

# SEGGER JLink Debugger

## Abstract

This document describes the usage of the SEGGER JLink debugger as target system. The minimum required version of the SEGGER GDB server is 4.94. The SEGGER GDB server installation contains no GDB client. Thus, for none-GNU compilers you will need a third party GDB client. For instance, for the Keil Arm Compiler Version 6 a Linaro GDB client is required.

**Please note:** *The SEGGER GDB debugger does not support interactive debugging features when executing tests with TESSY. (See 2 to learn how to debug interactively having your test data statically built into the target binary.)*

## Table of Contents

|  |   |
|--|---|
| Abstract .....                           | 1 |
| 1 SEGGER JLink Debugger .....            | 2 |
| 1.1 TESSY Environment Settings .....     | 2 |
| 2 Interactive Debugging.....             | 3 |
| 3 Known Issues .....                     | 5 |
| 3.1 Generating Target Binary Fails ..... | 5 |

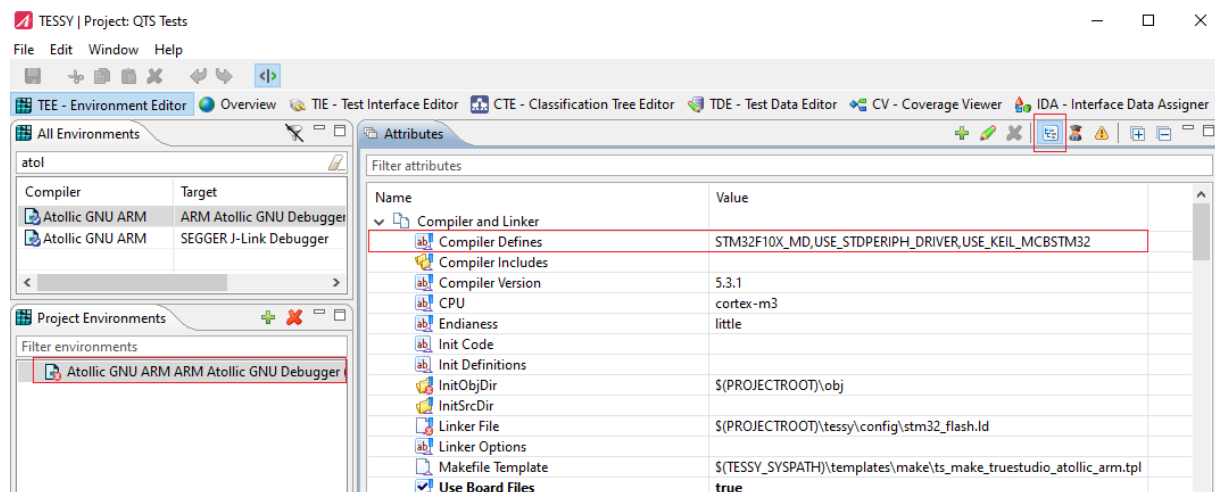
# 1 SEGGER JLink Debugger

The SEGGER JLink debugger uses a special GDB server as backend. This backend will be controlled by TESSY using a GDB client from Atollic or GNU Tools for ARM or Linaro when executing tests. The Linaro GDB client is required if the compiler is an Arm Compiler Version 6. The launch of the GDB server is controlled by the TEE attribute **Slave Call**. Please, adjust this attribute to match your specific GDB server command line settings. The test execution runs fully automated in this case.

In order to debug the test application interactively with the test case values provided from within TDE, you need to rebuild the test application in a special mode, i.e. the input values will be compiled into the application. You may then download the test application using the respective GDB client debugger and step through the test cases.

