

Renesas e2 studio Debugger

Abstract

This document describes the usage of the Renesas e2 studio debugger as target system. The minimum required version of e2 studio is 3.0.0.22. TESSY supports the RX controller family.

Table of Contents

1	Renesas e2 studio Debugger	2
1.1	Setup of the e2 studio IDE	2
2	TESSY Environment Settings.....	7
3	Test Run.....	9
3.1	Preparation.....	10
3.1.1	Startup may fail twice	10
3.1.2	Make sure ISServer is registered.....	10
3.2	Interactive Debugging	10

1 Renesas e2 studio Debugger

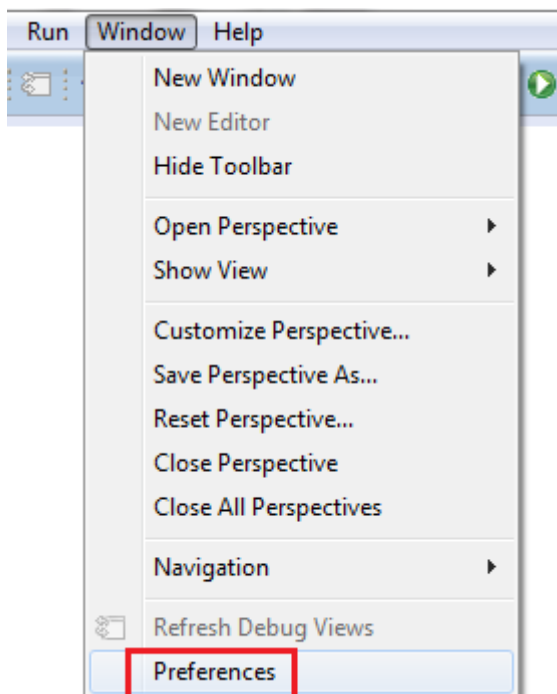
