

Using Imperas ISS

Abstract

This document describes the usage of the Imperas ISS as target system. The minimum tested version of the Imperas ISS is 20210408.0.

Important Note: You need a **functional** development project which can successfully build a target binary and launch a debug session. All required Imperas environment variables have to be set up for your user or on your system properly.

Please note: The Imperas ISS adaption does not support interactive debugging features when executing tests with TESSY. (See 4 to learn how to debug interactively having your test data statically built into the target binary.)

Table of Contents

Abstract	1
1 Introduction.....	2
2 TESSY Environment Settings.....	2
3 Parallel Built and Parallel Execution.....	3
4 Interactive Debugging.....	3
5 Troubleshooting.....	4

1 Introduction

The communication between TESSY and the Imperas ISS is based on GDB client and iss.exe as backend. The Imperas ISS is started by TESSY by executing the command line found in TEE attribute GDB Server Debugger. By default, the port number is chosen by the Imperas ISS itself, i.e. **GDB Server Port** is set to 0. *It is strongly recommended to keep this value at 0.* It is required to be 0 to let the iss.exe run in parallel. The iss.exe outputs its server port on command line, which TESSY's master reads and uses to connect. TEE attribute **Port Trigger** holds the unique string, which precedes the port number directly.

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 (see chapter 3).

2 TESSY Environment Settings

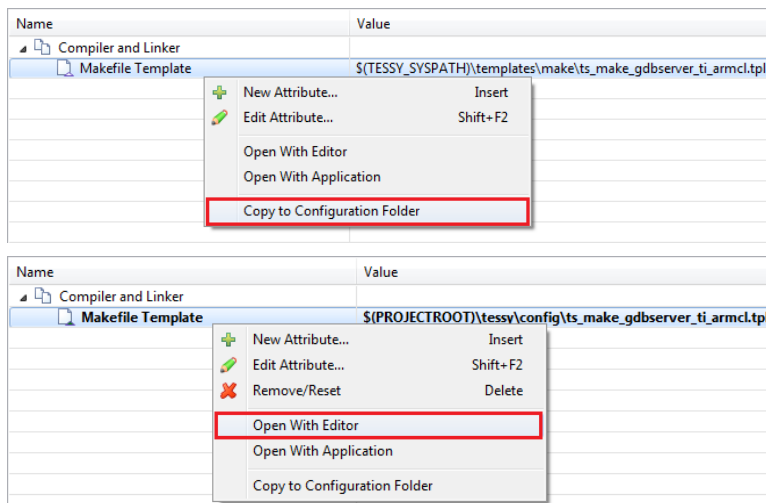
There are several TEE attributes you have to pay attention to. At first, check the path to the compiler and to the Imperas installation directory.

Installation Paths	
Compiler Install Path	C:\ti\ccs1000\ccs\tools\compiler\ti-cgt-arm_20.2.4.LTS
Target Install Path	\$(IMPERAS_HOME)

Check the GDB client debugger path and the parameters used to start the debugger.

Test Execution	
Debugger Path	\$(Target Install Path)\bin\Windows64\iss.exe
GDB Client Debugger	C:\ti\ccs1000\ccs\tools\compiler\gcc-arm-none-eabi-9-2019-q4-major\bin\arm-none-eabi-gdb.exe
GDB Server Debugger	"\$(Debugger Path)" --port \$(GDB Server Port) --processorvendor \$(Processor Vendor) --processorna...
GDB Server Port	0
Override Parameter	iss/cpu0/override_SCTLR_IE=T
Processor Name	arm
Processor Variant	Cortex-R4F
Processor Vendor	arm.ovpworld.org

Finally, check the **Compiler and Linker** attributes within the TEE. For more advanced compiler and linker settings copy the Makefile template into your TESSY project's config folder, open the file, and adjust the variables **S_COMP_OPTIONS** and **S_LINK_OPTIONS** accordingly to your local settings.

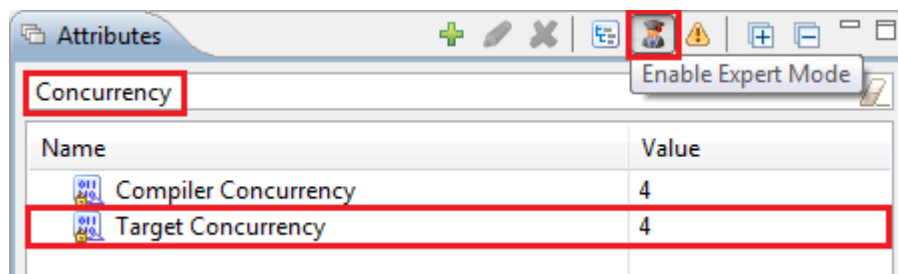


Name	Value
Compiler and Linker	
Makefile Template	\$(TESSY_SYSPATH)\templates\make\ts_make_gdbserver_ti_armcl.tpl

Name	Value
Compiler and Linker	
Makefile Template	\$(PROJECTROOT)\tessy\config\ts_make_gdbserver_ti_armcl.tpl

