

Keil μ Vision Debugger

Abstract

This document describes the usage of the Keil/ARM μ Vision Debugger as target system. What you need is a project file with the correct settings for your target hardware (or simulator). Such a μ Vision project file needs to be adapted for usage with TESSY as described in this document.

Please note:

- 1) *The communication between TESSY and μ Vision uses different communication interfaces depending on the μ Vision version. Please refer to chapter 1 of this document for details.*
- 2) *There is a new TEE attribute **CMSIS Version** which regards the version of the installed compiler. Please refer to chapter 5.1 of this document for details.*
- 3) *Since TESSY version 4.3.10 the `tthd` is used to start the μ Vision Debugger before the test run and it is used to close the debugger after all tests are executed. Refer to chapter 4 for further details.*
- 4) *For microcontrollers like EFM8BB3 from Silicon Labs it may be required to set TEE attribute **Send Single Bytes** to true. That is, writing data in blocks is not supported by the debugger API for this microcontroller. Enabling this attribute may result in slower running tests.*

Table of Contents

Abstract	1
1 Keil/ARM μ Vision Interfaces	3
2 Setup for μ Vision	3
2.1 μ Vision4 or higher	4
2.2 μ Vision Legacy Versions 3.xx	5
3 μ Vision Project File Preparation	6
3.1 μ Vision Target Settings	6
3.1.1 Step 1: Select μ Vision Project	6
3.1.2 Step 2: Create a Project Target.....	7
3.1.3 Step 3: Adjust the Name/Folder of the Executable.....	8
3.2 μ Vision Configuration Settings	10
3.3 Flashing the Executable File	12
4 Start the μ Vision Debugger for a test run manually	12
5 Example: Preparing TESSY for an ARM Microcontroller	12
5.1 Adjust TEE attribute CMSIS Path.....	12
5.2 Step 1: Select a μ Vision Project	13
5.3 Step 2: Create a new TESSY Project.....	14
5.4 Step 3: TEE Configuration.....	15
5.5 Step 4: Edit the Makefile Template.....	15
6 Troubleshooting.....	17
6.1 Error Executing the Test.....	17
6.2 HEAP Space Setting for C++	18
6.3 Warning after Starting a Test	19
6.4 Removing Breakpoints	19
6.5 The debugger IDE window vanishes	20

1 Keil/ARM μ Vision Interfaces

The communication between TESSY and μ Vision4 or higher uses the UVSC interface of μ Vision. This interface allows remote controlling of μ Vision for test execution.

Former legacy versions of μ Vision (version 3 and 2) used two different interfaces depending on the specific version. The table below shows the applicable versions and the kind of interface used. Both interfaces are part of the respective Keil/ARM μ Vision installation.

Controller	μ Vision Version	Description
C166/C51	3.62 and higher	UVSC interface
ARM	3.80 and higher	UVSC interface
ARM/C166/XC166/C51	4	UVSC interface
ARM/C166/XC166/C51	5	UVSC interface

μ Vision 3.80 and later provide stable versions of the UVSC DLL. Prior versions (3.51 to 3.61 for all microcontrollers and 3.51 to 3.80 for ARM controller) are no longer supported. Please, utilize TESSY 4.3 in case you need one of these obsolete versions.

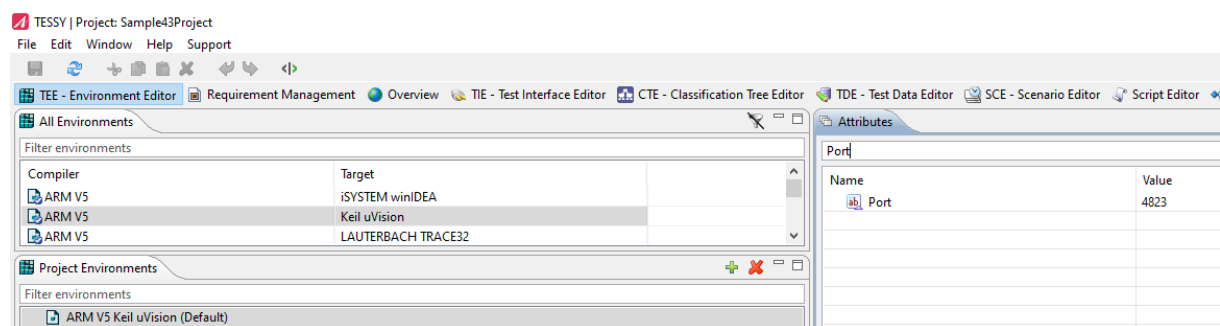
Important note: The UVSC interface requires the communication port number within μ Vision, i.e. UVSOCK (TCP/IP) Settings, to be enabled as described in chapter 2.1.

2 Setup for μ Vision

The UVSC interface requires setting a port number which is available at different locations within the μ Vision GUI (depending on the version). You may choose a different port number than the default one, but this requires also changing the **Port** attribute within the TESSY Environment Editor (TEE) as shown below.

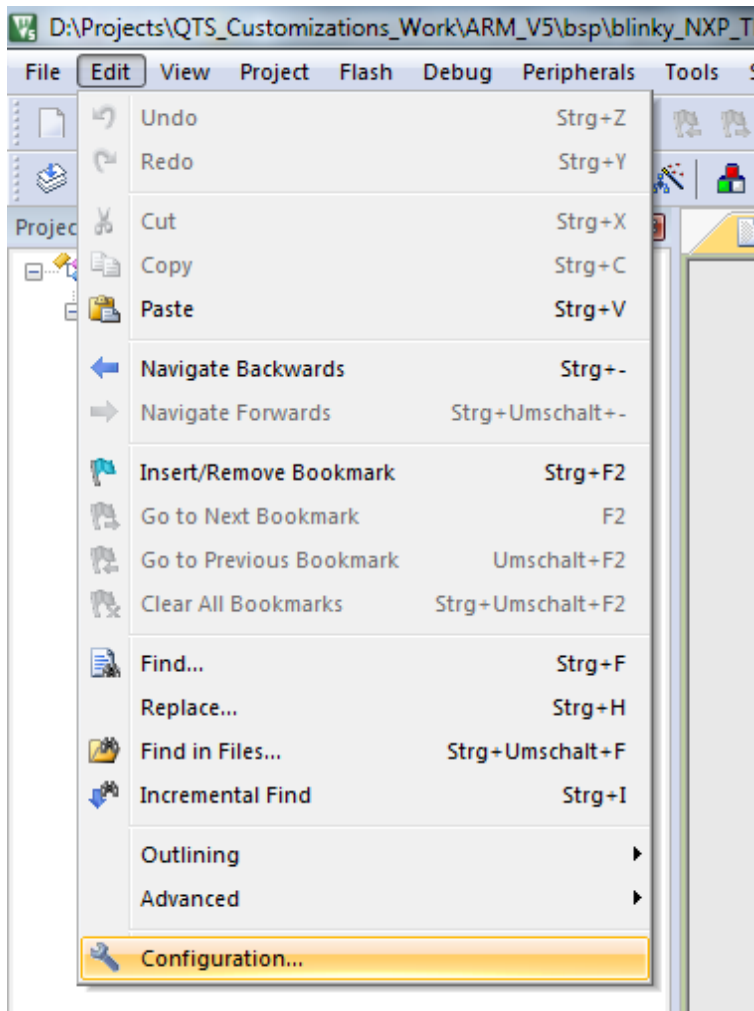
Please note:

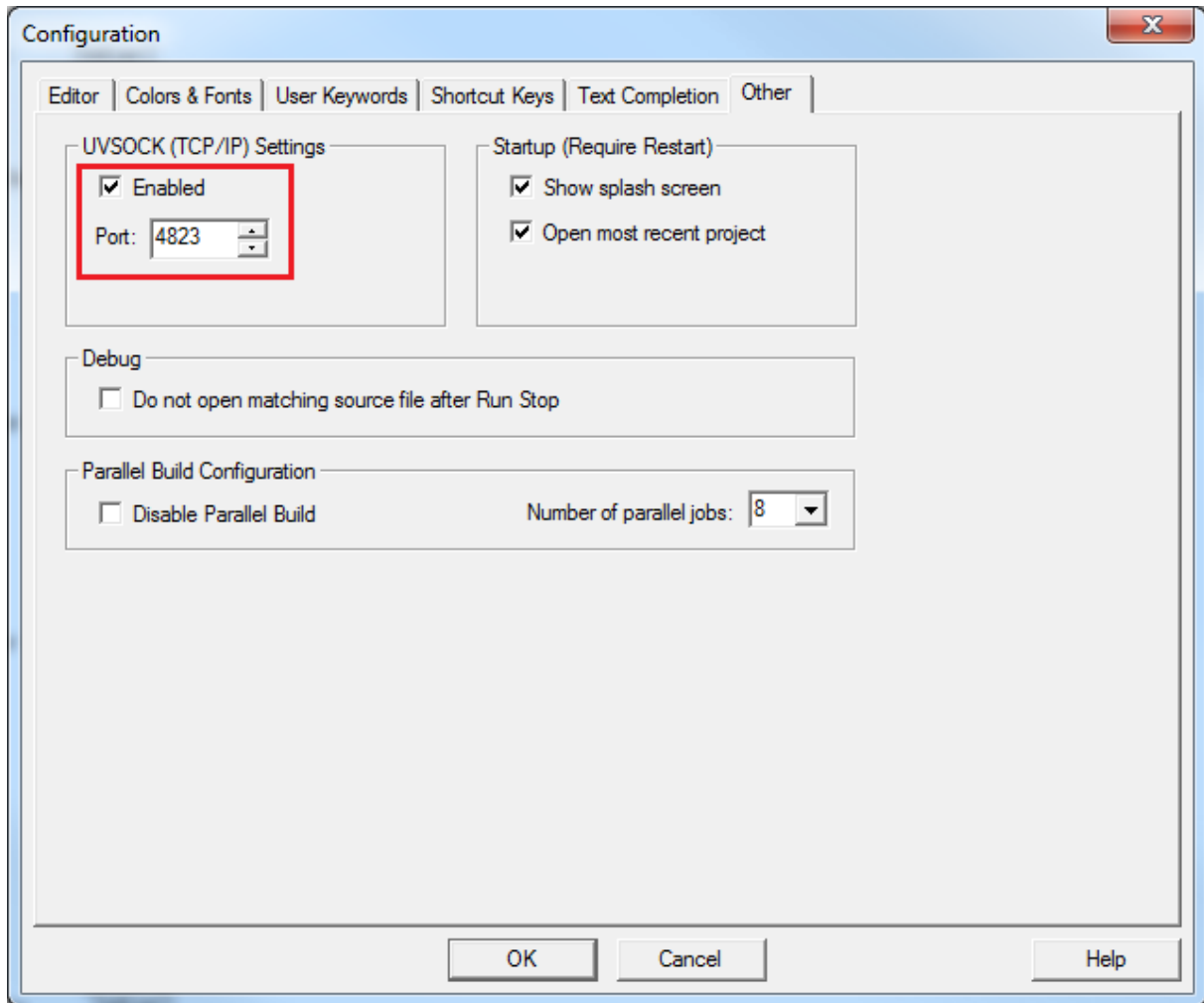
Please make sure that the port number selected for TEE attribute **Port** corresponds to the port number you have selected within μ Vision.