The communication between TESSY and e2 studio is based on the e2 studio Integration Service, which uses the Microsoft Component Object Model (COM). The Integration Service requires the connection to the target to be correctly setup before running tests with TESSY. The section below describes how to perform the initial settings. It is also necessary to configure the eclipse based e2 studio workspace in order to execute the tests with acceptable speed.

The IDE is automatically started before the test run (e.g. batch test) and is deliberately kept open until the test run finishes.

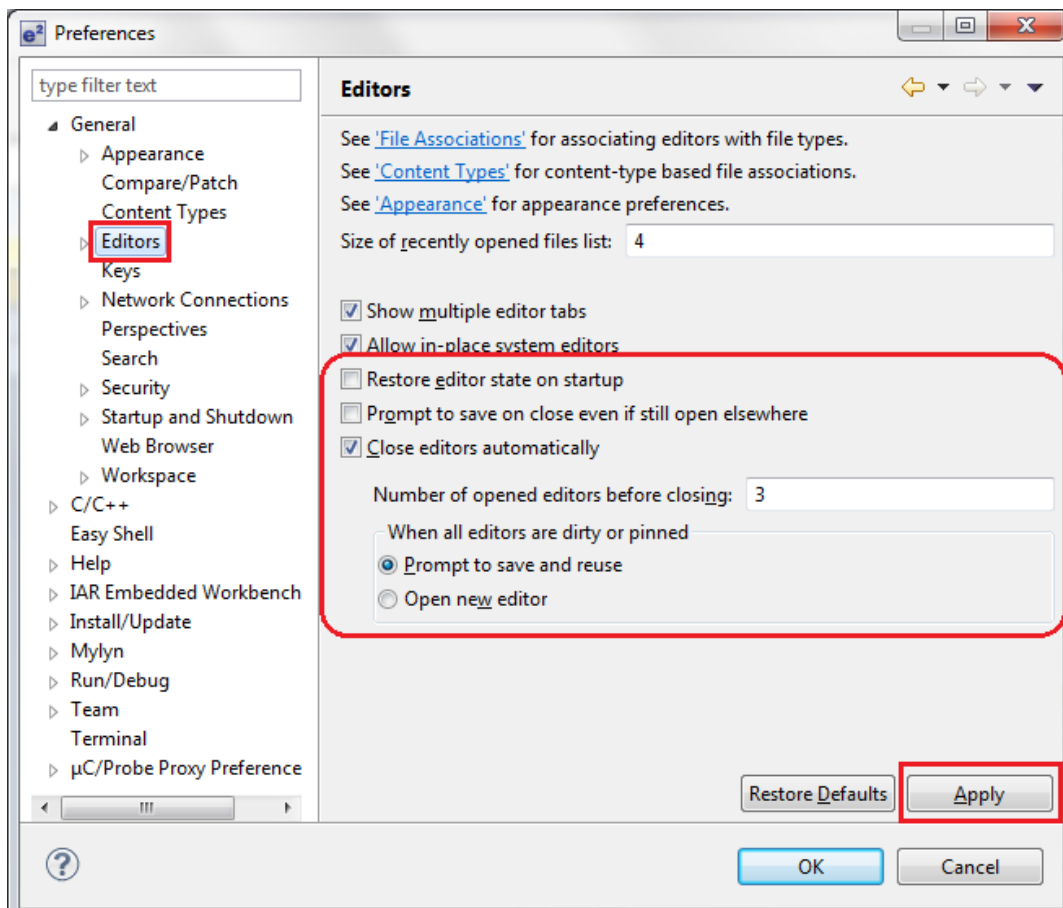
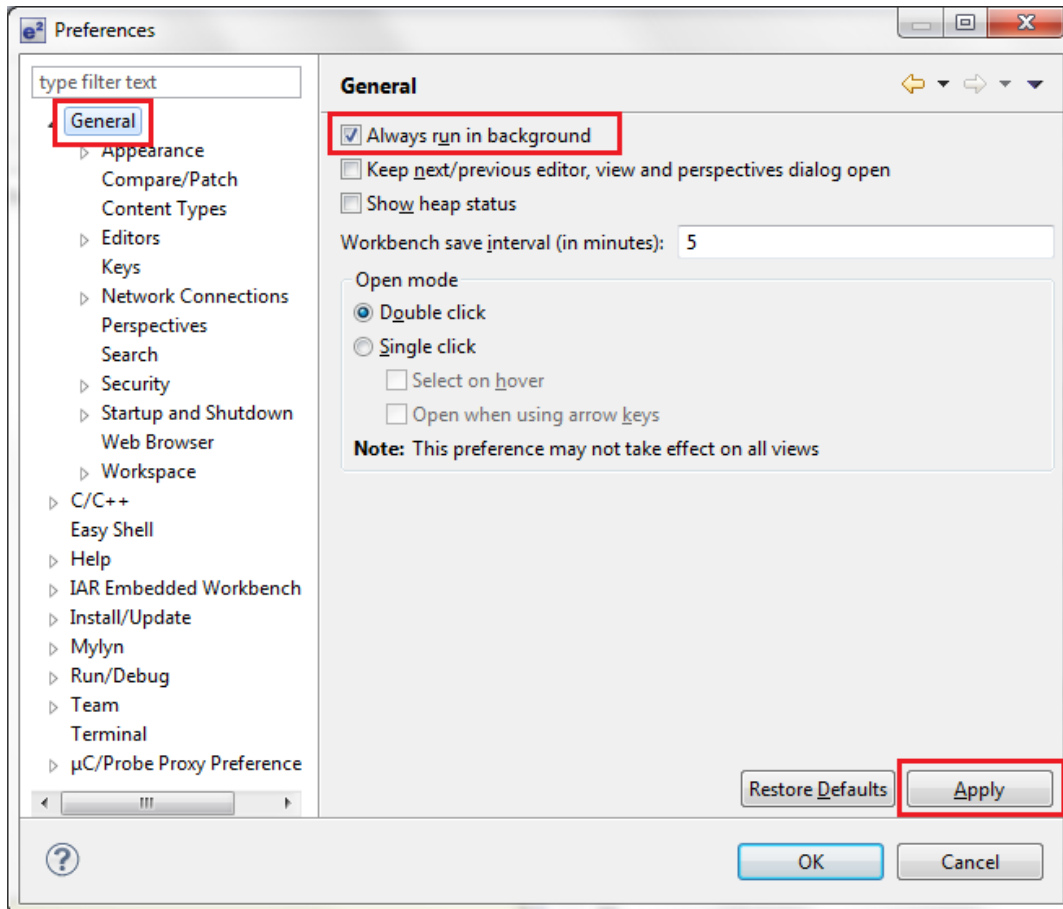
1.1 Setup of the e2 studio IDE

