

DS-5 Debugger

Abstract

This document describes the usage of the DS-5 debugger as target system. The minimum required version of the DS-5 debugger is 5.28.

Table of Contents

| | |
|---|---|
| Abstract | 1 |
| 1 DS-5 Debugger | 2 |
| 1.1 Launch Configuration | 2 |
| 2 TESSY Environment Settings..... | 3 |
| 3 Interactive Debugging..... | 4 |
| 3.1 Debug Configuration | 4 |
| 3.2 Install TESSY plugins..... | 5 |
| 3.3 Define Breakpoint at Test Object..... | 5 |
| 3.4 Activate the Plugin..... | 6 |
| 3.5 Stop Debugging..... | 7 |
| 3.6 Expressions View..... | 7 |
| 4 Troubleshooting..... | 8 |
| 4.1 Breakpoint cannot be reached | 8 |

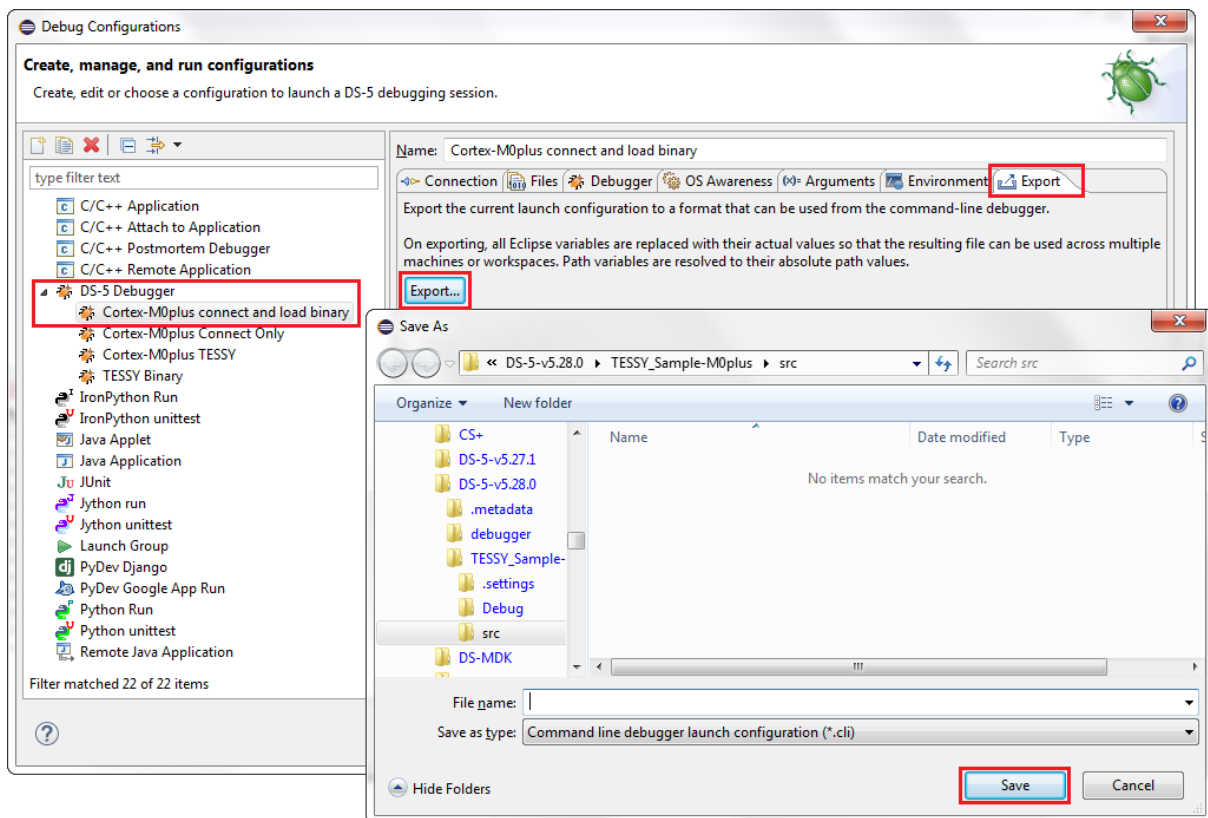
1 DS-5 Debugger

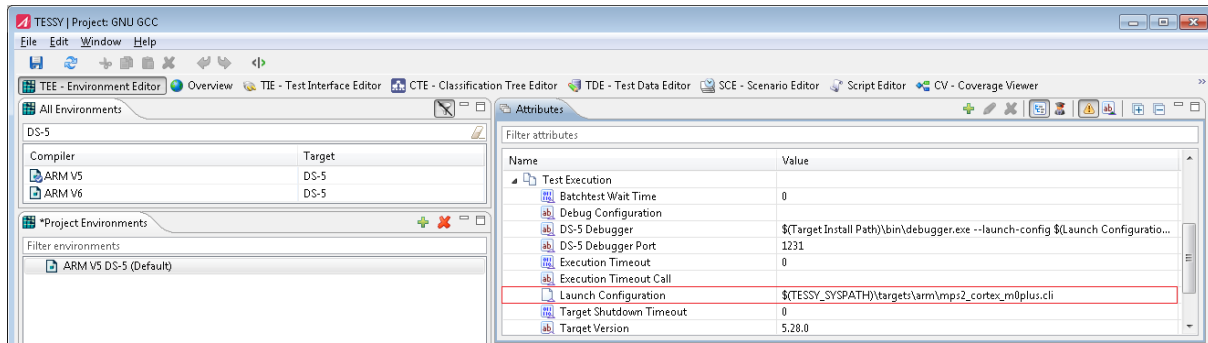
The communication between TESSY and the DS-5 debugger is based on TCP/IP protocol. DS-5 debugger is started by TESSY by executing the command line found in TEE attribute **DS-5 Debugger**. The command line contains the keyword *DS-5 Debugger Port* which will be replaced by TESSY's master by the contents of the TEE attribute **DS-5 Debugger Port**. *Since TESSY cannot determine the port number the DS-5 debugger is actually running on you must not set the port number to 0!