

VectorCAST/C++™

Unit and Integration Test for C/C++

Test Solutions for C and C++

VectorCAST/C++ is an integrated software test solution that significantly reduces the time, effort, and cost associated with testing C/C++ software components necessary for validating safety and mission-critical embedded systems.

Overview of Advantages

- > Complete test-harness construction for unit and integration testing
- > Test execution from GUI or scripts
- > Code coverage analysis
- > Regression Testing
- > Code complexity calculation
- > Automatic test creation based on decision paths
- > User-defined tests for requirements-based testing
- > Test execution trace and playback to assist in debugging
- > Integrations with best of breed requirements traceability tools
- > Supports Agile and Test Driven Development Methods

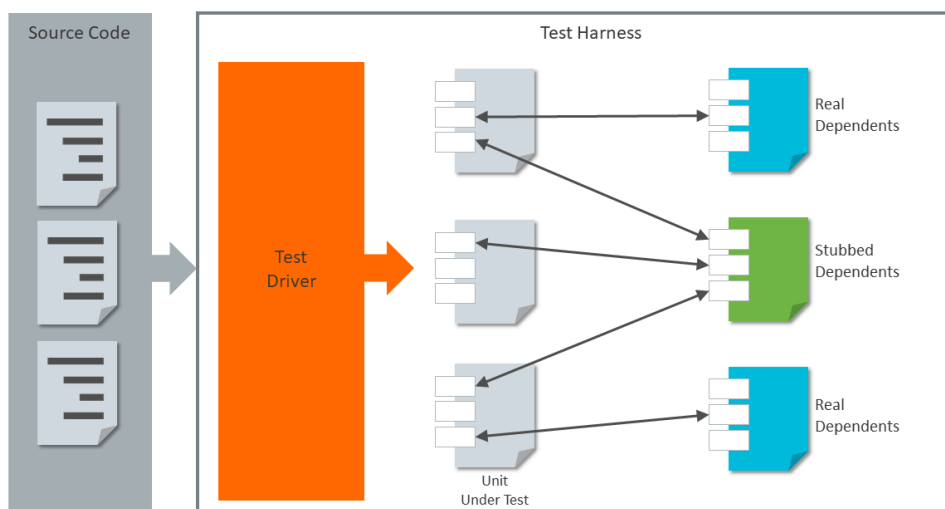
Why VectorCAST/C++

Generally, software component testing requires generating at least one line of test code (in the form of stubs, drivers, and test data) for each line of application code to be tested.

The necessity to create this “disposable” test software is the main reason why manual component testing is so expensive and inefficient. Test software not only has to be written but also has to be debugged to ensure that it performs as expected. With VectorCAST/C++, component testing can be performed without writing a single line of test code.

Highlights

- > Compatible with UNIX, Linux, and Windows Compilers
- > Eliminates Need to Build Test Drivers and Stubs Manually
- > Integrated Code Coverage Capabilities, including MC/DC
- > Supports Host, Simulator, or Embedded Target Testing
- > Automates Regression Testing
- > User Configurable Compiler Interface
- > Supports DO-178B/C, DO-330, ISO 26262, IEC 61508, FDA, IEC 62304, CENELEC Test Requirements



VectorCAST builds test harnesses automatically

Integrated Code Coverage

Without a code coverage tool, it is difficult to determine which portions of the source code have been exercised during testing. VectorCAST/C++ provides an integrated code coverage utility that allows you to gauge the effectiveness of your component testing by reporting on the source code statements or decision points exercised during individual or multiple test runs. Code coverage data can also be shared with VectorCAST/QA to produce combined coverage reports that reflect unit, integration, and system testing.

Testing is Repeatable

Once test cases have been developed, you can use VectorCAST/C++ to automatically run test cases against successive versions of the source code. The management of test execution and the cataloging of test results are automatic. Comparing results of the same test cases against new software versions, prior to system integration, results in fewer surprises caused by "one small change" to a software component.

Supports Integration Testing

Multiple units can be tested in a single VectorCAST/C++ test environment. This allows you to create complex test scenarios that stimulate multiple functions across multiple units.

Compiler Integration

All VectorCAST/C++ generated test harness components are automatically compiled and linked using your compiler. An interface to your compiler's debugger is also provided so that you can run test cases under control of the debugger.

Test Driven Development

VectorCAST/C++ supports Agile and Test Driven Development methodologies. Test case development now becomes the initial activity once the design is complete. This allows you to construct all unit tests prior to any application code being developed. Initially, unit tests will fail due to the absence of source code. But, with the incremental development of code for individual units, the unit tests will begin to pass. The unit test suite can then be regression tested automatically.

Embedded Target Testing

VectorCAST/C++ when used in conjunction with VectorCAST RSP allows testing directly on your embedded target system. VectorCAST RSP is integrated with the cross compiler and RTOS, making it the perfect tool for testing real-time applications. Tests may be developed in a

host environment and then re-executed on an embedded target to verify target and cross-compiler performance.

Integrations

VectorCAST is designed to work with your existing software development toolchain and build system. This allows projects to test in multiple environments and to use the compiler, operating system and hardware appropriate for that environment. This means you can use VectorCAST for both host-based development testing as well as testing in your final embedded hardware target environment.

	PRO	ENTERPRISE
Automatic Test Harness Generation, including complete test driver and intelligent stubs/mocks	●	●
Automated Test Generation	●	●
Type-aware Point and Click Test Editor	●	●
Script-based Regression Testing	●	●
Source Code Instrumentation for Statement, Branch, and MC/DC Coverage	●	●
Mapping between Software Requirements and Test Cases	●	●
Code Coverage Viewer and Reporting	●	●
VectorCAST RSP for Embedded Target Support	●	●
Static Analysis with MISRA checking *	Additional cost	Additional cost
Enterprise Testing Features, including Testing multiple configurations, Python API to access test data, CLI for CI/CT integration including Jenkins		●
Change Based Testing		●
Covered by Analysis (CBA)		●
Probe Points for Fault Injection and White-Box Testing		●
VectorCAST/Analytics		●

More information: www.vector.com/vectorcast