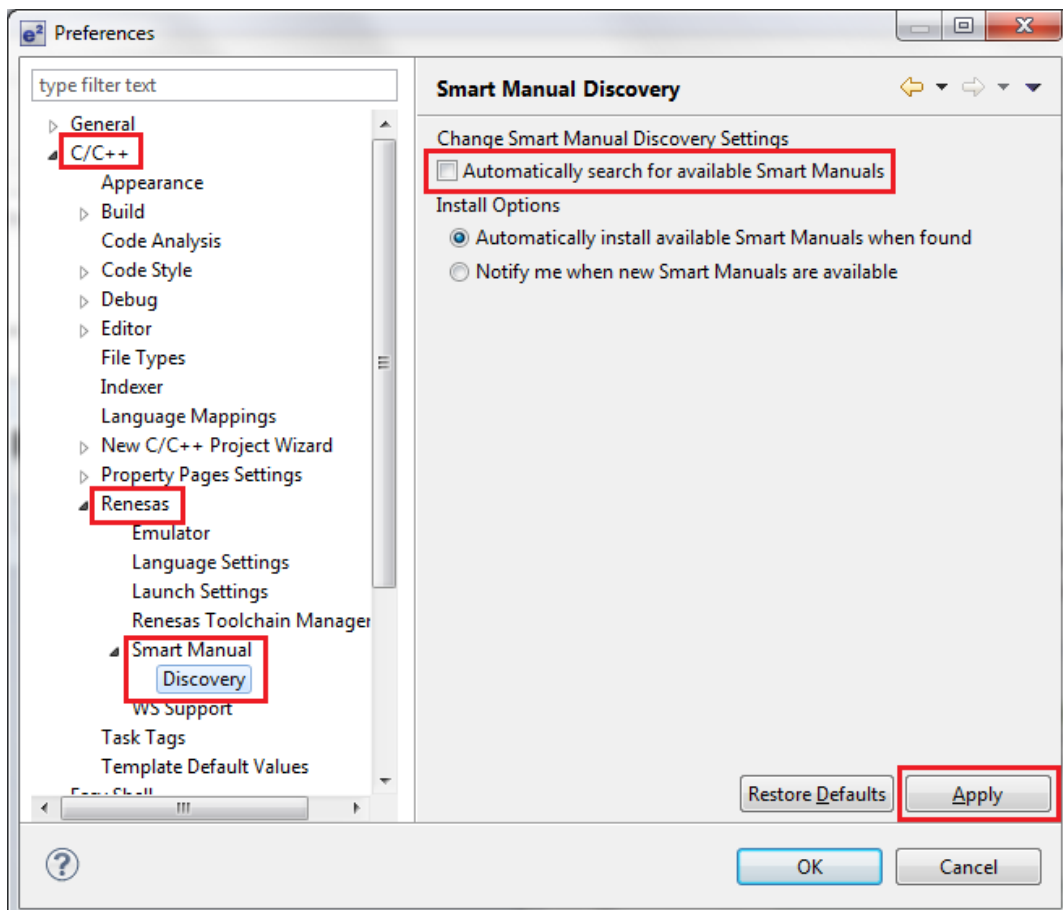
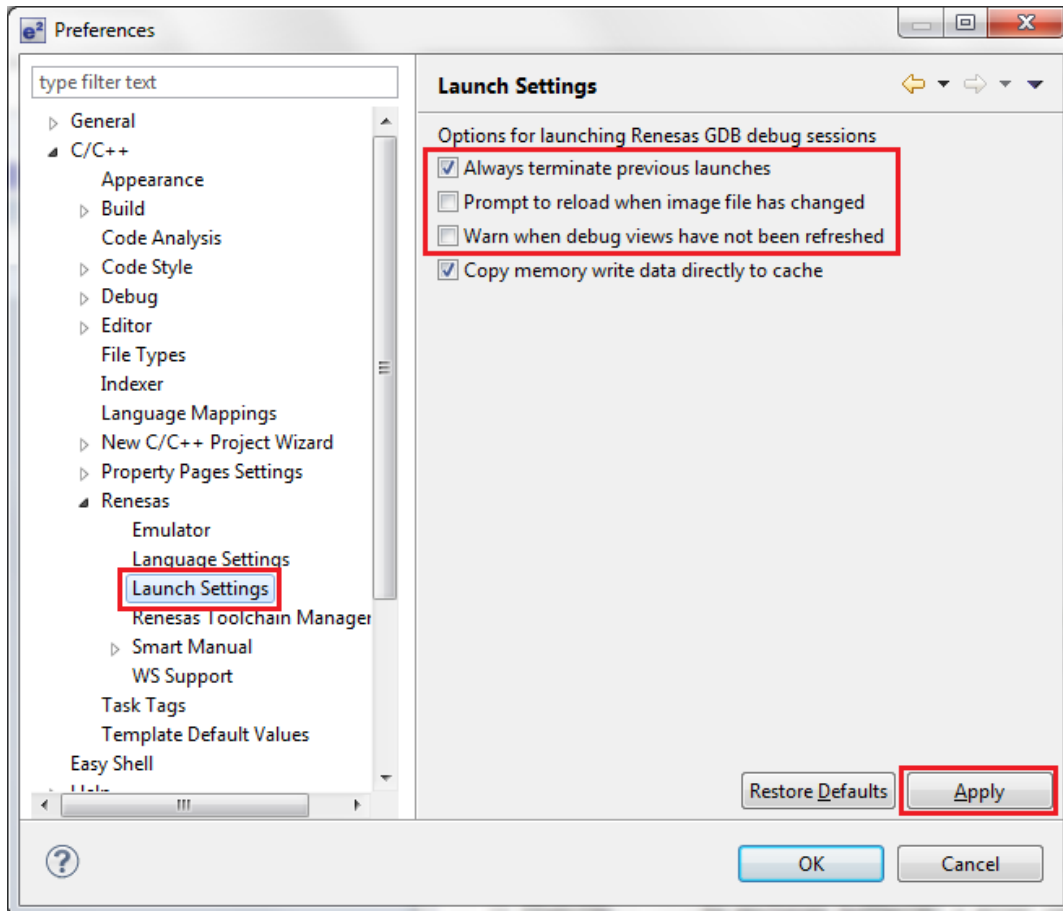
Before running a test from TESSY, please make sure that the debug hardware and the target controller on the target board are working correctly when operated from within e2 studio. You may create a sample project or use your current development project and download and run this project on the debug hardware.

