

# CodeWarrior Development Studio

## 1 Abstract

This document describes the setup and handling for the CodeWarrior Development Studio V10.2 and later. Please note the difference to the legacy versions of CodeWarrior: This new integration supports the new Eclipse based version of CodeWarrior Development Studio.

## Table of Contents

1	Abstract .....	1
2	Introduction.....	2
3	Setup .....	2
3.1	TESSY Environment Editor (TEE) Settings.....	3
3.1.1	Startup Code and Linker File.....	3
3.1.2	Usage of Evaluation Boards.....	4
3.1.3	Debug Configuration .....	4
4	Starting the Debug Service.....	6

## 2 Introduction

TESSY supports the latest CodeWarrior IDE from Freescale (currently V10.7) that is built on Eclipse. The integration requires the setup of Eclipse plugins within the CodeWarrior IDE installation directory.

The communication between TESSY and CodeWarrior is based on a TCL Script Server that needs to be running when executing tests.

## 3 Setup

Please copy all jar packages from the TESSY installation folder:

```
...\TESSY_4.X\sys\targets\codewarrior\plugins-10.2
```

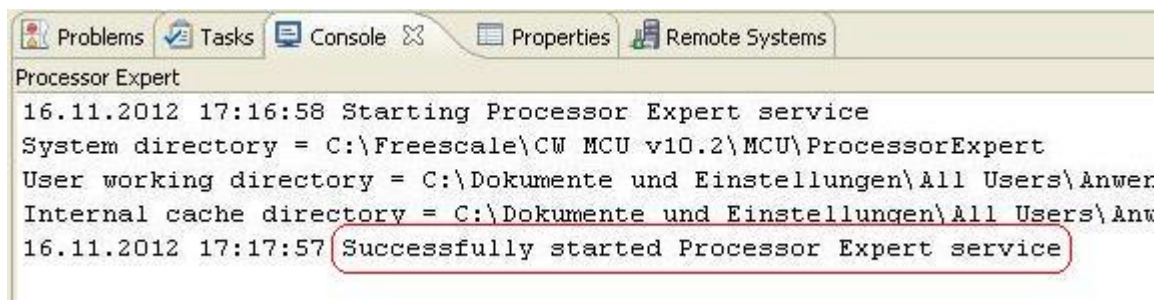
into the `dropins` folder of your Freescale CodeWarrior installation directory, e.g.

```
C:\Freescale\CW MCU v10.7\eclipse\dropins
```

or into the `plugins` folder of an older Freescale CodeWarrior installation directory

```
C:\Freescale\CW MCU v10.2\eclipse\plugins
```

**Important Note:** When starting Freescale CodeWarrior, wait for CodeWarrior's Processor Expert service before you launch the TESSY TCL script server.



```
Processor Expert
16.11.2012 17:16:58 Starting Processor Expert service
System directory = C:\Freescale\CW MCU v10.2\MCU\ProcessorExpert
User working directory = C:\Dokumente und Einstellungen\All Users\Anwe
Internal cache directory = C:\Dokumente und Einstellungen\All Users\Anw
16.11.2012 17:17:57 Successfully started Processor Expert service
```

The screenshot above shows the output of the CodeWarrior Console view when the processor expert service has started.

The plugin can also be configured to start automatically for automated testing environments by adding the following line to `cwide.ini`, which is found within the `eclipse` folder of the CodeWarrior installation directory.

```
-Dcom.razorcat.targets.enableAutoStart=true
```

The default time to wait before the plugin starts is 60 seconds. You can alter this amount of time in seconds by adding the following to `cwide.ini` as well

```
-Dcom.razorcat.targets.autoStartDelaySeconds=50
```

which would set the value to 50 seconds. Please, keep in mind, that the CodeWarrior has to be ready before the plugin can be launched. If it is launched too early, it may fail.

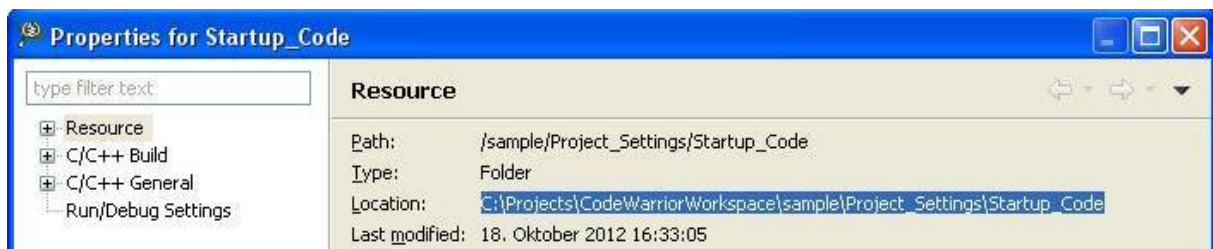
### 3.1 TESSY Environment Editor (TEE) Settings

There are some necessary settings within the environment editor TEE that reference settings of your CodeWarrior project.