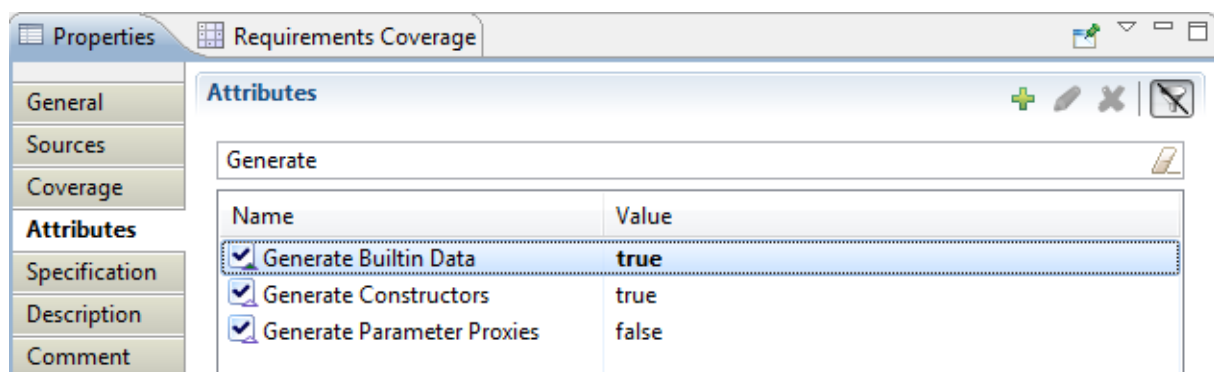
3 Parallel Built and Parallel Execution

By default, the parallel built is enabled, because the compiler usually provides parallel compiler and linker calls. You can disable the parallel built by setting TEE attribute **Compiler Concurrency** to **1**. However, the parallel execution is disabled by default, because the Imperas ISS requires a fresh license for each program execution of the same processor type. Nevertheless, did we test the parallel execution successfully. If you have access to more than one license for the Imperas ISS, than you can enable the parallel execution of your test run by entering the number of licenses you can use into the TEE attribute **Target Concurrency**. Select **Enable Expert Mode** of the menu of TESSY's TEE **Attributes** view, enter **Concurrency** into the search bar, and enter the number of licenses you can use for the test run.

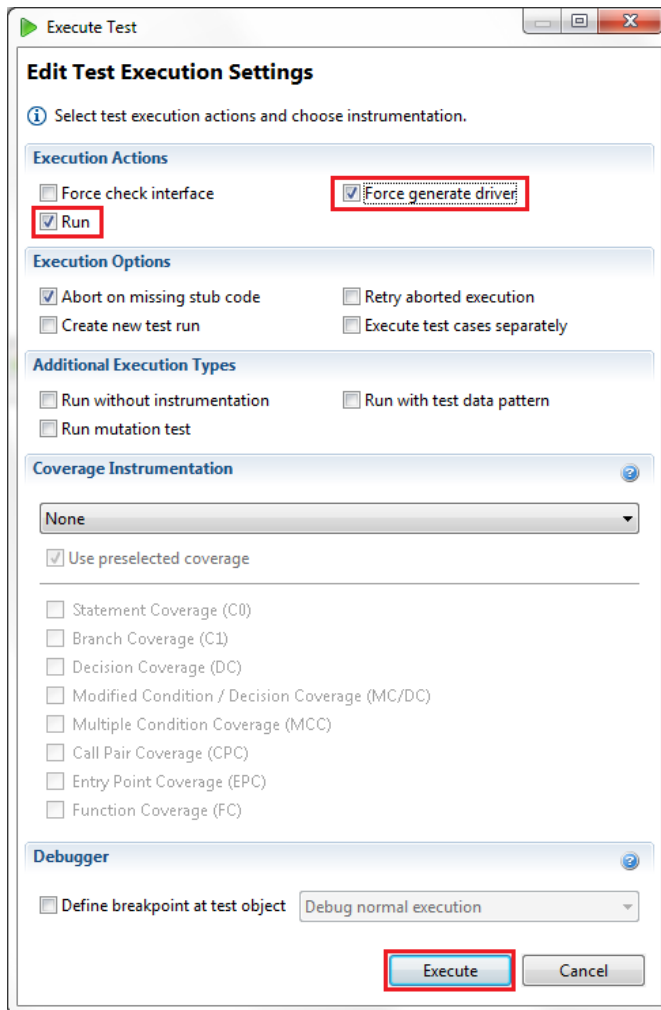


4 Interactive Debugging

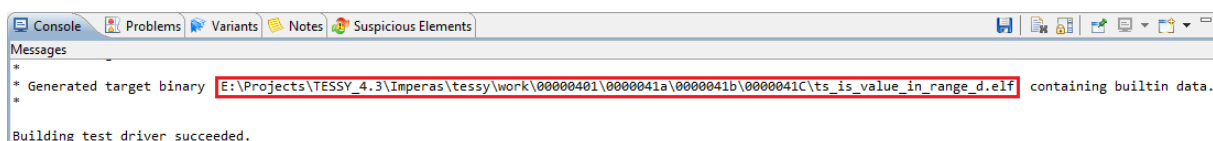
The TESSY Imperas ISS adaption does not support interactive debugging during a test run. But it is possible to debug your test object interactively having the test data built-in which might be useful in case of errors during a test run. So, in order to debug the test object interactively TESSY provides 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 it. 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 of the generated target binary from TESSY's **Console** view and use the corresponding GDB client to debug the application. Please, consult the GNU GDB documentation found on the internet on how to debug the application.

5 Troubleshooting

By now, there are no known problems. Please contact support@razorcat.com if you encounter any unsolvable problems.