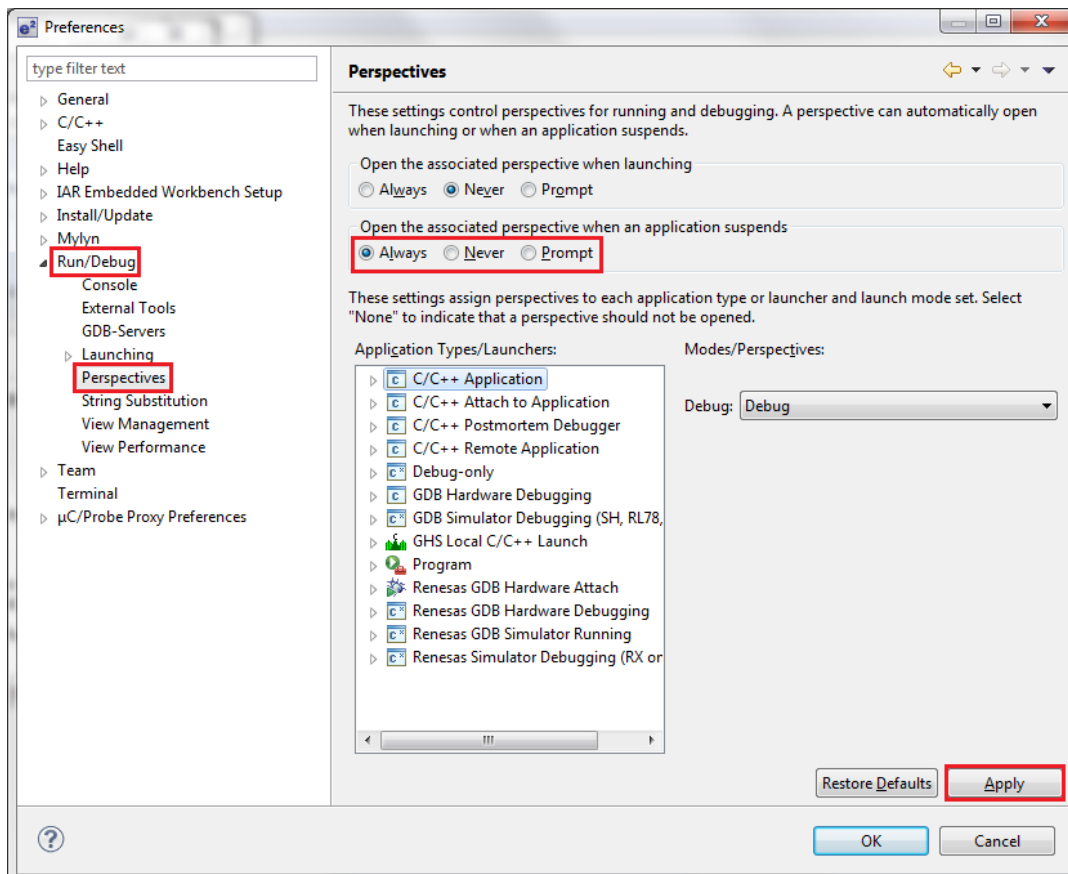
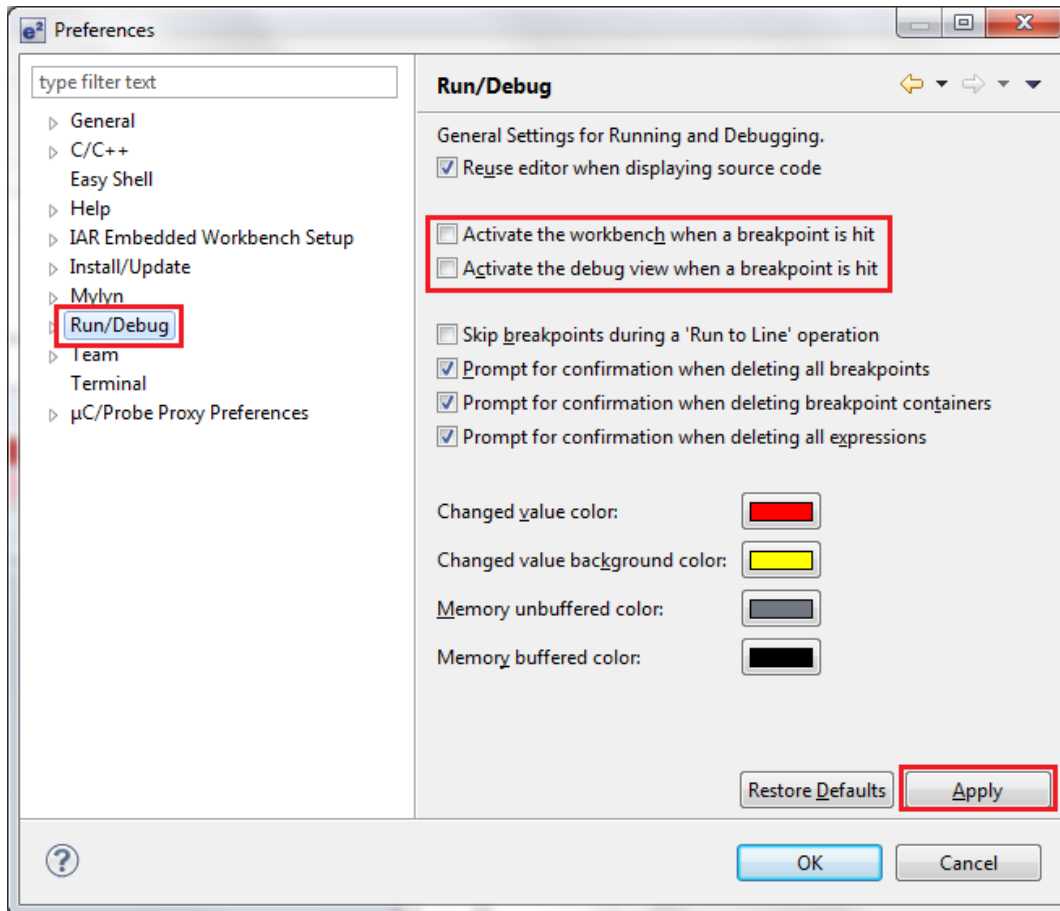
The following e2 studio workspace settings should be applied in order to run the TESSY tests with acceptable speed. At first select the **Debug** perspective and close all editor windows and all redundant views. e2 studio will open editors and views as needed automatically. From e2 studio's **Window** menu choose **Preferences** and compare the screenshots below with your workspace settings.