2.1 μ Vision4 or higher

Since version 4 μ Vision uses the UVSC interface which has to be activated by enabling the port number which is used to connect to the μ Vision UVSC interface. Choose **Configuration...** from the **Edit** menu to open the **Configuration** dialog and choose the **Other** tab.

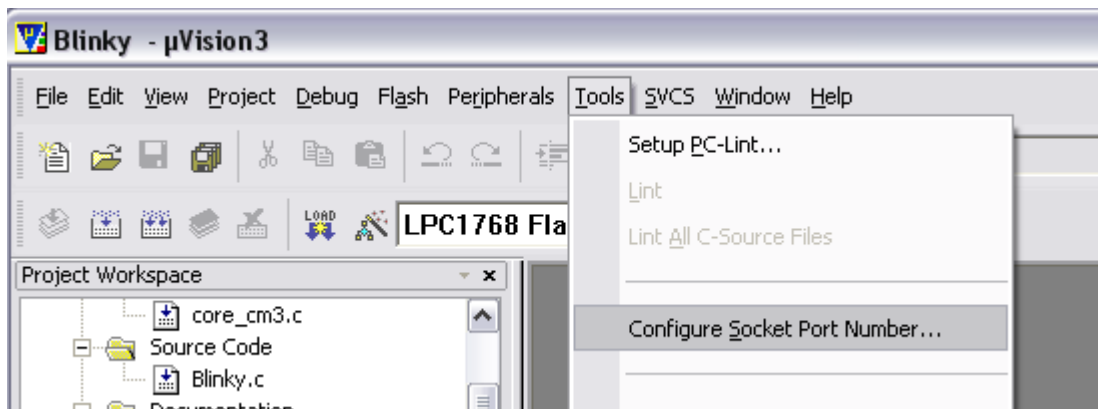




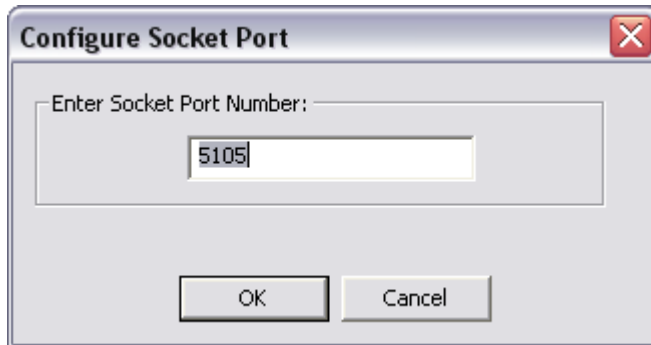
The default port number to be used with TESSY is **4823**. Please make sure that the **Enabled** toggle button is checked.

2.2 μ Vision Legacy Versions 3.xx

μ Vision version 3.62 and later started to use the UVSC interface, within which you need to setup the port number used to connect to the old μ Vision interface as described in chapter 2. Choose **Configure Socket Port Number** from the **Tools** menu as shown below:



This will launch the **Configure Socket Port** dialog as shown below.



The default port number to be used with TESSY is the **4823** whereas the default port number of μ Vision version 3 is the **5105**. You may either change the port number within μ Vision or adapt the TEE attribute **Port** as shown in chapter 2.

3 μ Vision Project File Preparation

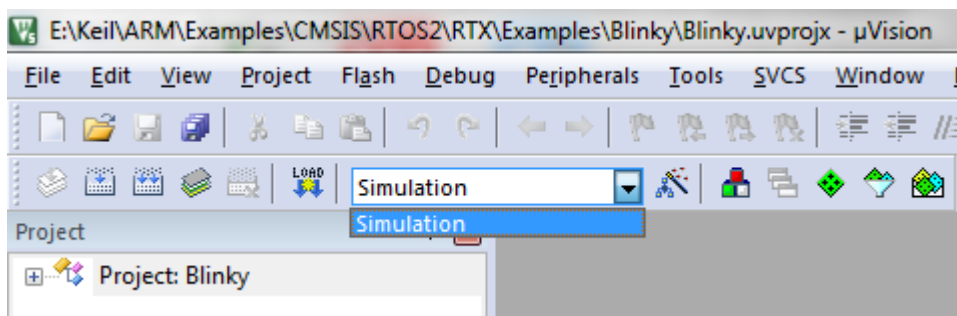
In order to run any test, you need to start μ Vision with a μ Vision project file that has been prepared as described below. We suggest using the project file that you are using for your development project. You may as well start with any other μ Vision project file (e.g. we use the “Blinky” from the examples projects provided by μ Vision’s Pack Installer).

3.1 μ Vision Target Settings

μ Vision manages so called **Targets** within a μ Vision project file in order to handle multiple configurations within the same project file. The targets are displayed within a combo box within the μ Vision toolbar. It is recommended to add a new target for use with TESSY. Please follow the steps below to create and configure such a target.

3.1.1 Step 1: Select μ Vision Project

Start μ Vision and open any of the “Blinky” projects copied from μ Vision’s Pack Installer. The open combo box as shown in the screenshot below contains the available μ Vision target configurations.

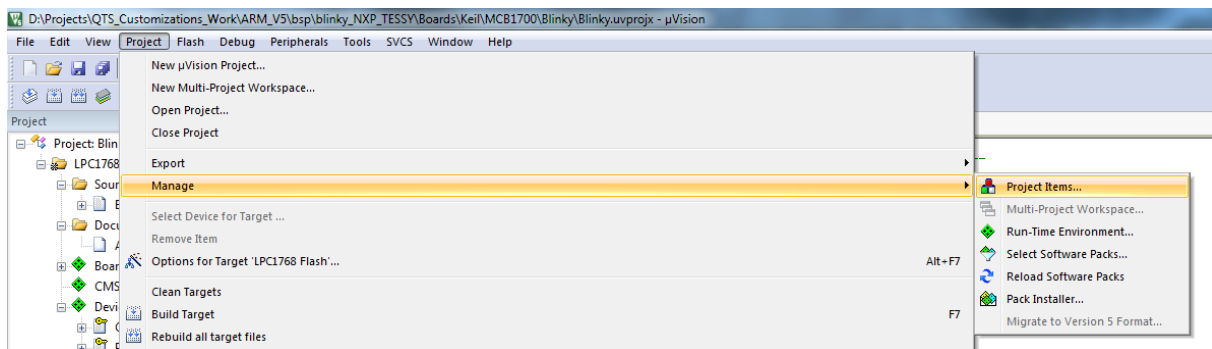


Select the target configuration that you would like to use for running tests with TESSY (e.g. “Simulator”).