#### 3.1.1 Startup Code and Linker File

Start TESSY and from within TESSY start the Environment Editor (TEE) using the File menu.

Start Freescale CodeWarrior and open your current project. From the properties dialog of your **Startup\_Code** tree node copy the location:



Create or fill the following TEE attributes with suitable values:

Attribute name	Type	Description
InitObjDir	Directory	Any directory where additional object files will be compiled into. The directory must exist.
InitSrcDir	Directory	Any directory that will contain additional source files that shall be linked to the test application binary. Normally, this will be the startup code of your CodeWarrior project.
Linker File	File	The linker file of your CodeWarrior project

The **InitSrcDir** attribute shall contain the path to the **Startup\_Code** tree node of your CodeWarrior project like shown below:



Make sure the path given by InitObjDir exists. Enter the location of your project's linker file likewise.

### 3.1.2 Usage of Evaluation Boards

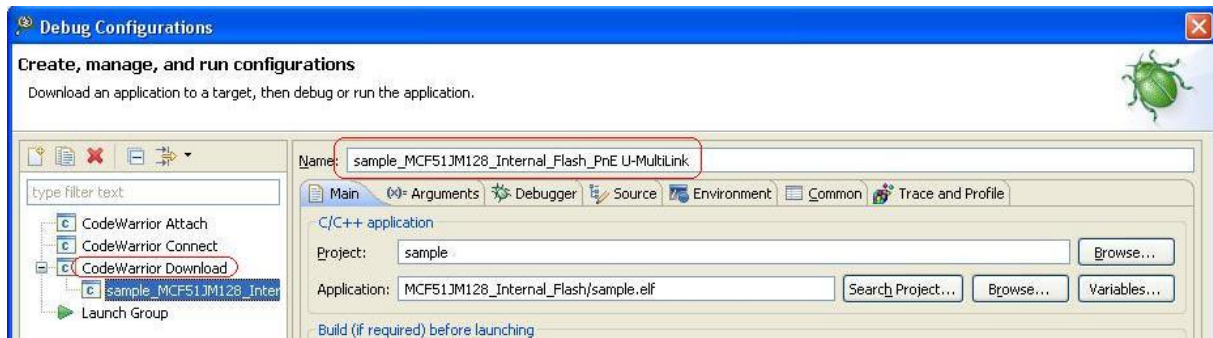
If you are using an evaluation board and not a simulated target, you need to set the TEE attribute **Use Board File** to “true” in order to compile and link the target board configuration files into the test application binary:



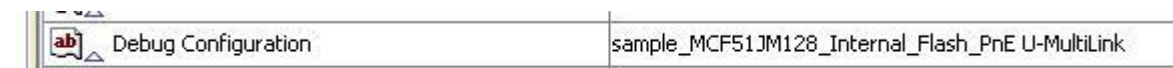
For more information about startup code see application note *Startup Code of the Test Program*.

### 3.1.3 Debug Configuration

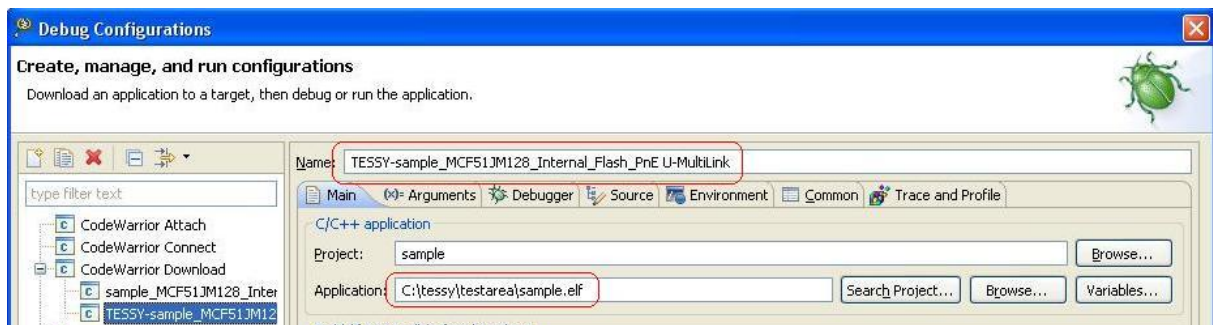
You need to tell TESSY about your CodeWarrior debug configuration. Open the **Debug Configuration** of Freescale CodeWarrior:



Copy the name of the debug configuration into the TESSY TEE **Debug Configuration** attribute like shown below:



TESSY will use a copy of this Debug Configuration for its test runs.