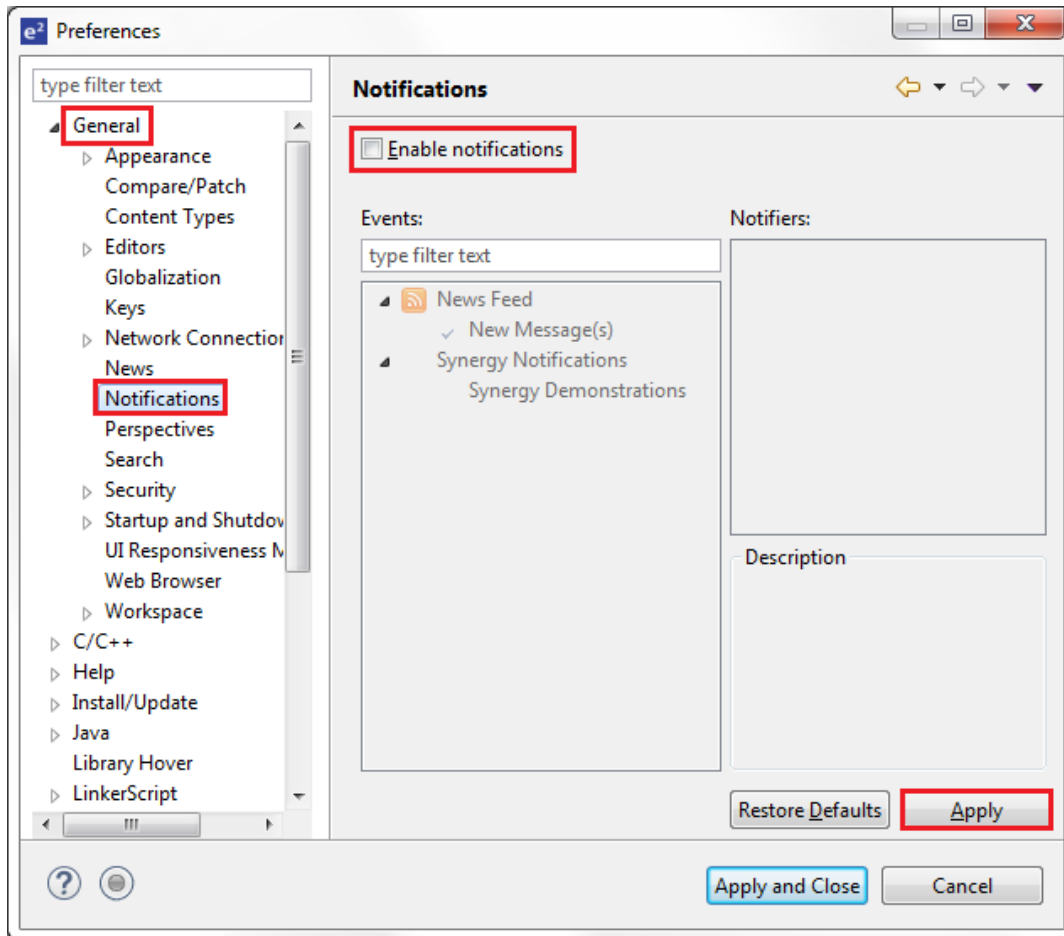


Always click **Apply** before switching to the next section. Finally click **Ok**.