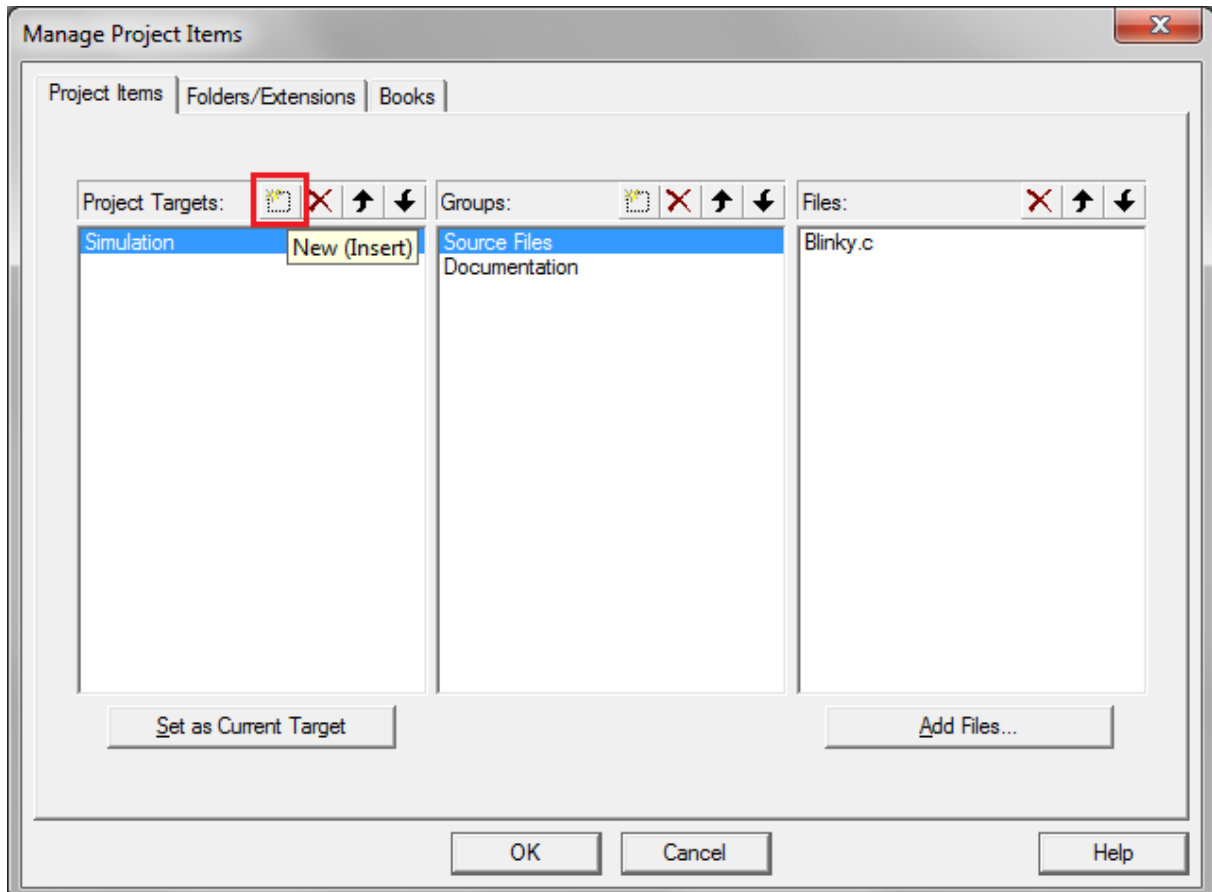
Please note: You may as well use any other of the available target configurations. But make sure that the selected configuration runs fine with the simulator target or your connected hardware.

3.1.2 Step 2: Create a Project Target

Select the **Project Items...** entry of the **Manage** sub menu of the **Project** menu.

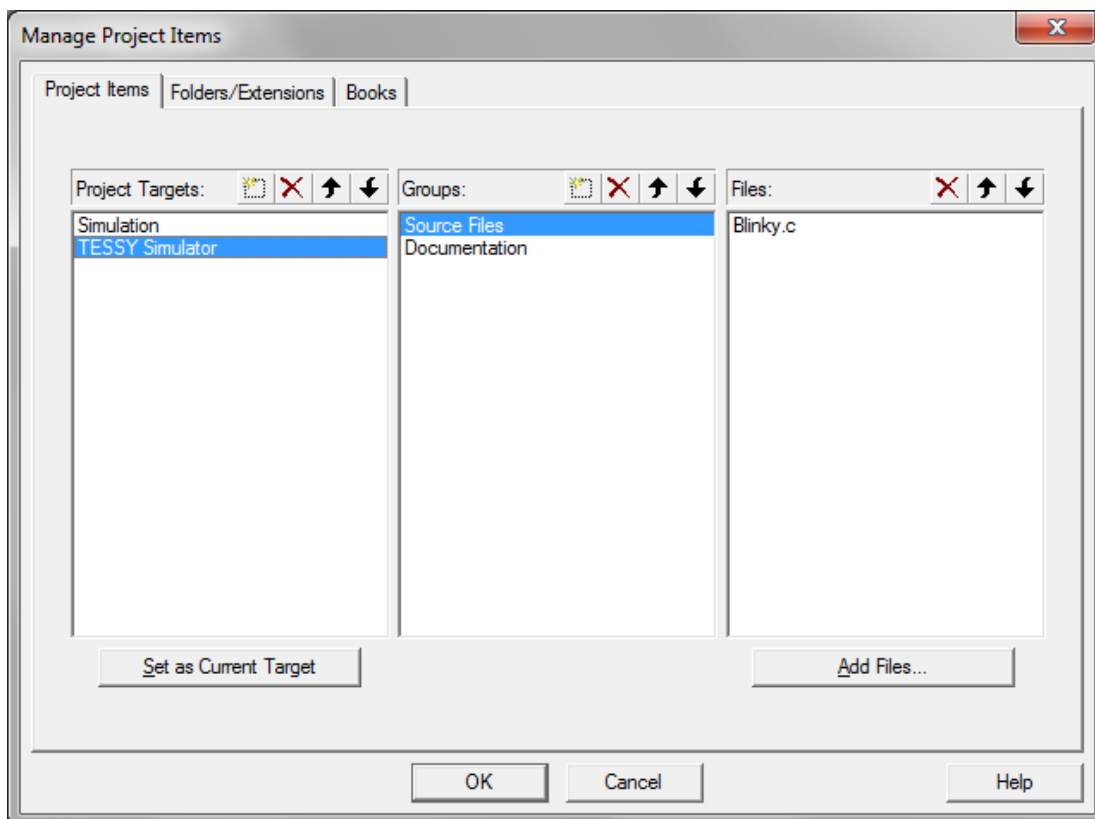


This will show the **Manage Project Items** dialog.



Within this dialog, you need to create a new entry derived from the currently selected entry. We will use the **Simulator** configuration as an example. Add a new entry to the **Project Targets** list by pressing the **New (Insert)** button. You may choose an arbitrary name for the new entry (e.g. “TESSY Simulator”).

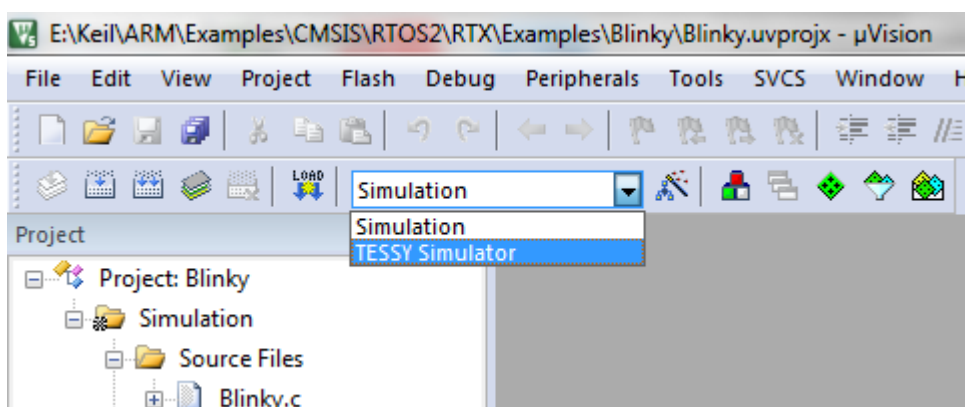
Please note: A new project target will be copied from the project target that is currently selected within the μ Vision main window (refer to section 3.1.1)



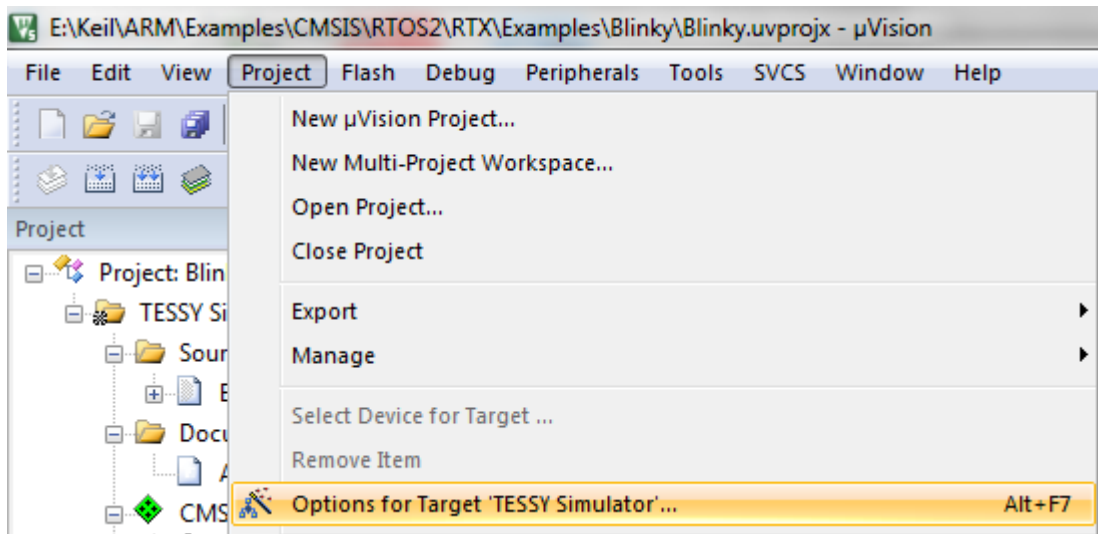
Press **OK** to save your new configuration.

