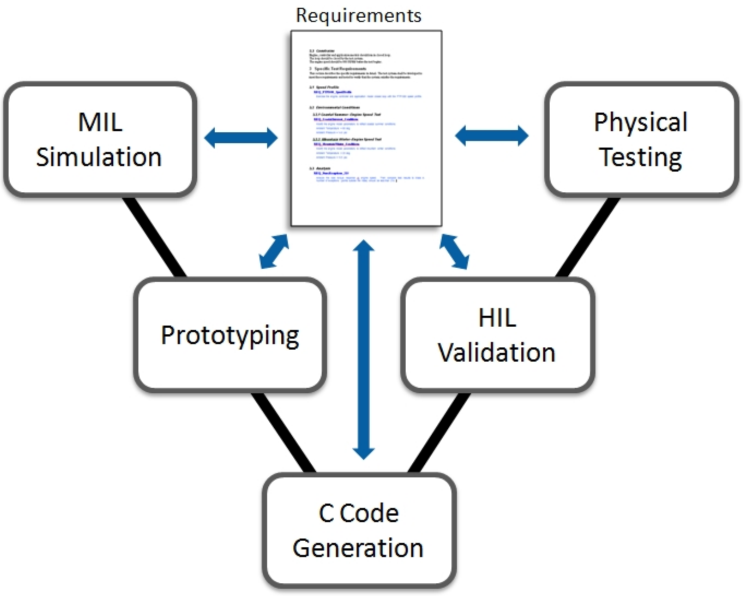

Figure 1: All stages of system-level testing throughout the development cycle must trace back to requirements.

As the testing requirements change, it is important to have a platform that can adapt as needed and has the flexibility to allow companies to seamlessly move from one stage to the next without extra effort. This is where the idea of test component reuse can prove very beneficial. In all of the stages outlined above, there are a variety of test components used in each, including models, stimulus profiles, user interfaces, reports, and data logging. It is also clear that many of these components can be used in multiple steps in order to avoid duplicate work and to ensure continuity.

For example, the same plant model can be used in MIL simulation and HIL validation. Having the same user interface and data logging capabilities can make the data analysis and processing easier for understanding test results. There is no need to re-create the components or the functionality if it can be reused.