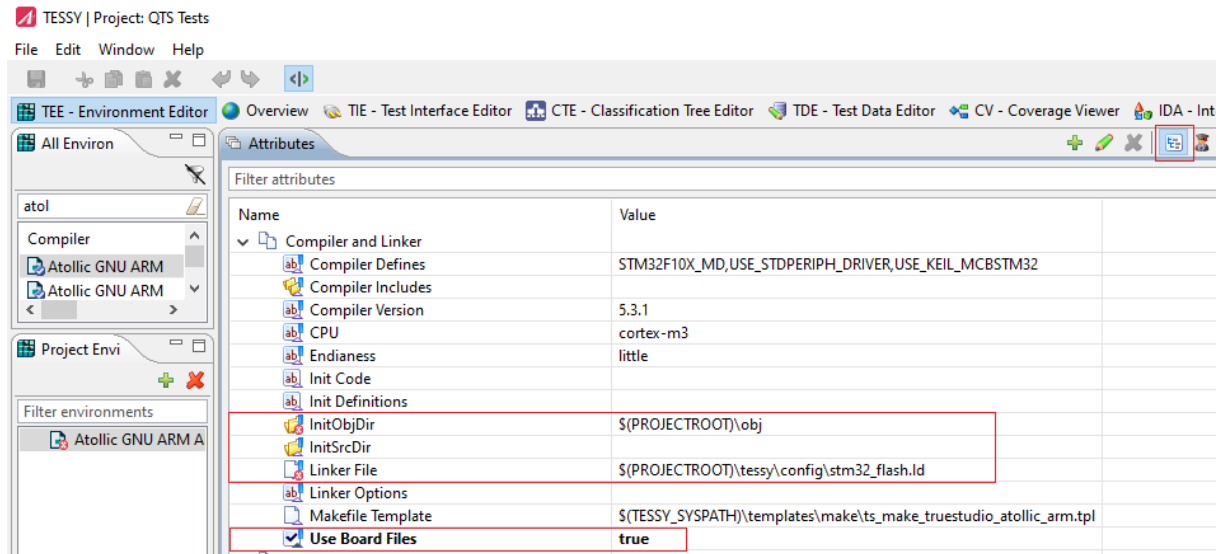
## 1.1 TESSY Environment Settings

There might be specific board defines which you can set for Atollic in TESSY's TEE as shown below. The same goes for GNU Tools for ARM correspondingly.



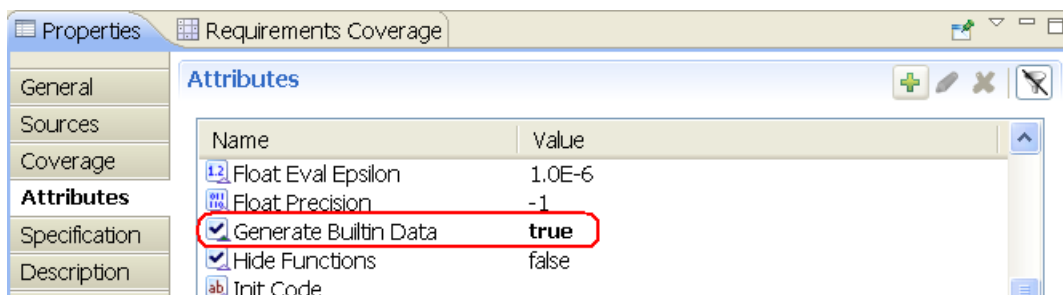
In order to link special start up files, also called board files, with each test object automatically you may use the attributes **InitSrcDir**, **InitObjDir**, and **Use Board Files**. Make sure the folders the attributes **InitSrcDir** and **InitObjDir** point to exist. Copy all needed source files and header files into the directory pointed to by the TEE attribute **InitSrcDir**.

It is also possible to provide the precompiled startup code only. If so, **InitObjDir** has to point to the directory containing the object files while the value of attribute **InitSrcDir** must be empty. Either way, do not forget to add the correct linker file path.