* Furthermore, the DS-5 debugger is started with a special launch configuration file given by attribute **Launch Configuration** (See chapter 1.1).

In order to debug the test application interactively some preconditions have to be fulfilled. There are special plugins which have to be dropped into DS-5's installation directory (see chapter 3). These plugins will setup the proper launch configuration to connect to the by TESSY started running debugger instance.

1.1 Launch Configuration

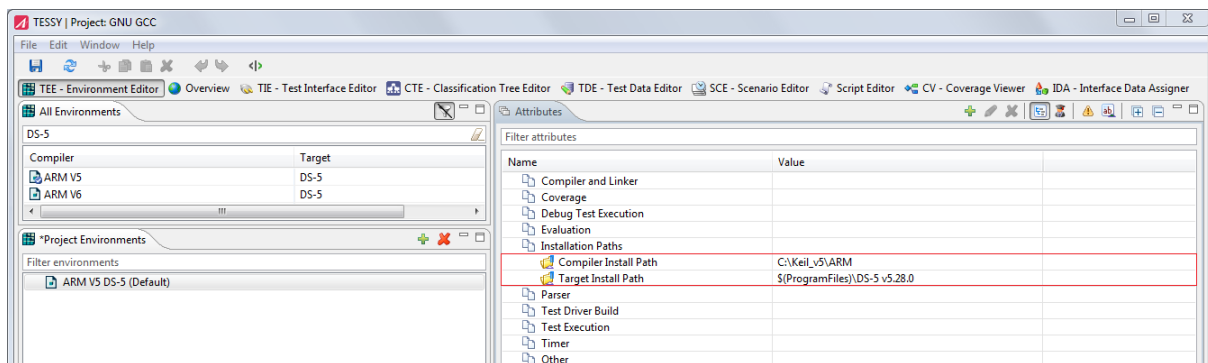
In order to run unit tests within the DS-5 debugger you have to provide a launch configuration file which can be generated by the **Debug Configuration** dialog of your *DS-5 Development Studio*. Select the debug configuration you are using for your own working DS-5 development project. Select the **Export** tab and click **Export...** within the file dialog chose a proper file name. Save the file into your TESSY project folder and set the full path of the file's location in TEE attribute **Launch Configuration**.