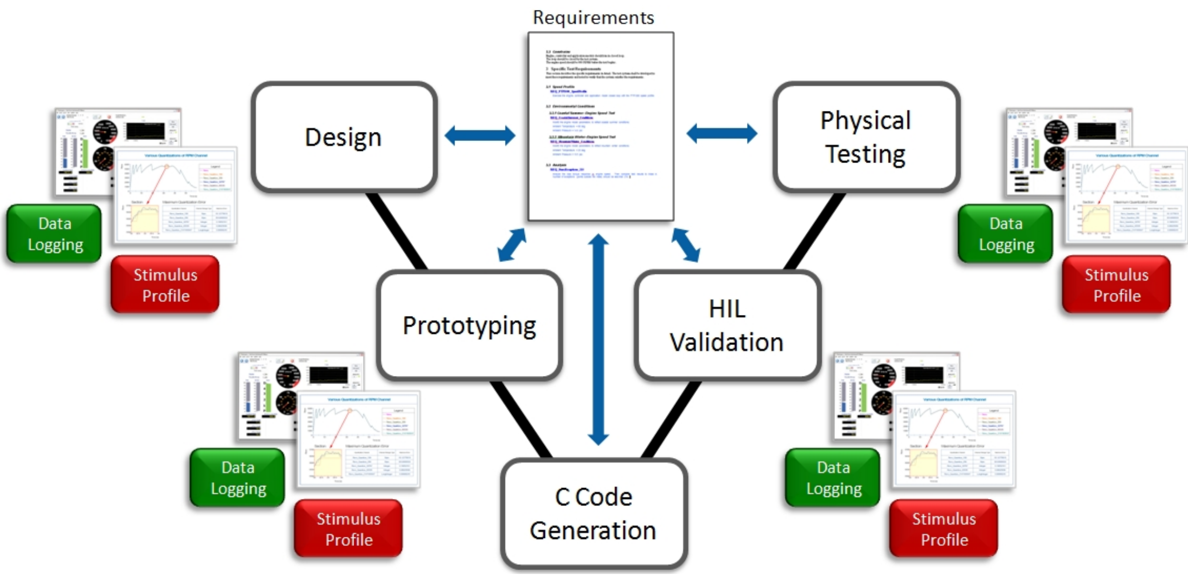

Figure 2: Reusing test components like data logging configurations and stimulus profiles throughout development can save both time and cost of a project.

National Instruments tools are ideal for developing a standardized testing system to perform functional safety tests. NI offers an integrated hardware and software platform that allows producers to quickly develop a testing application that satisfies functional safety requirements. Because NI offers a modular hardware platform, similar testing applications can be scaled to fit the testing requirements of smaller or more elaborate embedded systems. NI has a software defined, reconfigurable platform that allows producers to test a component throughout the design process.

Test Component Reuse and Embedded Code Testing

In addition to system-level testing, unit or module-level testing is extremely beneficial during the development process. These tests can prove functionality of a single function or several functions acting together. As development progresses into higher and higher levels of integration, new tests will be devised to ensure proper functionality, but the previous test cases do not become obsolete and, in fact, they still represent valuable tests against the code. These tests can be used in regression tests at later stages of development to ensure that new modifications have not impacted the inherent functionality of the code.

Other environments, like modeling tools, are commonplace in software development. Tests can be run and code generated based on these models. Model-generated code must be also be thoroughly tested to ensure no errors have been introduced during its creation. By leveraging the testing effort from these environments, and applying it to the generated source code as well as the model, a duplication of time and effort is avoided. This can be accomplished by creating unit-level tests that reuse the tests created within the modeling tool.

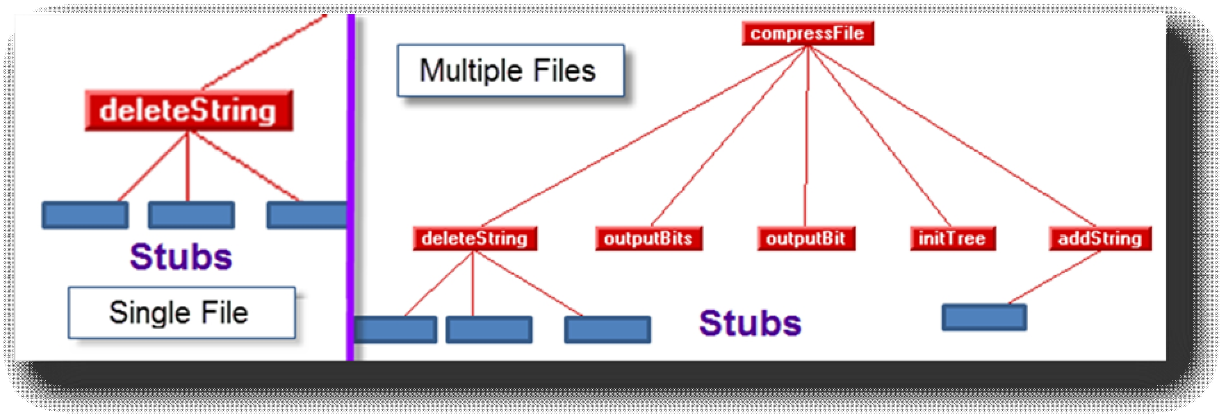

Figure 3: Unit tests can be used to verify the functionality of a single function or a group of functions in isolation. Stubs can be used in place of additional functions.

With the wide range of testing levels, from unit-level to system-level, how can developers be sure the tests being conducted are sufficient? This is accomplished through structural code coverage analysis. Structural code coverage analysis determines which code is executed during a given test run. The level of coverage can range from verifying that all lines of code execute, to verifying that all potential paths through the code have been exercised. The use of code coverage enables developers to assess tests for completeness, discover unreachable or dead code, or identify deficiencies that may exist in test, models, source code, or requirements.