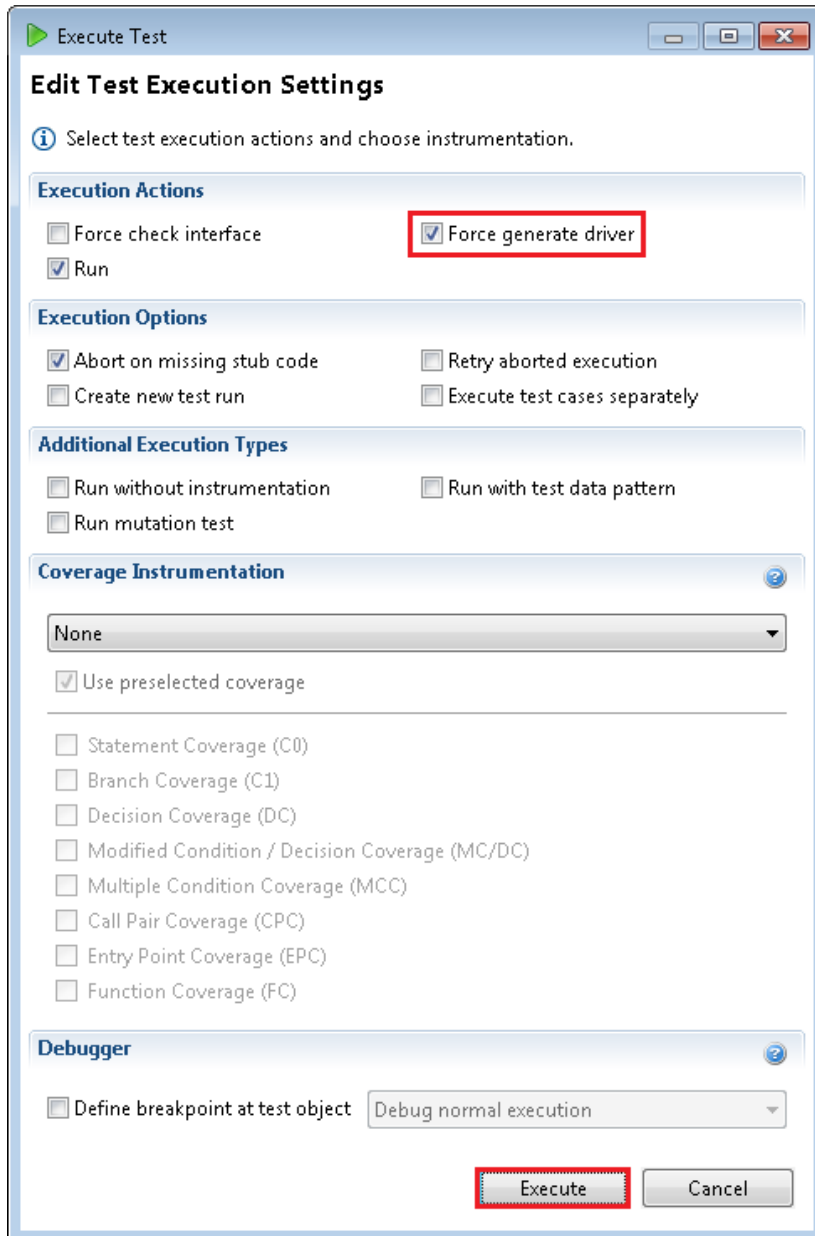


## 2 Interactive Debugging

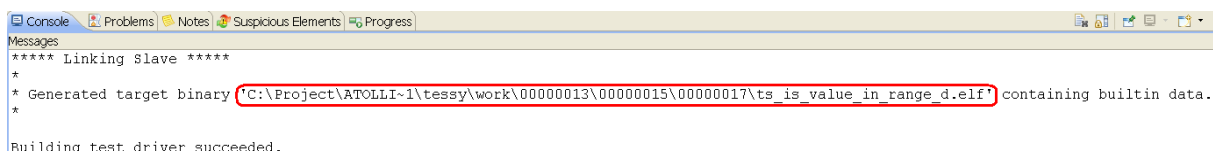
The TESSY SEGGER JLink debugger adaption does not support interactive debugging during a test run. But it is possible to debug your test object interactively having the test data built-in which might be useful in case of errors during a test run. In order to debug the test object interactively, TESSY provides the **Generate Builtin Data** attribute. The attribute is of type Boolean and, if set to **true**, TESSY will rebuilt your target binary during the next test run having the selected test data built-in, i.e. TESSY will not actually perform the test run but instead create the target binary with test data built-in. To disable this feature, you have to set the attribute to **false**.