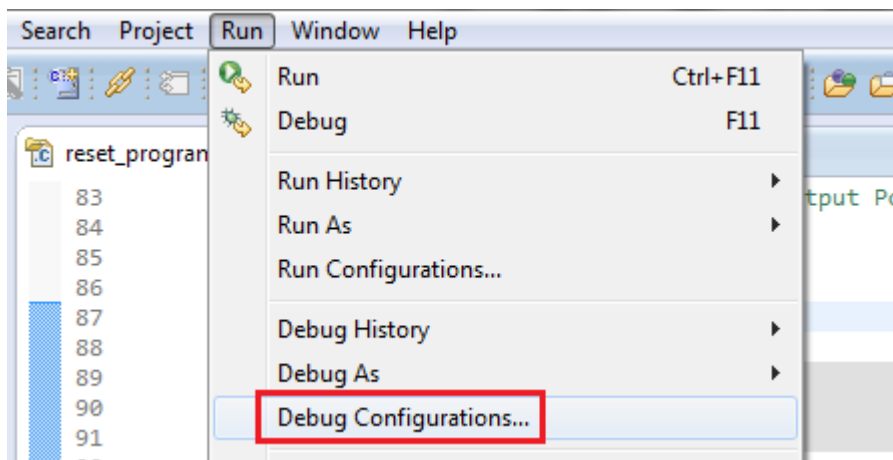


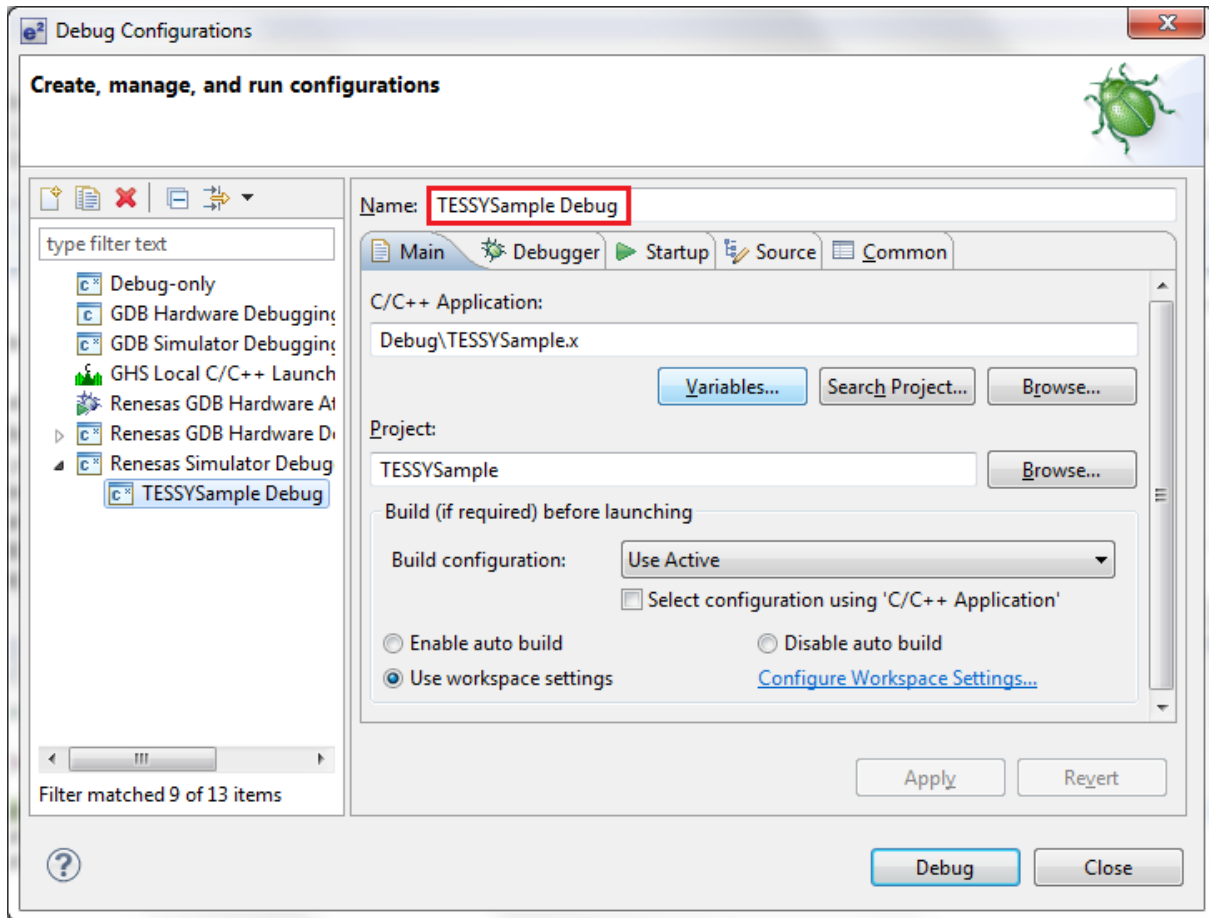






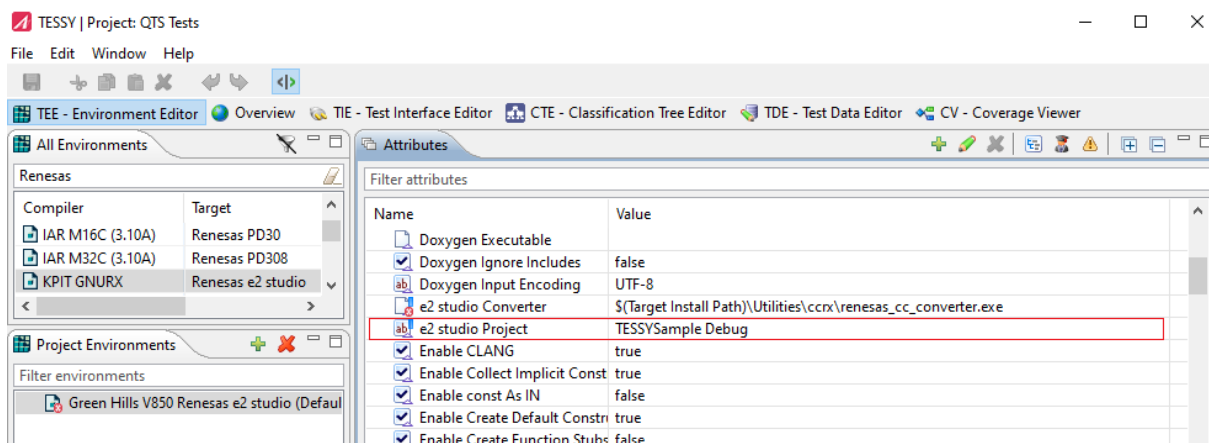
Now open the Debug Configurations dialog and copy the name of the debug configuration you use for your own e2 studio projects. TESSY's target handler will copy the generated target binary to the e2 studio's target binary and start the test run.