3.1.3 Step 3: Adjust the Name/Folder of the Executable

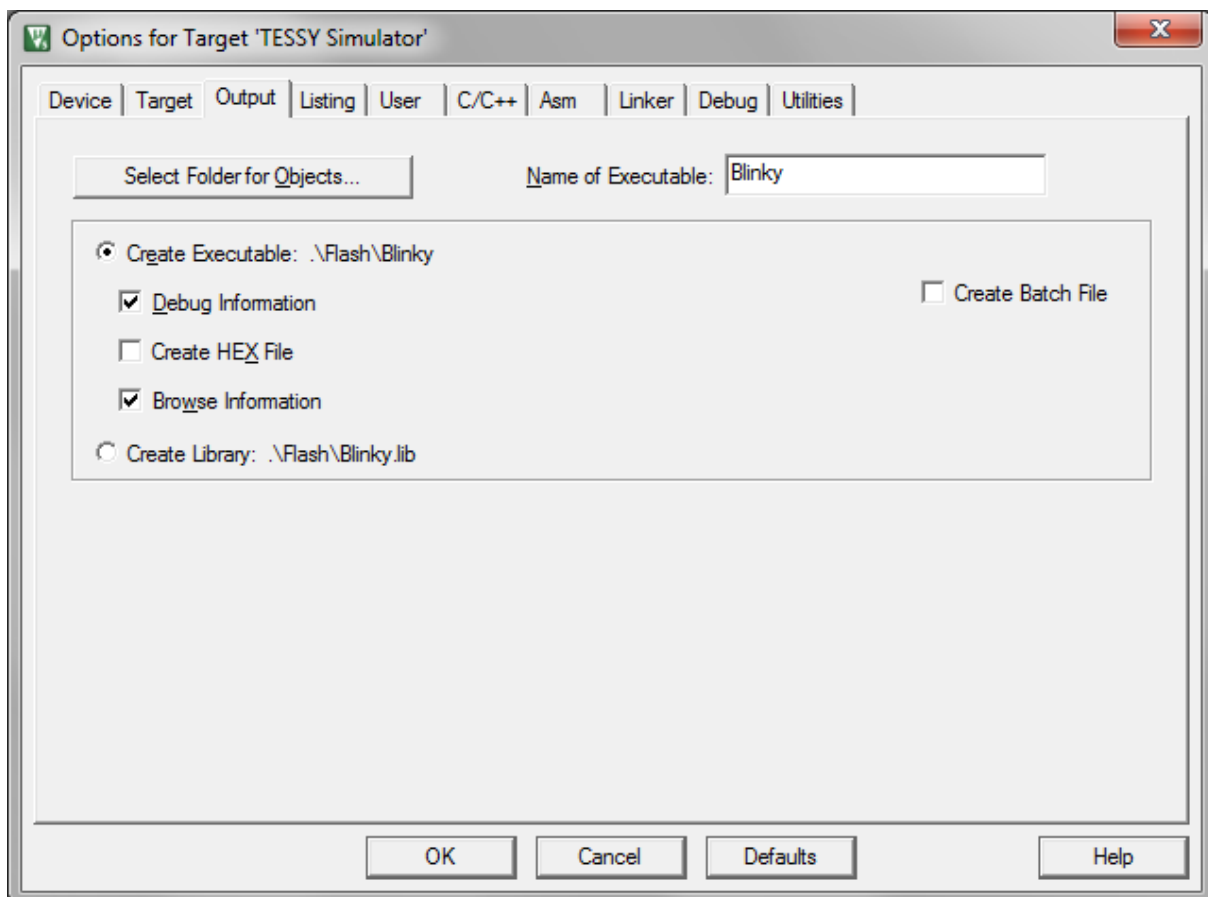
The newly created project target needs to be configured for TESSY. Please select the new project target “TESSY Simulator” within the project targets combo box of μ Vision as shown below.



Now choose **Options for Target 'TESSY Simulator'...** from the Project menu.

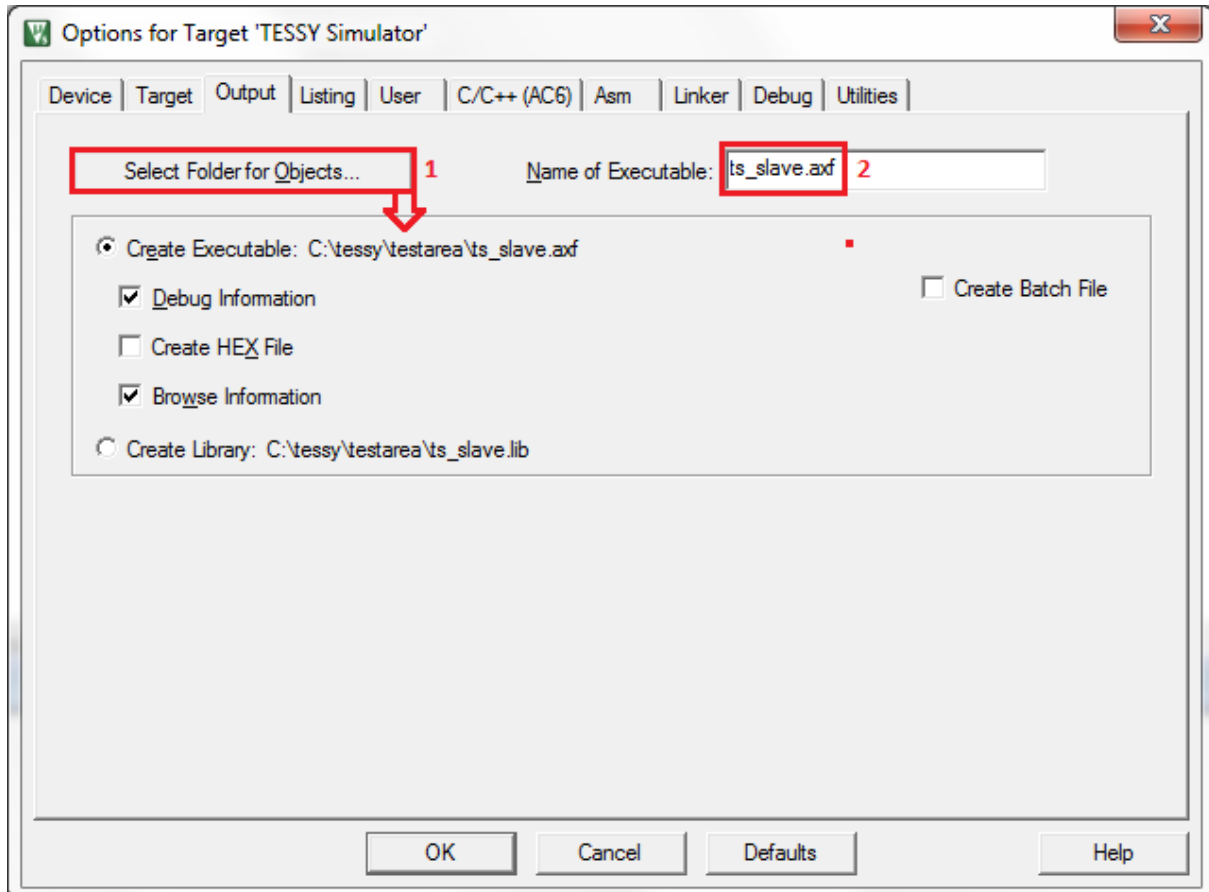


This will display the **Options for Target 'TESSY Simulator'** dialog. After switching to the **Output** tab, you will see the name and path of the “Blinky” executable.



The following changes are required within this dialog.

- (1) Press the **Select Folder for Objects...** button and choose the path to the TESSY testarea directory (normally `C:\tessy\testarea`).
- (2) Add the name `ts_slave.axf` into the **Name of Executable** text box.



The resulting name of the executable file should be

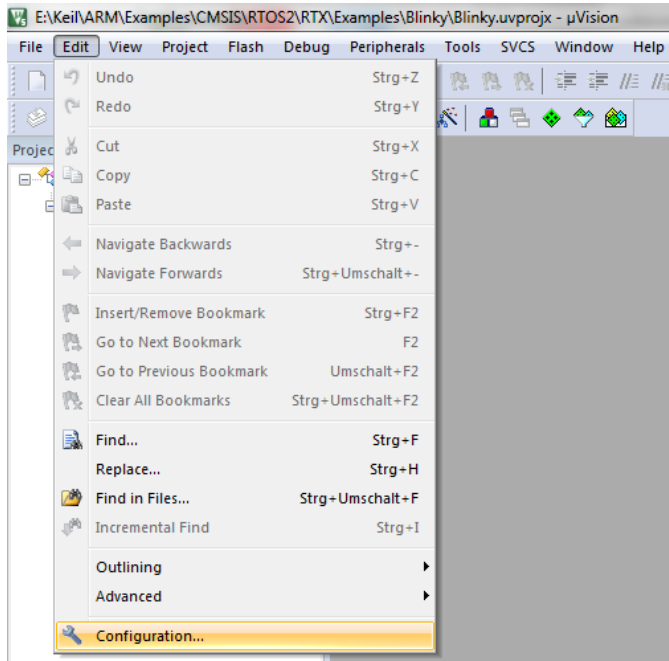
`C:\tessy\testarea\ts_slave.axf`

Now close the dialog by pressing the **OK** button. The new μ Vision project target is now prepared for your TESSY test runs.