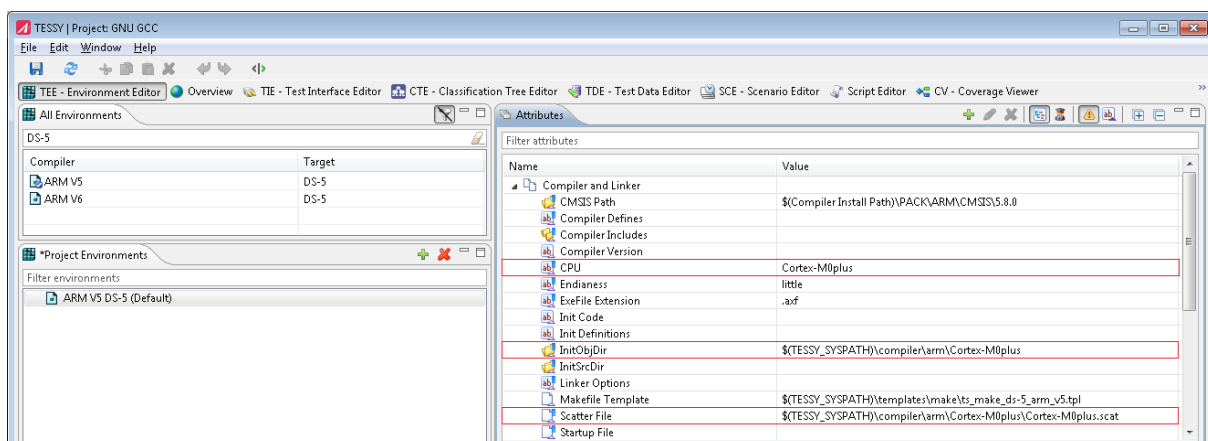


2 TESSY Environment Settings

There are several TEE attributes you have to pay attention to. At first set the path names to your compiler installation folder and to your target installation folder.



Set the proper **CPU** and do not forget to set the **Scatter File** and the directory path to your startup code within TEE attribute **InitObjDir**. If you do not know what startup code is about, peruse application note *063 Startup Code of the Test Program*.