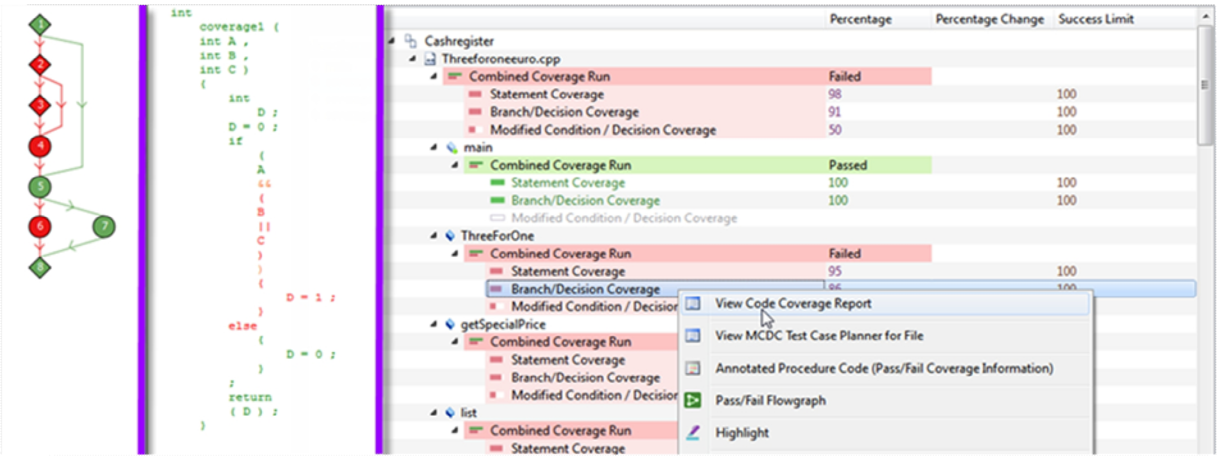

Figure 4: Collecting and reporting on structural code coverage identifies deficiencies in code, testing, or requirements.

The LDRA tool suite can be used throughout development to verify the correct implementation and functionality of developed software. LDRA provides unit-level testing based on a proprietary analysis of the source code. This allows the testers to create tests for a single function or several functions working together, create and maintain stubs and custom settings, and even automatically generate unit-level tests. These tests can then be run on the host or target platform. LDRA also includes the capability to instrument the source code for structural code coverage. This ability saves time at the unit-level as well as the system-level. Any functional unit or system test can be reused with LDRA instrumentation to gather required code coverage and verify that the system executes as expected.

Combining Tools for Test Component Reuse Throughout Development

There is no doubt that creating electronic control systems that must be certified to a functional safety standard requires thorough and consistent testing. This testing is broad in scope, ranging from system-level tests involving embedded software that is controlling actuators within the deployed hardware to embedded code unit-level tests and structural code coverage analysis. National Instruments and LDRA both share the philosophy of maximizing test component reuse throughout the development cycle to reduce the overhead of testing and to save as much time as possible.

For example, consider the process of developing an ECU. When using modeling environments to develop both controller and plant models, model-in-the-loop simulation can be performed using National Instruments tools to perform functional tests on the model capabilities. These same tests can be used in the subsequent stage of rapid control prototyping where only the control model is present and is used to control actual hardware. The same data logging, user interfaces, stimulus profiles, and controller model are reused completely from the previous stage.

Next, the embedded code can be generated from the controller model after functional tests are performed. Here, the LDRA tool suite can be used to generate and execute unit-level tests on the code while also performing code coverage analysis. These metrics are critical to show and to test when developing in compliance with functional safety standards.