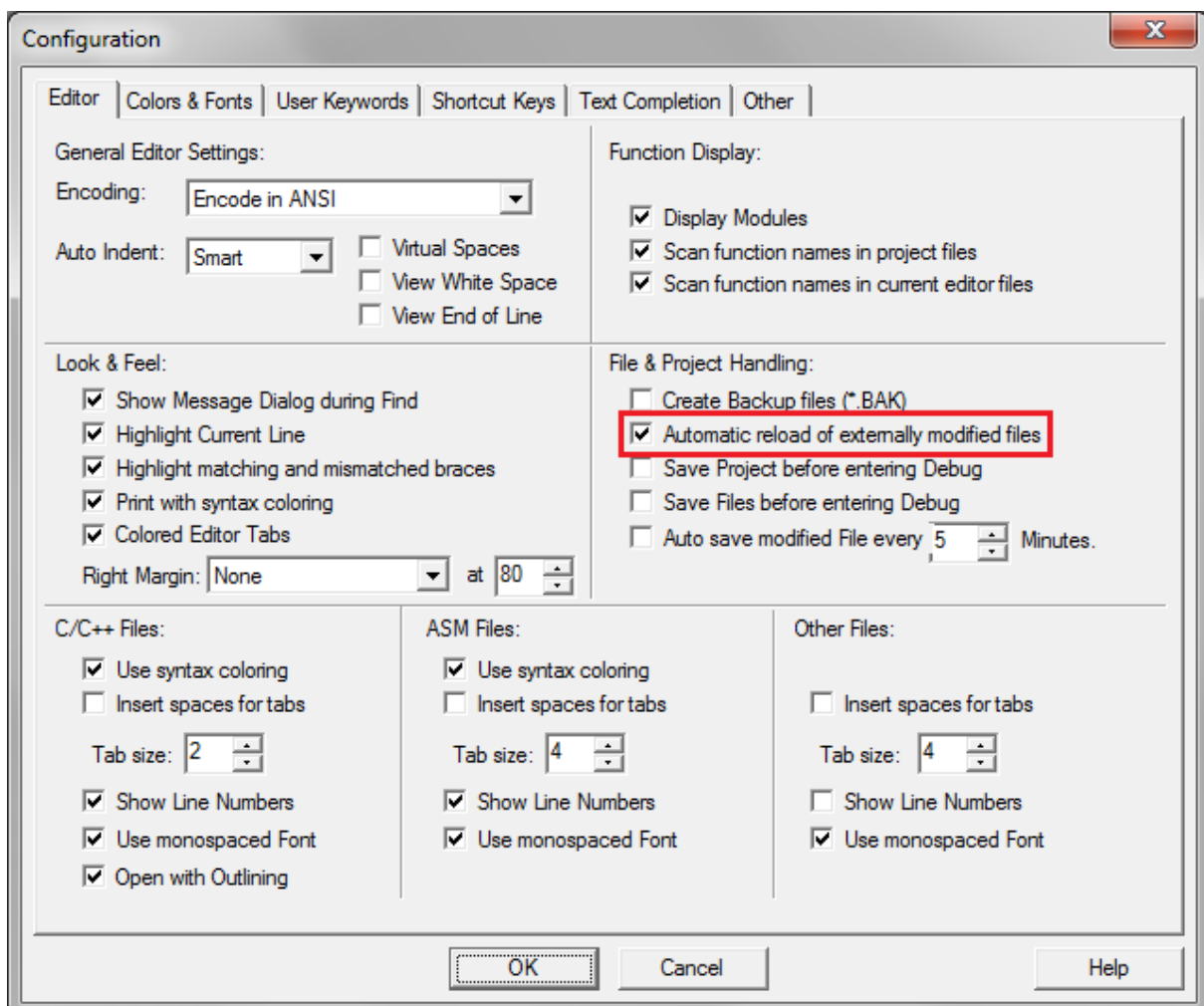
3.2 μ Vision Configuration Settings

To avoid warning dialogs during test execution, you should ensure the following configuration settings of μ Vision. Select the **Configuration ...** menu entry from the **Edit** menu.

TESSY Application Notes



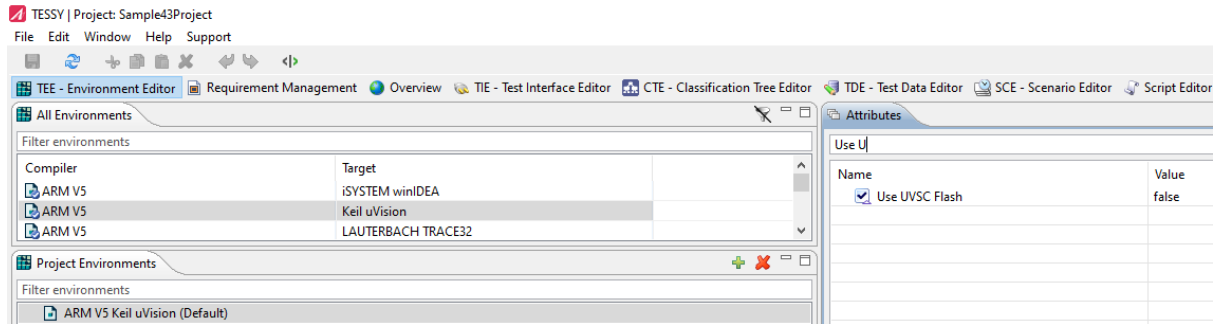
Within the Configuration dialog, select the red marked **Automatic reload of externally modified files** option as shown below.



3.3 Flashing the Executable File

Depending on the μ Vision version and the applicable μ Vision interface, the procedure for flashing the test driver application differs.

If you have built your test driver for execution using flash memory, you need to enable the **Use UVSC Flash** attribute within TEE.



If this attribute is set to **true**, the test driver application will be flashed automatically upon start of the test execution. Please make sure that the μ Vision project you are using contains all relevant settings for flashing (i.e. flashing should work when triggered manually from within your μ Vision project).

4 Start the μ Vision Debugger for a test run manually

Since TESSY version 4.3.10 the tthd is used to start the μ Vision Debugger before the test run and it is used to close the debugger after all tests are executed. If you prefer or if you need to start the μ Vision Debugger manually and let TESSY connect to it, then you will have to remove the contents of the TEE attributes **Slave Call** and **Post Execution Command**.

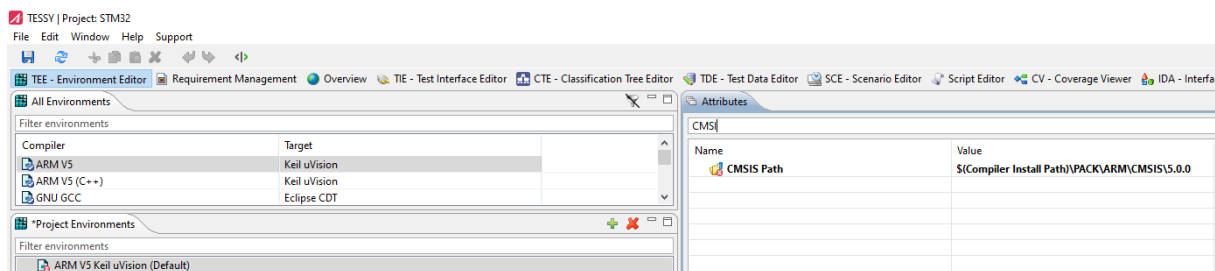
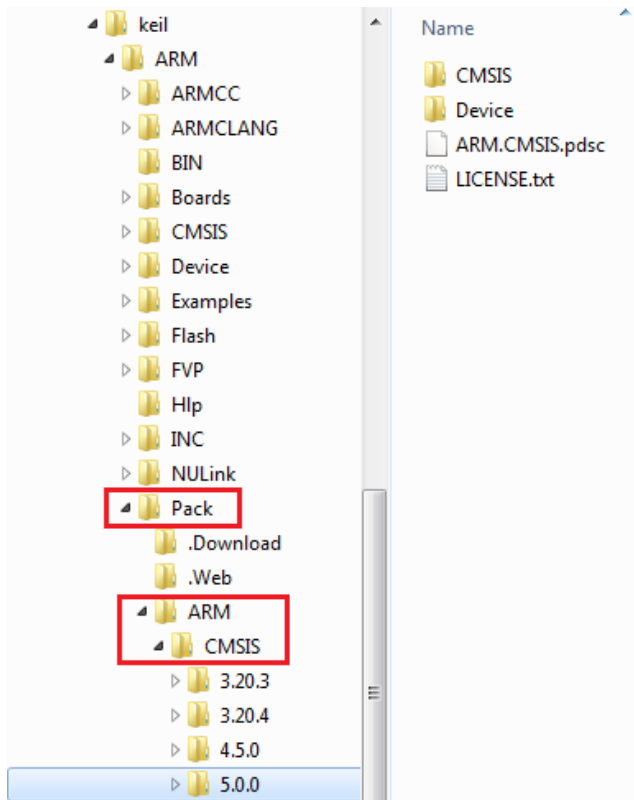
5 Example: Preparing TESSY for an ARM Microcontroller

Generally, TESSY provides a default configuration for each compiler/target combination. In case of ARM there is a configuration for STM32F10x available within the TESSY installation directory. This configuration was adapted from the μ Vision example project "Blinky" copied from the Keil Pack Installer.

If you are using another ARM microcontroller, you need to follow the steps below to create your specific configuration.

