CODESCROLL
CONTROLLER TESTER

Release Note

3.5 — Last update: Jun 14, 2021

Suresofttech

Table of Contents

1. Improvements

Fault Injection view

- You can insert user code before/after nodes in Fault Injection view.
- Marks the location of injected faults with a marker in Source Code Editor.

Class Factory view

- You can generate class codes in Class Factory view.
- You can apply class codes to tests.

Target environment settings

- Target environment settings are improved in Code Composer Studio environment.
- Target environment settings are improved in MPLAB environment.

C++ test

- Test codes and build stubs generated in C++ source codes are improved.

Report

- File coverage report and test stub report are added.
- Connected stub is added in report for each tests.

Navigate source codes

- Possible shortcuts are added in Source Code Editor.
- Context menu related to source code navigation is added in Source Code Editor.

2. Added Features

- [Requirement Traceability](#)
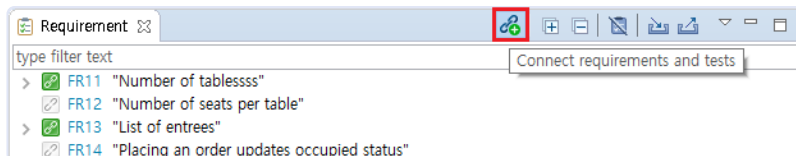- [DISCOVERY Plug-in](#)
- [New Toolchains](#)

2.1. Requirement Traceability

Import/Export requirements

You can import/export requirements from/to Polarion or in CSV file.

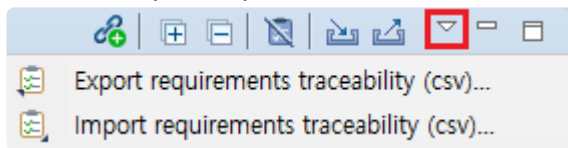## Auto/Manual connection with tests and requirements.

- When importing requirements, tests related to requirements are automatically connected.
- You can connect tests and requirements using [Connect requirements and tests] feature in Requirement view.

- You can connect requirements in Requirement view and tests in Test view using dragging and dropping.

Import/Export requirements traceability

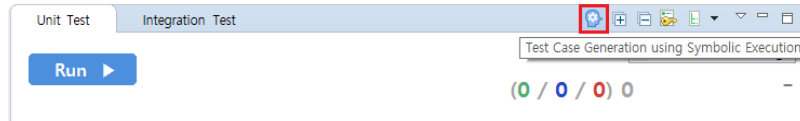- You can import/export information of test connected to requirements in CSV files.

2.2. DISCOVERY Plug-in

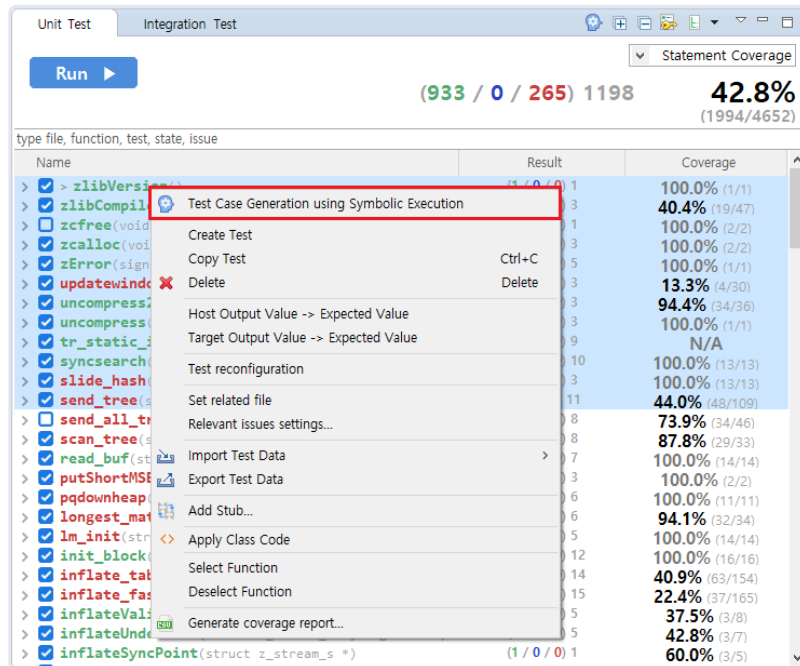You can achieve the coverage goal using [Test Case Generation using Symbolic Execution] feature.

Execute in Unit Test view

[Test Case Generation using Symbolic Execution] feature in Unit Test view allows to create and execute the checked test cases.

Execute in context menu

If you select multiple functions/tests and click [Test Case Generation using Symbolic Execution] in context menu, test cases are added to the selected functions/tests and executed.

2.3. New Toolchains

GCC 9.x and GCC 10.x are added to the list of supported toolchains. GCC 9.x and GCC 10.x toolchains can be added using [Add Toolchain] feature on the toolchain configuration page.

3. Bug Fixed

- Fixed an issue where the branch coverage tooltip was not displayed in Source Code Editor.
- Modify the mapping relationship to be initialized after the test execution is finished, if a test is newly created with the [Reset Test] function when executing a test.
- Fixed an issue that caused a link error when changing the test case after the initial test execution.
- Fixed an issue that test code was not properly generated when there was a template member function in the class.
- Fixed an issue that intermittent failures were occur when performing [Import Stub] in [Stub View].
- Fixed an issue that build with the previous shape if an error occurs due to a defect injection code during code converting.
- Fixed an issue that occurred when apply hexadecimal test data in a lump.
- Fixed an issue that occur when performing integration tests.
- Fixed an issue that not displayed when multiple branches were all covered.
- Fixed an issue that occurred in [Source Code Editor] when adding variable/expression to debug and modifying the source code.
- Fixed an issue that failed when analyzing/executing code containing new C++ feature with Visual Studio 2019 toolchain.
- When setting the project virtual memory address, the virtual memory address is sorted and displayed in the drop-down.
- Fixed an issue where analysis settings were not correctly entered when extracting toolchain info the Visual Studio 2017 toolchain.
- Fixed an issue that a compilation error occurred when selecting [Automatically Generate Test Cases]-[Search-based (CS Labs)].