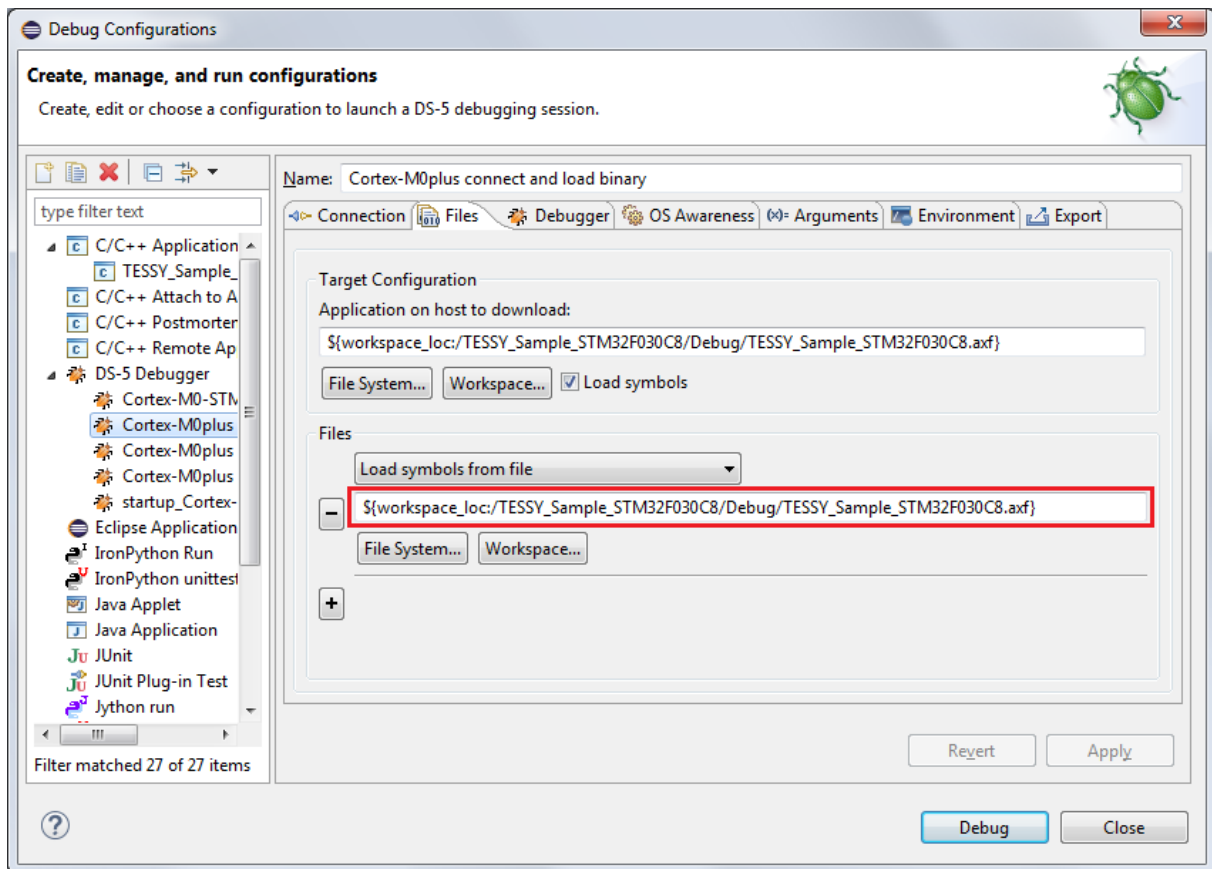


3 Interactive Debugging

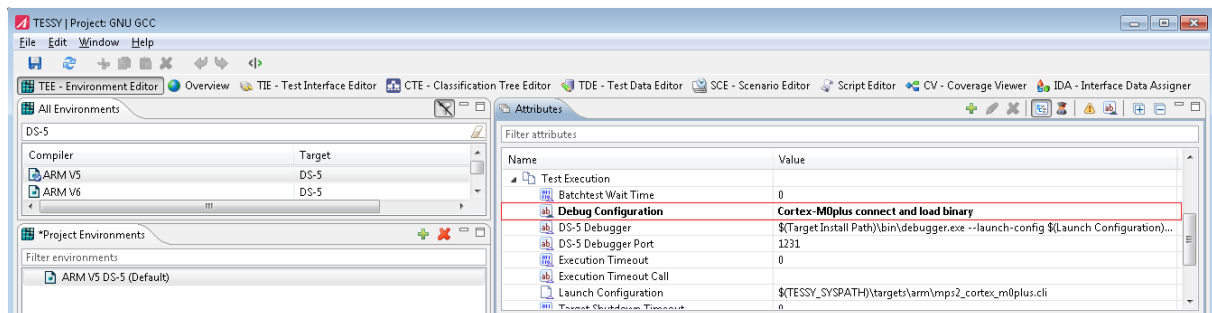
TESSY's DS-5 debugger adaption supports interactive debugging during a test run. However, some preconditions have to be fulfilled.

3.1 Debug Configuration

It is crucial to select a working debug configuration! The debug configuration also has to contain a symbol file. You may just copy the DS-5 project's application to be loaded as for instance shown below. The TESSY Plugin will create a temporary copy of this debug configuration and adjust and launch the temporary debug configuration. The temporary debug configuration will be removed when the TESSY Plugin is disabled.



The name of the debug configuration has to be set in TEE attribute **Debug Configuration**.