5.1 Adjust TEE attribute CMSIS Path

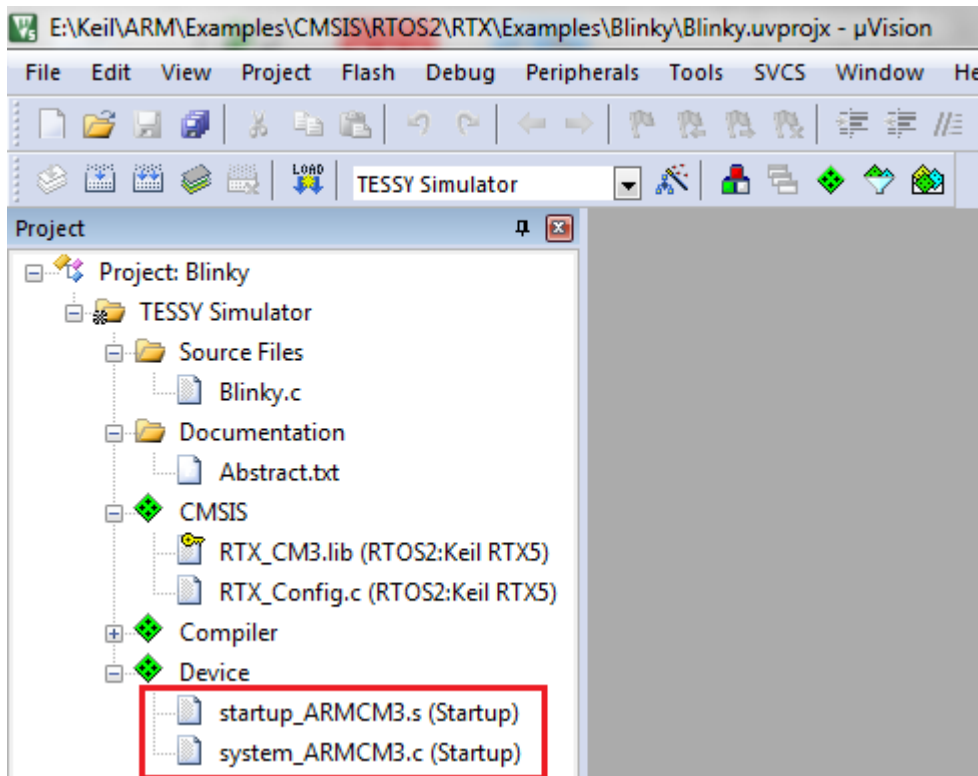
Browse to the path of your Keil ARM installation folder to look up the currently used library version and adjust the corresponding TEE attribute **CMSIS Path**.



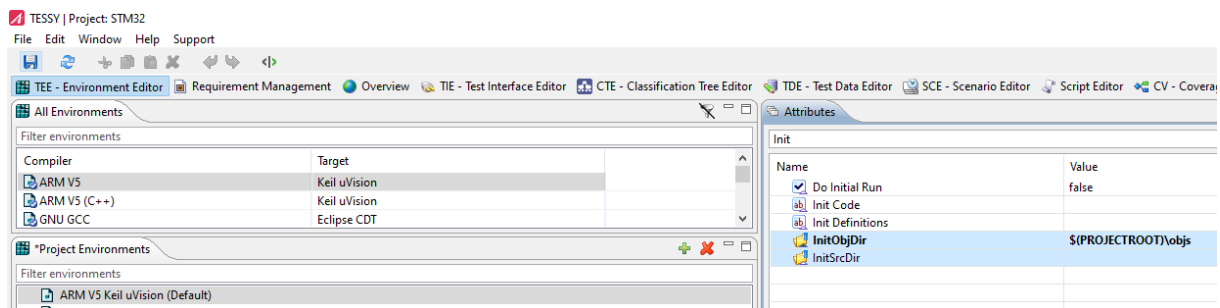
5.2 Step 1: Select a μ Vision Project

Choose a suitable μ Vision project file that contains all the necessary settings for your microcontroller. This may either be the project file that you are already using for development, or one of the Keil/ARM example projects from the Pack Installer or you may create a new project from scratch.

After successfully building the project within μ Vision, the following files should be available.



The easiest way is now to find the objects of these files in your μ Vision project folder and copy them into a separate folder within your TESSY project folder. Eventually, let TEE attribute **InitObjDir** point to it and keep TEE attribute **InitSrcDir** empty.



5.3 Step 2: Create a new TESSY Project

Please consult chapter 3.1 *TEE: Creating databases and working with the file system* from TESSY's user manual if you have not done by now.

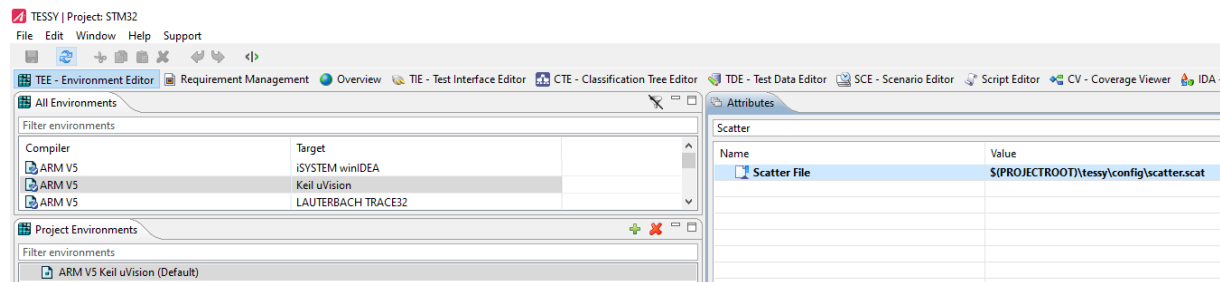
Please note: Make sure that the project was built by μ Vision before proceeding. Some of the files required for TESSY (e.g. the scatter file) will be generated during the μ Vision build process.

5.4 Step 3: TEE Configuration

Please consult chapter 6.5 *TEE: Configuring the test environment* from TESSY's user manual if you have not done by now.

As described in step 1 (see 5.2) you can either use the object files or set the attributes **Startup File** respectively **System File**.

Change the default configuration settings in order to reference the **Scatter File** attribute to the file created within step 1. We suggest that those attributes reference files from within the **\$(PROJECTROOT)** directory.



5.5 Step 4: Edit the Makefile Template

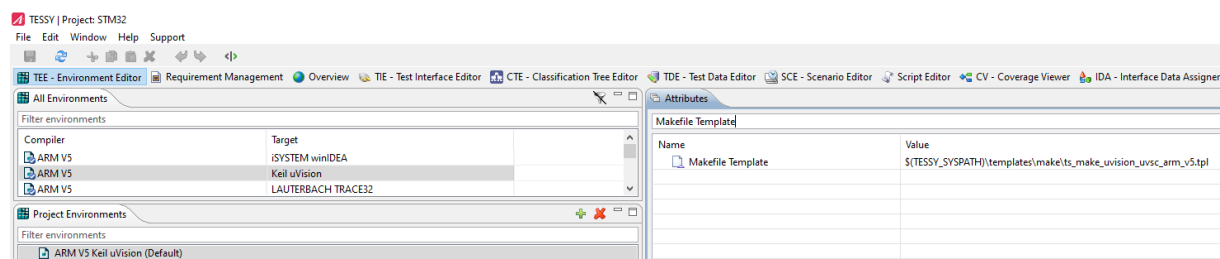
When using another microcontroller than the Cortex M3, you might need to adjust the makefile template. In this case, copy the makefile template from the TESSY installation folder (the most current file is the one with "uvsc" in its name)

```
... \sys\templates\make\ts_make_uvision_uvsc_arm_v5.tpl
```

into your project folder

```
$(PROJECTROOT)\config\ts_make_uvision_uvsc_arm_v5.tpl
```

Now open the TEE again and change the **Makefile Template** attribute accordingly (i.e. choose **Edit Attribute Value** from the context menu):