The next stage of system-level testing is with hardware-in-the-loop simulation where the embedded code is deployed onto actual hardware and a hardware test system uses a plant model and I/O to simulate the world around the control unit. This is where test component reuse can be leveraged from both National Instruments and the LDRA tool suite. Just like with the system-level testing from before, the same models, stimulus profiles, data logging, and other components used with the National Instruments tools can be reused. The biggest difference is that the plant model is used instead of the control model. National Instruments and LDRA have worked together to also ensure that the system-level platform can be used to perform embedded code testing at the same time. The LDRA tool suite has the ability to instrument models and code that expose the necessary details to see the results of unit-level and structural code coverage tests. These tests can be the exact same ones developed previously, meaning that these test components are also reused throughout development.

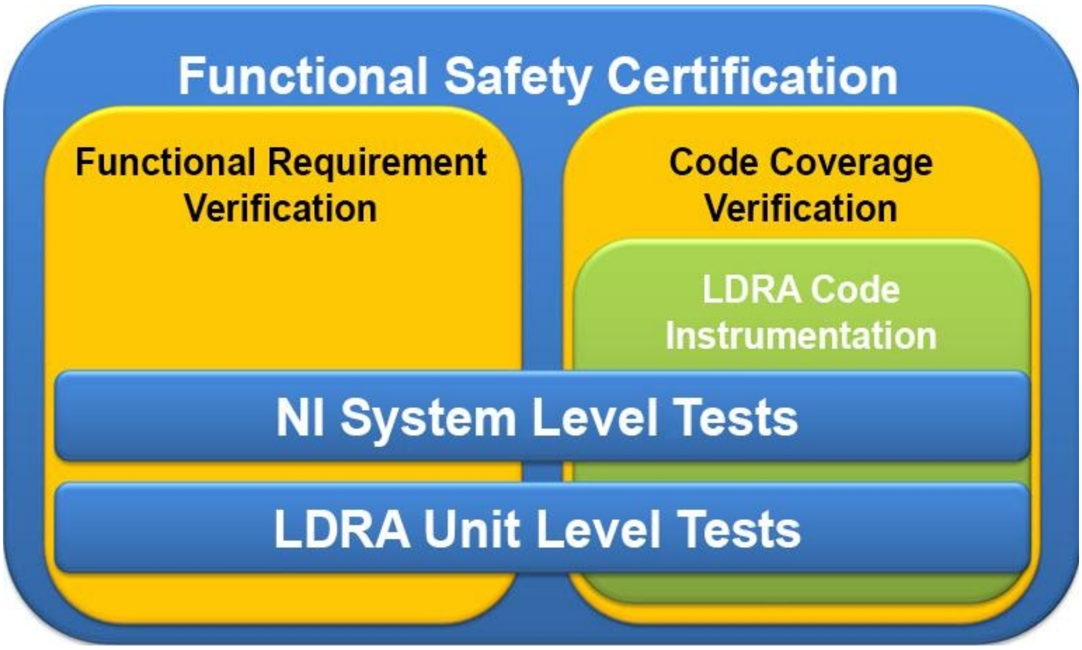

Figure 5: Through the combination of NI and the LDRA tool suite, a single test platform can be used throughout development to show traceability to functional and code coverage requirements.

Test component reuse throughout the software development lifecycle is a valuable tool for the verification of safety critical systems. Tests can be recycled to demonstrate functional correctness as well as ensure completion of safety standards. Use of the National Instruments and the LDRA tool suite can enable the creation and reuse of critical testing components. Through the reduction of the test development phase and maximization of test component reuse the overall time and cost associated with these activities can be greatly reduced.

www.ldra.com

Certificate Number FM 26376

LDRA UK & Worldwide
Portside, Monks Ferry, Wirral, CH41 5LH
Tel: +44 (0)151 649 9300
e-mail: info@ldra.com

LDRA Technology, Inc.
2540 King Arthur Blvd, Suite #228 Lewisville Texas 75056
Tel: +1 (855) 855 5372
e-mail: info@ldra.com

LDRA Technology Pvt. Ltd
#2989/1B, 3rd Floor, 12th Main, 80 Feet Road,
HAL II Stage, Bangalore- 560008. Near BSNL Building
Tel: +91 80 4080 8707
e-mail: india@ldra.com