3.2 Install TESSY plugins

Copy all the plugins from TESSY's installation directory

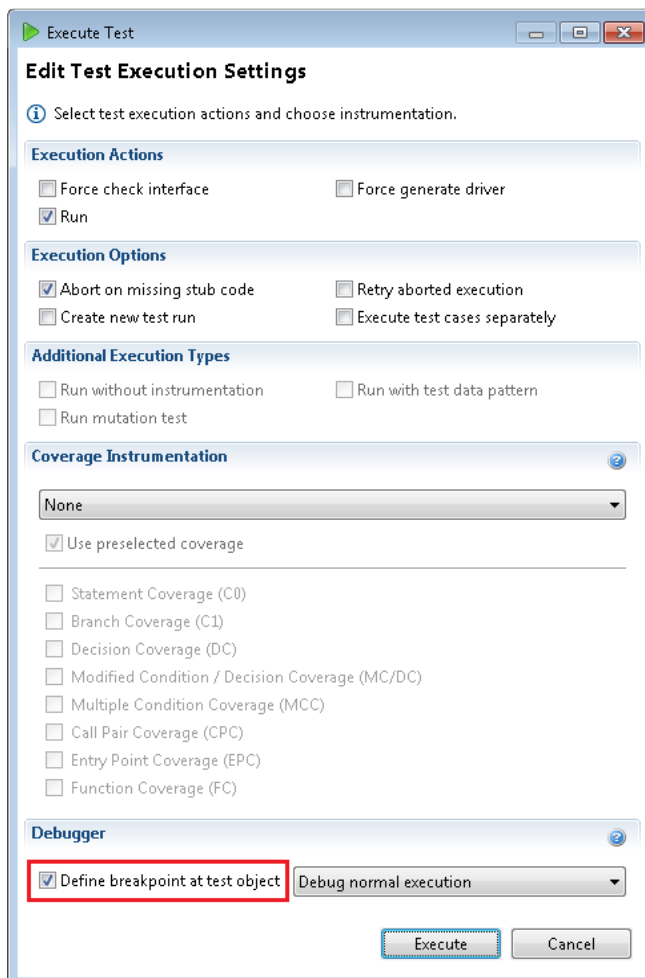
```
TESSY_SYS_PATH\targets\arm\DS-5 plugins
```

into DS-5's *plugins* folder

```
...\DS-5 v5.29.1\sw\eclipse\dropins\plugins
```

3.3 Define Breakpoint at Test Object

Open the **Execute Test** dialog and select **Define Breakpoint at Test Object**.



Click **Execute** and wait until TESSY's **Console** view displays the following message.

TESSY Application Notes

```
+-----+
| Execute test |
+-----+

Launching debugger.....done.

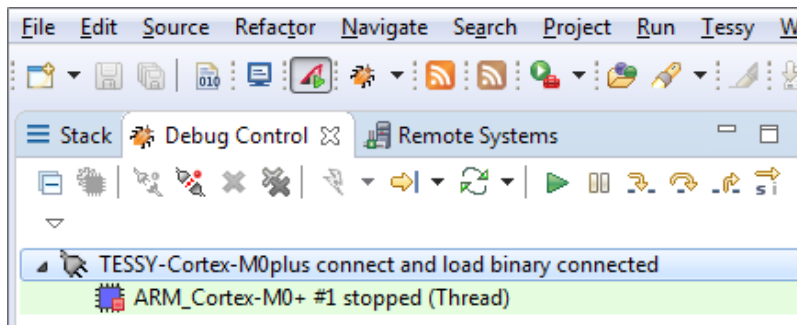
*****
* Please activate the TESSY plugin from within DS-5 now. *
*****
```

3.4 Activate the Plugin

Now press the corresponding button from DS-5's menu.



DS-5 will connect to the running debugger instance.



The editor will now display the current position of the test run which should be your test object as for instance shown below.

```
ts_src01.c
27 // However, the implementation is intentionally
28 // erroneous: v1 == 7 results "yes" instead of "no"!
29 //
30
31 result is_value_in_range (struct range r1, value v1)
32 {
33     if (v1 < r1.range_start)
34         return no;
35
36     if (v1 > (r1.range_start + r1.range_len))
37         return no;
38
39     return yes;
40 }
41 }
```

Stack window showing function prototypes and source lines:

| Function Prototype | Source/Line |
|---------------------------|------------------------------|
| is_value_in_range(r1, v1) | ts_src01.c:33 |
| TESSY_TestobjectCall() | ts_src01.c:193 |
| tessy_execute_task() | ts_is_value_in_range_s.c:167 |
| main() | ts_is_value_in_range_s.c:215 |

Variable watch window:

| Name | Value | Type | Count | Size | Location | Access |
|-------------|-------|--------------|-------|------|------------|--------|
| r1 | | struct range | | 64 | 0x2000055C | R/W |
| range_start | 10 | int | | 32 | 0x2000055C | R/W |
| range_len | 20 | int | | 32 | 0x20000560 | R/W |
| v1 | 15 | value | | 32 | \$R2 | R/W |

Code editor showing the implementation of is_value_in_range:

```

30
31 result is_value_in_range (struct range r1, value v1)
32 {
33     if (v1 < r1.range_start)
34         return no;
35
36     if (v1 > (r1.range_start + r1.range_len))
37         return no;
38
39     return yes;
40 }
41
  
```

3.5 Stop Debugging

In order to stop the debugging session press the TESSY Plugin button again.