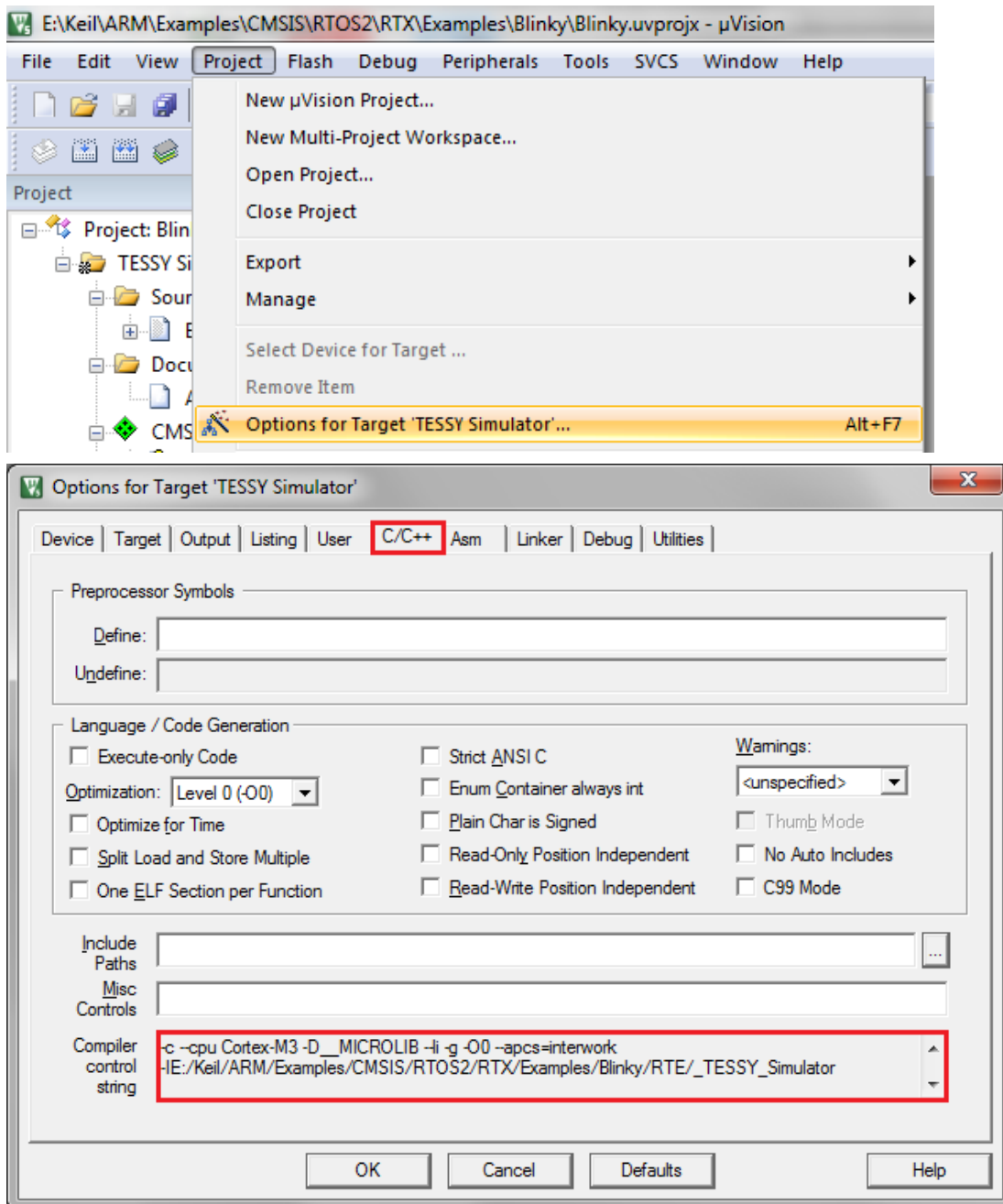


Open the makefile template within an editor and check compiler and linker settings. Variable `S_COMP_OPTIONS` contains the compiler settings. You must not remove the "TS_" defines because they are needed to build the proper TESSY communication modules. Variable `S_LINK_OPTIONS` contains the linker settings.

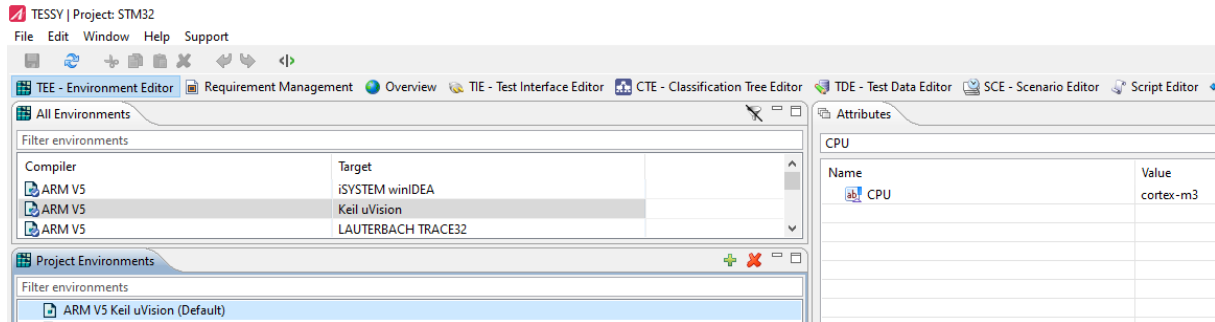
```
#
# SLAVE
# (You may change these settings to use your own compiler/linker options)
#
S_COMP_OPTIONS := $(CPU_OPT) -g -O0 --apcs=interwork -I$(ARM_CC_INC) -I$(ARM_INC) -I$(ARM_DEVICE_INC) \
-DTESSY -DTS_SLAVE -DTS_RVDS_ARM -DTS_ARM -DTS_HAVE_INT64 -DTS_HAVE_FLOAT64 \
-DTS_ALIGN_BUFFER=$(ALIGN_COMM_BUFFER) -DTS_HAVE__int64 -D_MICROLIB
S_LINK_OPTIONS := --library_type=microlib $(CPU_OPT) --scatter $(SCATTER_FILE) --autoat --libpath $(ARM_LIB)
S_INCLUDES := -I$(TESSY_SYS_DOS)\include\tessy\comm -I$(MODULE_PATH_DOS)
```

TESSY Application Notes

The respective settings can be found within the μ Vision project that we created within step 1 (Refer to chapter 5.2). Open the **Options for Target 'TESSY Simulator'** dialog and switch to the **C/C++** tab.



Within the **Compiler control string** you will find the **--cpu** option. Adjust the value within TEE accordingly.



All relevant configuration settings are now available within the **config** directory of your project.

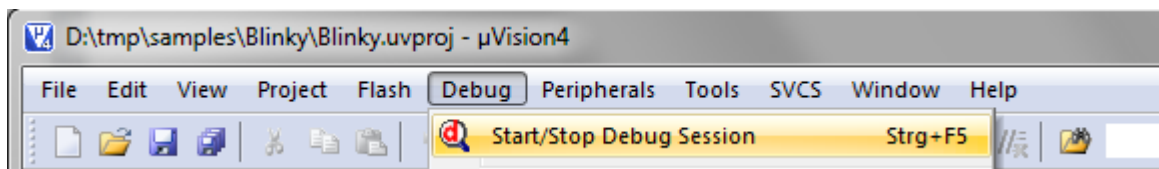
6 Troubleshooting

6.1 Error Executing the Test

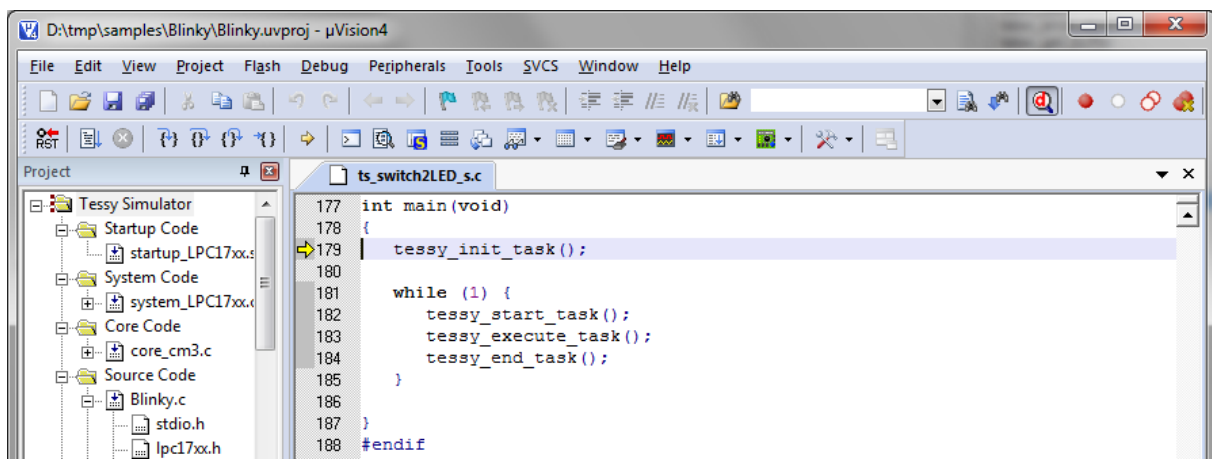
If TESSY fails to execute the test, please check the following:

- make sure that **µVision is running** and the correct **Project Target** is selected within µVision as described in the previous chapters
- check the **Output** settings for the currently selected µVision project target (default is `c:\tessy\testarea` for the path and `ts_slave.hex` (latest versions of TESSY use `.axf` instead of `.hex` as file extension) for the executable) (see 3.1.3)

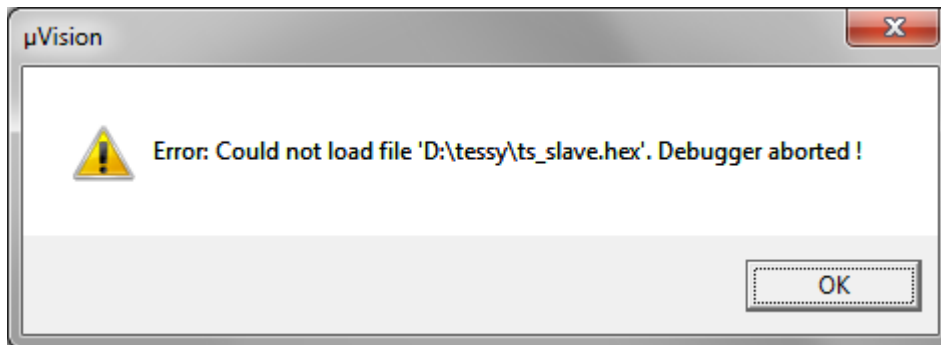
In order to check your settings, you may press the **Debug** button within µVision to enter the debug mode.