## TESSY Application Notes

---

This copy will be created on-the-fly and will be removed when you disable the TESSY service.

## 4 Starting the Debug Service

When the TESSY plugins have been installed successfully into the CodeWarrior IDE, there should be a **Start/Stop TESSY Script** button within the CodeWarrior main toolbar.

In order to prepare CodeWarrior for test execution, you need to start the TESSY TCL Script Server by pressing the TESSY button from within the CodeWarrior toolbar like shown below:

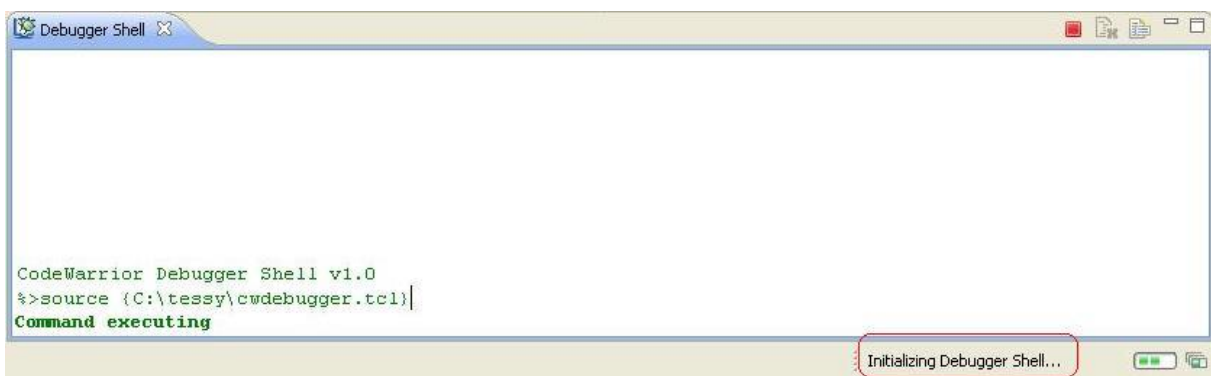


If you have setup more than one TESSY TEE configuration for your target a dialog will be shown and let you choose the environment you want to use.

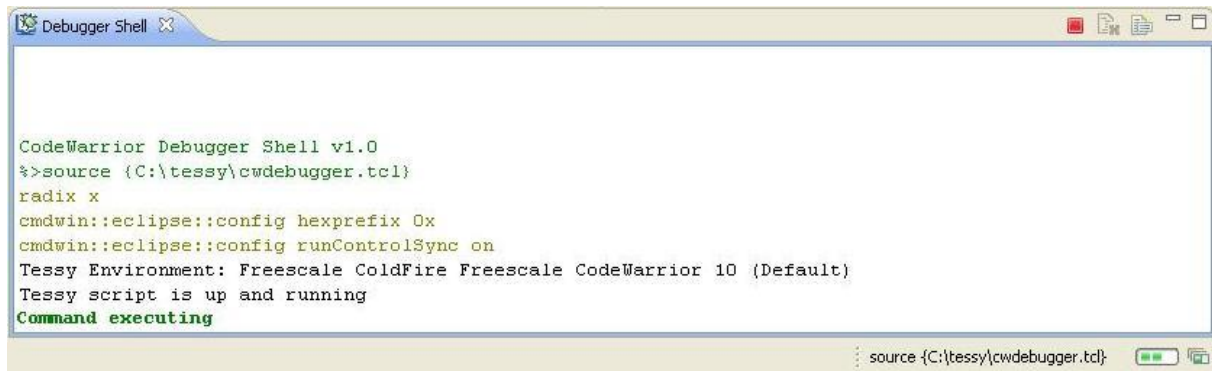


If there is only one environment the dialog will not show.

The TESSY service will launch the **Debugger Shell** of Freescale CodeWarrior. The first time after start-up will take some time. So, please, be patient.

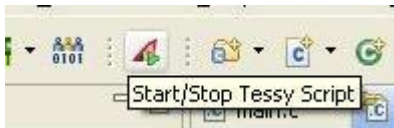


When the TESSY TCL server script is loaded and ready for test run it will display the following message:



```
CodeWarrior Debugger Shell v1.0
%>source {C:\tessy\cwdebugger.tcl}
radix x
cmdwin::eclipse::config hexprefix 0x
cmdwin::eclipse::config runControlSync on
Tessy Environment: Freescale ColdFire Freescale CodeWarrior 10 (Default)
Tessy script is up and running
Command executing
```

You can now start your TESSY test runs. When done you may stop the TESSY TCL server script by pressing the TESSY button again.



This will restore your Freescale CodeWarrior desktop, i.e. close the **Debugger Shell** if it had not been open before and switch back to the perspective that was open when you pushed the TESSY button initially.