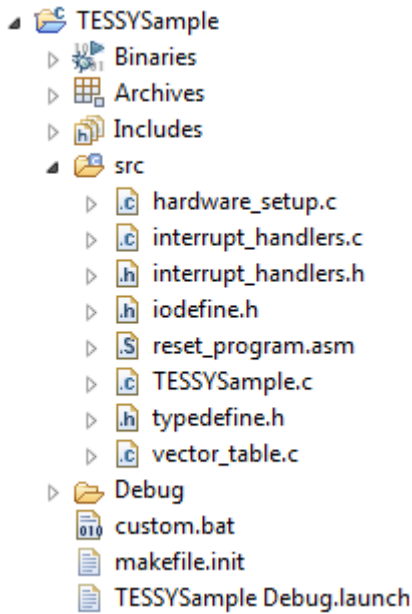


2 TESSY Environment Settings

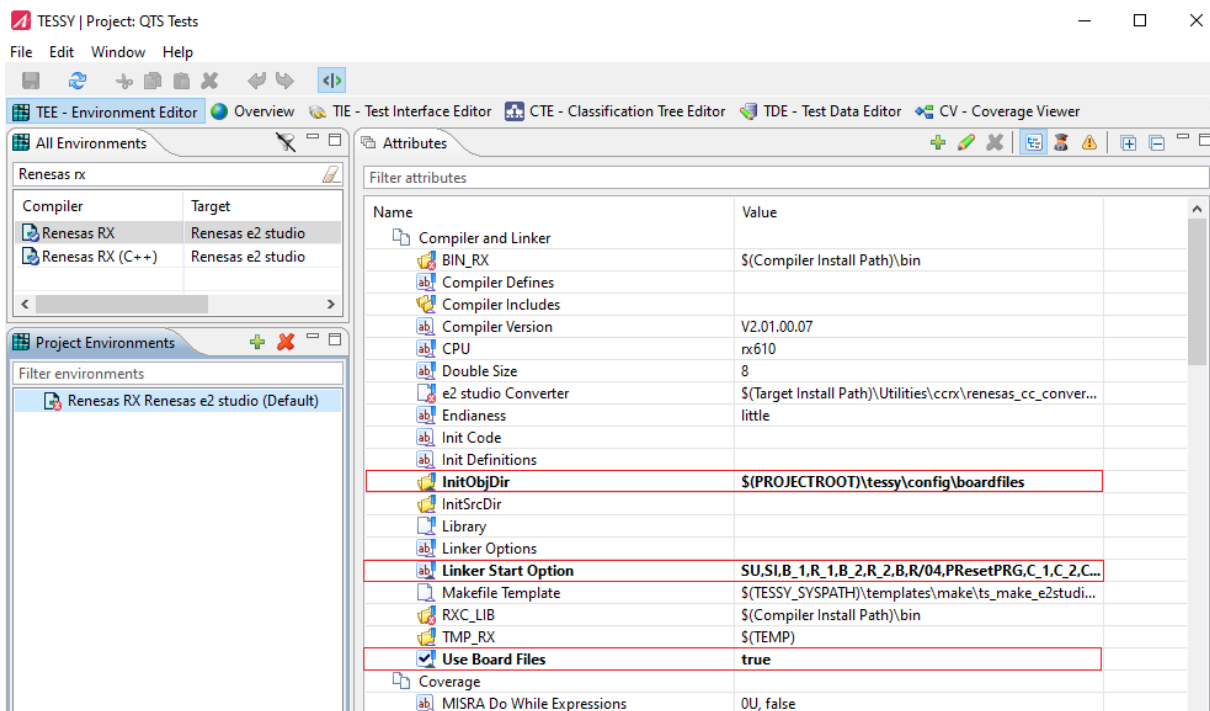
Paste the copied name from the previous section into TESSY's TEE attribute **e2 studio Project**.



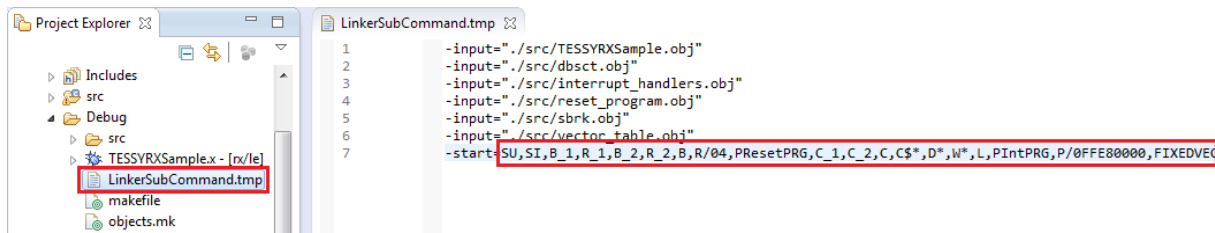
Keep in mind that you will do need board files or startup code, which is found within your e2 studio project's source folder.



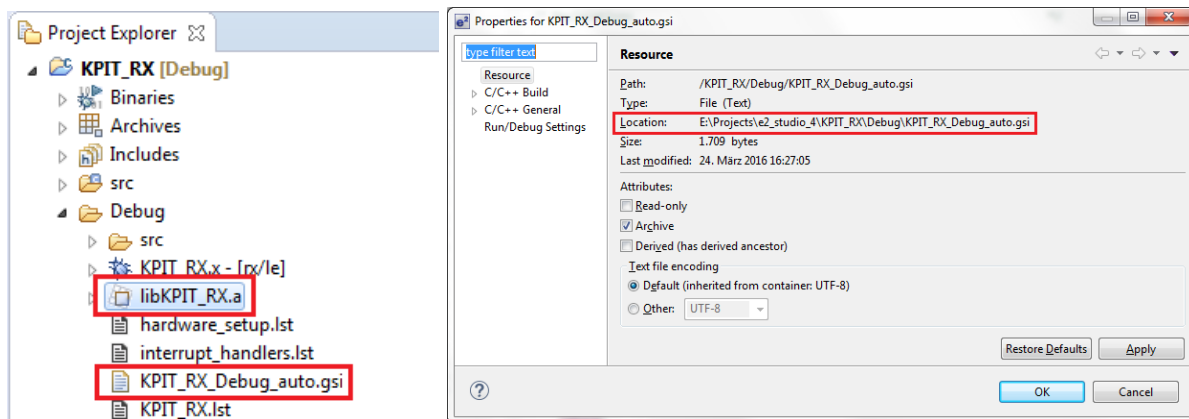
Create a separate object folder where the generated objects will be saved. Specify the paths within TEE attribute **InitObjDir** as shown below. *Keep in mind that you have to update these object files manually if you change the startup code files.*