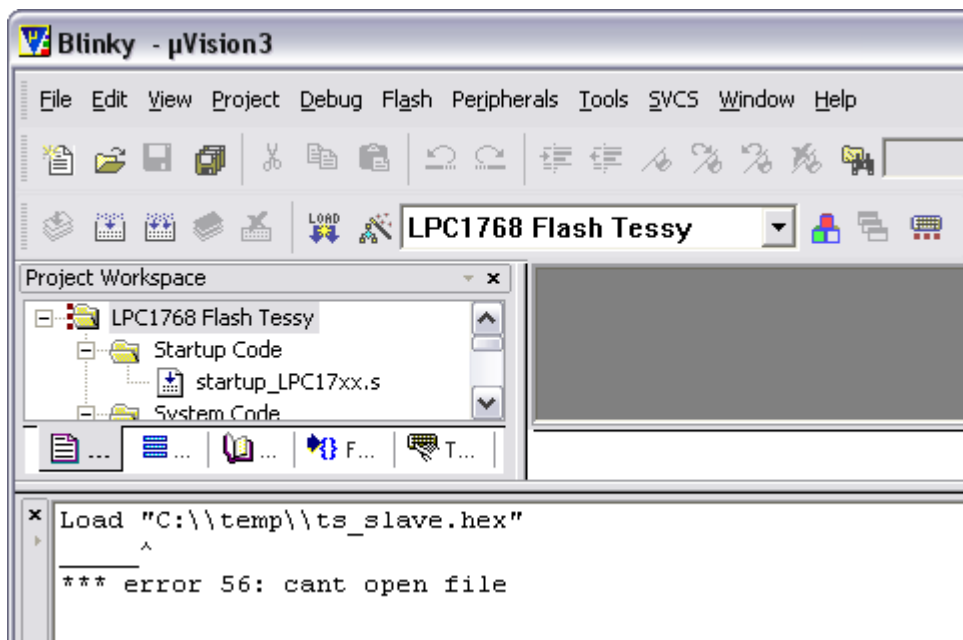
This will load the executable test driver into the debugger. If everything is correct, the debugger should stop the target application at the main function of the TESSY test driver as shown below.



If the **Output** setting of the currently selected **Project Target** is wrong, μ Vision4 and μ Vision5 will show the following dialog when trying to enter into debug mode:

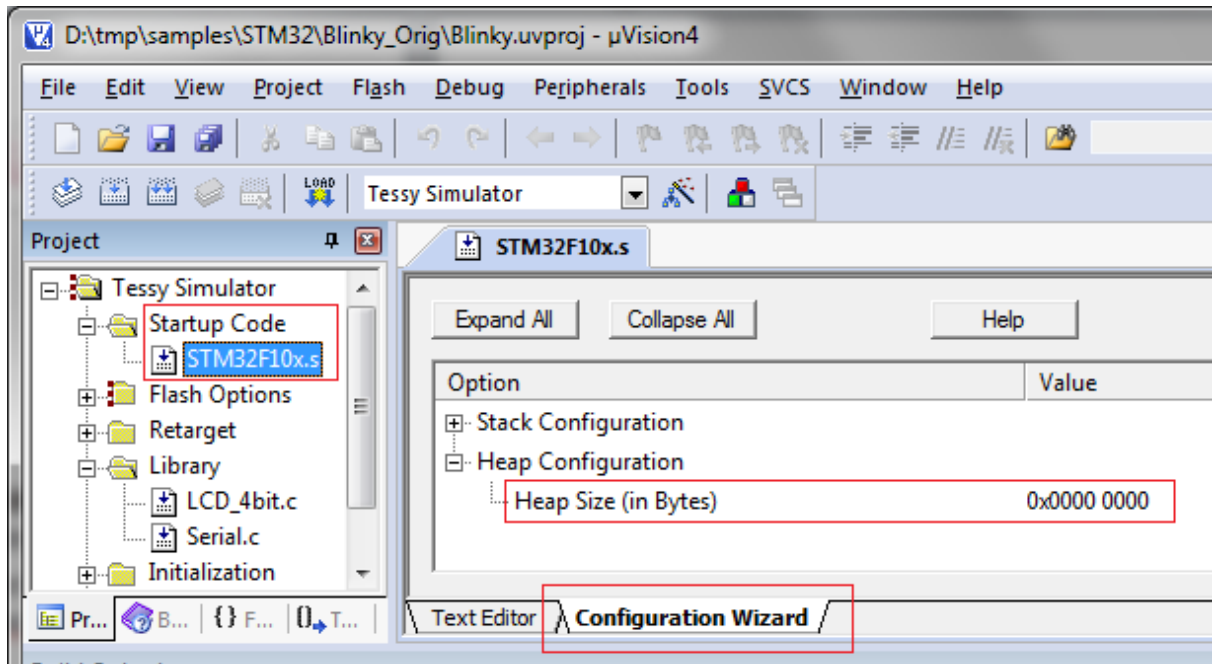


When working with μ Vision3 you will see something like the following message within the μ Vision message window:



6.2 HEAP Space Setting for C++

If you are executing a C++ test object and the test driver binary fails to start executing properly (i.e. does not reach function `main()` when started manually) you may need to adjust the HEAP space (this is zero by default). This setting is provided within the startup code (e.g. `STM32F10x.s`). You may either edit the file directly or use the **Configuration Wizard** from within μ Vision to change the settings.

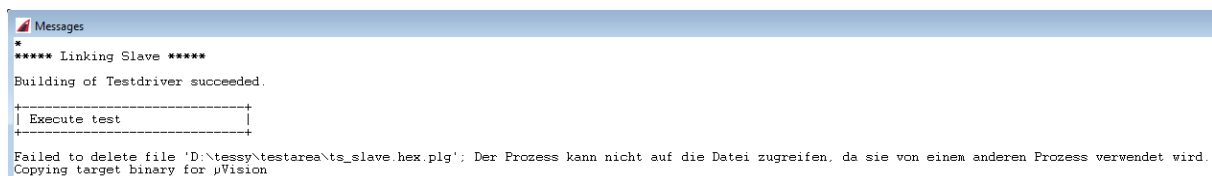


The default value within the STM32F10x.s startup code file delivered with TESSY is 0x1000.

Please note: Other startup code files will have different options within the Configuration Wizard.

6.3 Warning after Starting a Test

The following warning message may be displayed when repeatedly running tests with µVision:



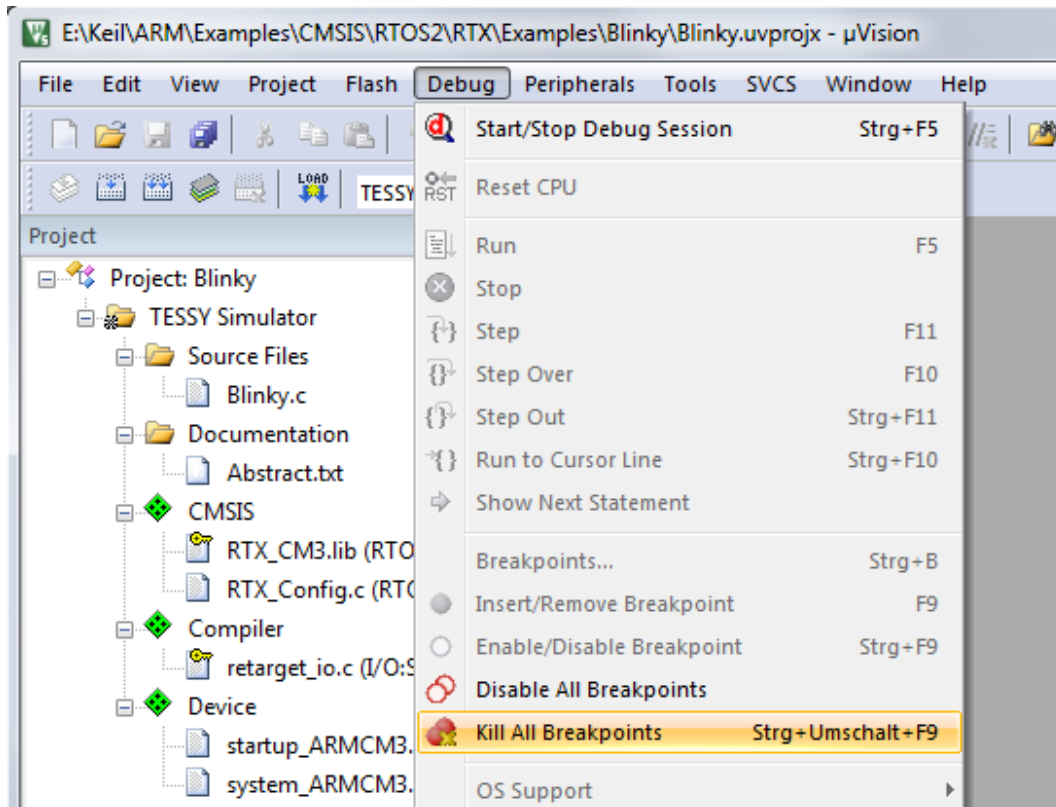
This warning is caused by TESSY when trying to delete all files within the testarea directory as preparation for the next test run. If a test with µVision has been run before, µVision still has an open file descriptor on this temporary file.

⇒ This warning can be ignored.

6.4 Removing Breakpoints

When running a TESSY test non-interactively, i.e. without the **Define Breakpoint** feature selected, TESSY will normally delete all breakpoints within the µVision project before starting the actual test.

If this fails, there may be invalid breakpoints left from previous test runs or debugging sessions. In this case select the **Kill All Breakpoints** menu entry from the **Debug** menu before starting the test with TESSY.



6.5 The debugger IDE window vanishes

By default, the μ Vision debugger window will be hidden unless the test was started with **Define Breakpoint at Test Object** or logging is enabled. The IDE's window will vanish from the taskbar. The μ Vision window can be restored by typing ALT+TAB and selecting μ Vision. If the debugger window should always be visible set TEE attribute **Hide IDE** to false.