Open the **Execute Test** dialog and make sure **Force Generate Driver** is selected.



Now execute the test by pressing the **Execute** button. TESSY displays the path to the generated built-in target binary in the **Console** view.

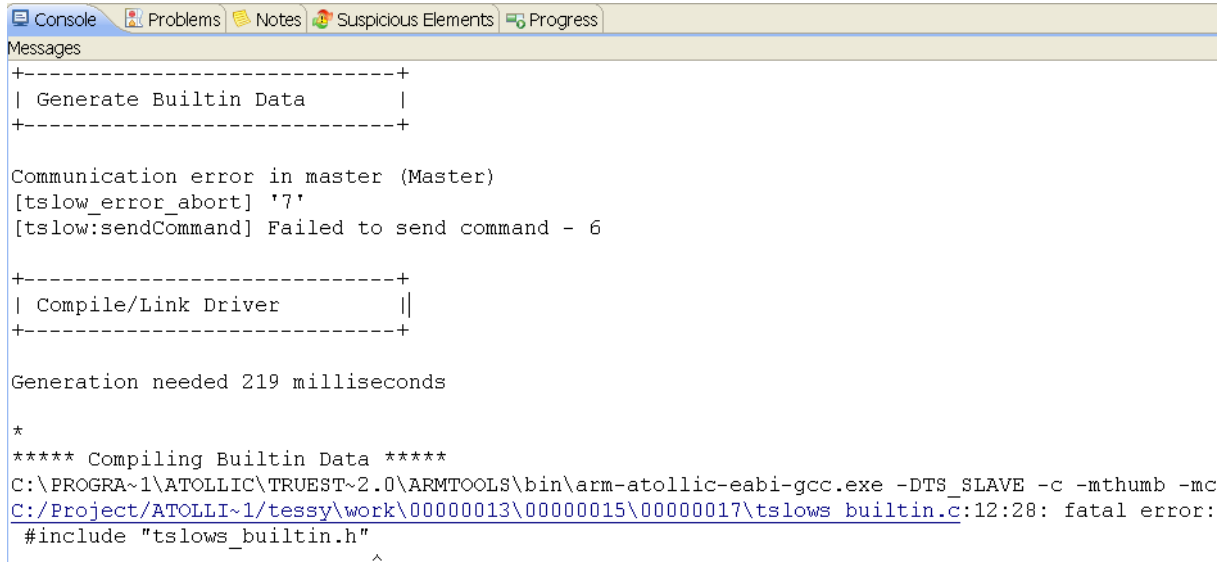


You can copy the path to the generated target binary from TESSY's **Console** view and use the corresponding GDB client to debug the application. Please, consult the SEGGER JLink respectively the GNU GDB documentation found on the internet on how to debug the application.

## 3 Known Issues

### 3.1 Generating Target Binary Fails

If the following error occurs during the generation of the target binary, please disable the **Comm Checksum** feature by setting the corresponding TEE attribute to **false**.



```
Console Problems Notes Suspicious Elements Progress
Messages
+-----+
| Generate Builtin Data |
+-----+

Communication error in master (Master)
[tslow_error_abort] '7'
[tslow:sendCommand] Failed to send command - 6

+-----+
| Compile/Link Driver ||
+-----+

Generation needed 219 milliseconds

*
***** Compiling Builtin Data *****
C:\PROGRA~1\ATOLLIC\TRUEST~2.0\ARMTOOLS\bin\arm-atollic-eabi-gcc.exe -DTS_SLAVE -c -mthumb -mc
C:/Project/ATOLLI~1/tessy/work\00000013\00000015\00000017\tslows_builtin.c:12:28: fatal error:
#include "tslows_builtin.h"
^
```