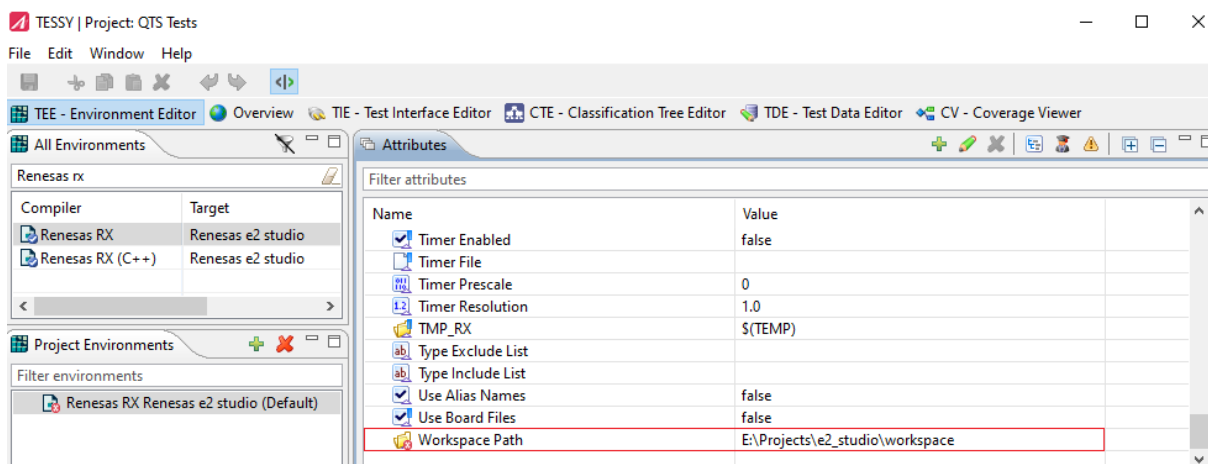
Set the attribute **Use Board Files** to **true** and for the *Renesas RX* compiler do not forget to fill in the attribute **Linker Start Option** as well, which can be found in file **LinkerSubCommand.tmp** in your e2 studio project tree as shown below. Enter the dollar sign twice for each occurrence.



For the *KPIT GNURX* compiler you will need to set the TEE attribute **Generated Library** to the absolute path of the generated library as well as the **Linker File** attribute both of which you will find in your e2 studio project's debug folder as shown below. Right click on the selected entries to open up the context menu and select Properties.



Finally, fill in the e2 studio workspace path within TEE attribute **Workspace Path**.



3 Test Run

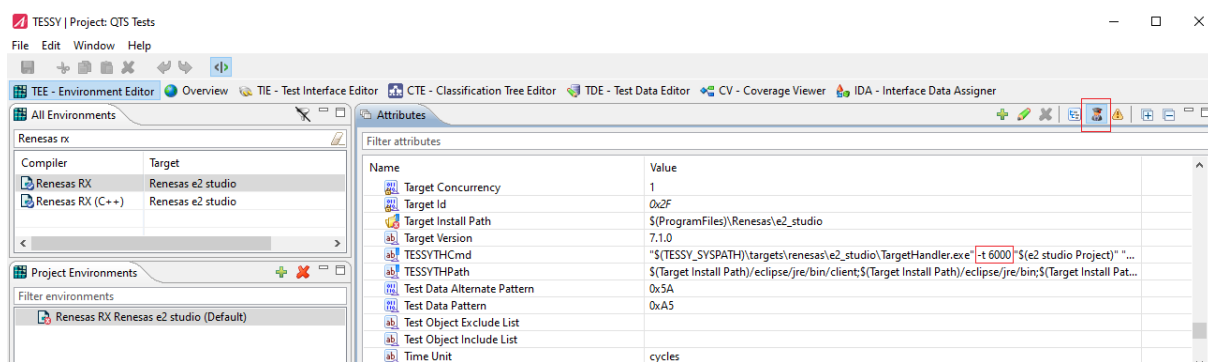
Since the e2 studio Integration Service utilizes the e2 studio IDE during a test run. Interactive debugging is also supported by this target adaption. Before a test will be executed, TESSY's target handler will launch the e2 studio by commands send to the Integration Service and keep it open until TESSY is closed.

3.1 Preparation

The initial startup may fail for two reasons. These are described in the next section.

3.1.1 Startup may fail twice

Unfortunately, the e2 studio debugger API does not signal when it is ready for connections after startup. So TESSY's target handler simply uses a timeout value which is given with the target handler's command line arguments. If the first try to start the IDE fails, the target handler terminates the e2 studio IDE and restarts it. If the second try also fails, you will have to increase the startup timeout value, which is found within the **Expert Mode** TEE attribute **TESSYTHCmd** as shown below. **6000** means six seconds.



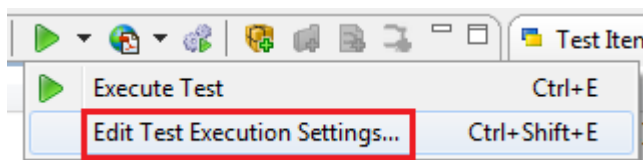
3.1.2 Make sure ISServer is registered

In order to use the ISServer you will have to initially register it with the following command line. **You do need Administrator privileges to run this command!**

```
ISServer.exe /regserver
```

3.2 Interactive Debugging

For interactive debugging open TESSY's **Execute Test** dialog



and select **Define Breakpoint at Test Object** as shown below. Clicking **Execute** will start the test run and let the e2 studio debugger halt at your test object containing your corresponding test data. The global variables `TS_CURRENT_TESTCASE` and `TS_CURRENT_TESTSTEP` contain the test case respectively the test step number.