The screenshot shows the IDE's toolbar with the 'Enable TESSY Plugin' button (represented by a red and white icon) highlighted with a red box. Below the toolbar, the status bar indicates 'TESSY-Cortex-M0plus connect and load binary connected' and 'ARM_Cortex-M0+ #1 stopped (Thread)'.

3.6 Expressions View

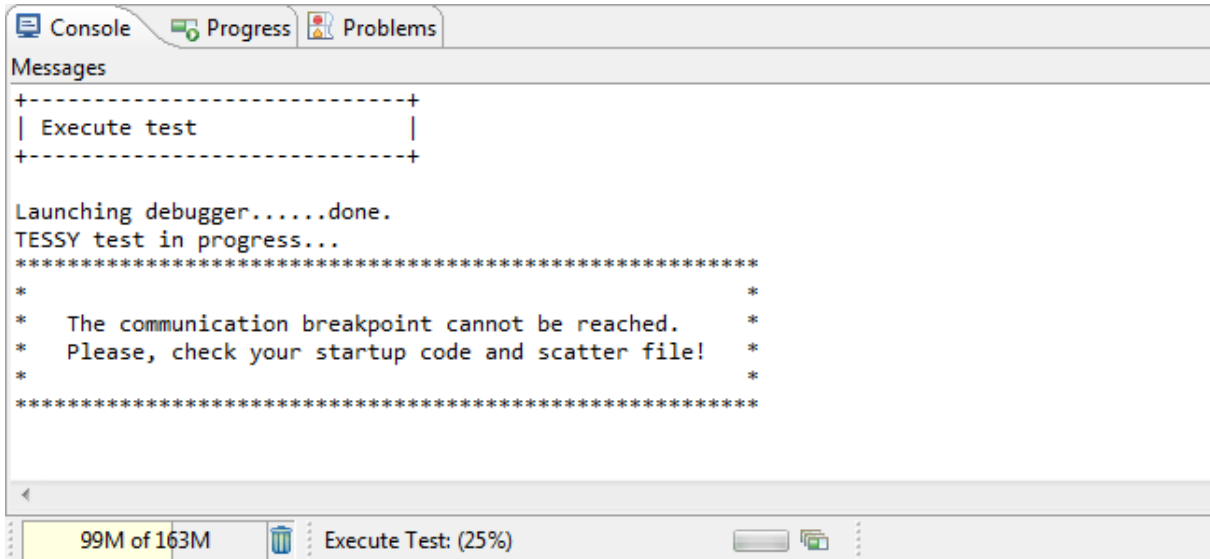
By default, DS-5's **Expressions** view contains the current test case and test step number as shown below. So for the sake of lucidity you might find it helpful to display it.

Expressions view window showing the following data:

| Name | Value | Type | Count | Size | Location | Access |
|---------------------------|-------|--------------|-------|------|------------|--------|
| TS_CURRENT_TESTCASE | 1 | TESSY_uint32 | | 32 | 0x20000004 | R/W |
| TS_CURRENT_TESTSTEP | 1 | TESSY_uint32 | | 32 | 0x20000008 | R/W |
| Enter new expression here | | | | | | |

4 Troubleshooting

4.1 Breakpoint cannot be reached



```
Console Progress Problems
Messages
+-----+
| Execute test |
+-----+

Launching debugger.....done.
TESSY test in progress...
*****
*
* The communication breakpoint cannot be reached.
* Please, check your startup code and scatter file!
*
*****

99M of 163M Execute Test: (25%)
```

As the message suggests make sure you added the proper startup code and scatter file. Please refer to application note *063 Startup Code of the Test Program* which will give you an idea what startup code is about and where to find it.