



Getting Started with Freescale MQX™ RTOS for Kinetis SDK and MDK-ARM Keil® µVision5®

1 Read Me First

This document describes the steps required to configure Keil® MDK to build, run, and debug MQX™ RTOS demo applications and necessary driver libraries provided in the KSDK framework. The Hello World demo application targeted for the TWR-K64F120M Tower System hardware platform is used as an example in this guide.

Additionally, more information related to Freescale Kinetis platform support in µVision®5 tools is in the arm.com/files/pdf/Kinetis_LAB.pdf document.

Contents

1	Read Me First.....	1
2	Import project files into KEIL workspace.....	2
3	Build library project files.....	3
4	Build the application project file	5
5	Run a demo application.....	7
6	Revision history	13

2 Import project files into KEIL workspace

Each demo application of MQX RTOS for Kinetis SDK requires several libraries to be built together with a demo application itself. For user convenience, the Freescale MQX RTOS package contains multi-project workspace files containing all required libraries. The workspaces are available on following locations:

`<install_dir>/rtos/mqx/mqx/examples/<demo_name>/build/mdk/<demo_name>_<board_name>/<demo_name>_<board_name>.uvmpw`

For example, with hello demo application for TWR-K64F120M, open the workspace file in the Keil MDK.

`<install_dir>/rtos/mqx/mqx/examples/hello/build/mdk/hello_twrk64f120m/hello_twrk64f120m.uvmpw`

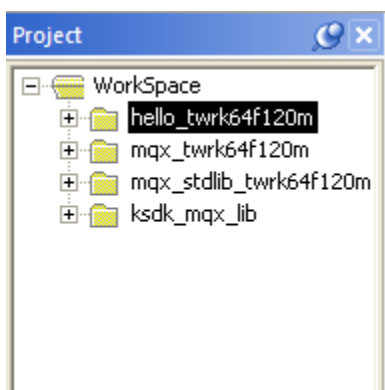


Figure 1 : Import project files into workspace

3 Build library project files

This section will guide you to build the library project files of the hello example in MQX RTOS for KSDK for TWR-K64F120M. Follow the steps to build libraries with different build target at the same time.

1. Go to “Project” and select “Batch Build” from the toolbar menu options to start the batch build.

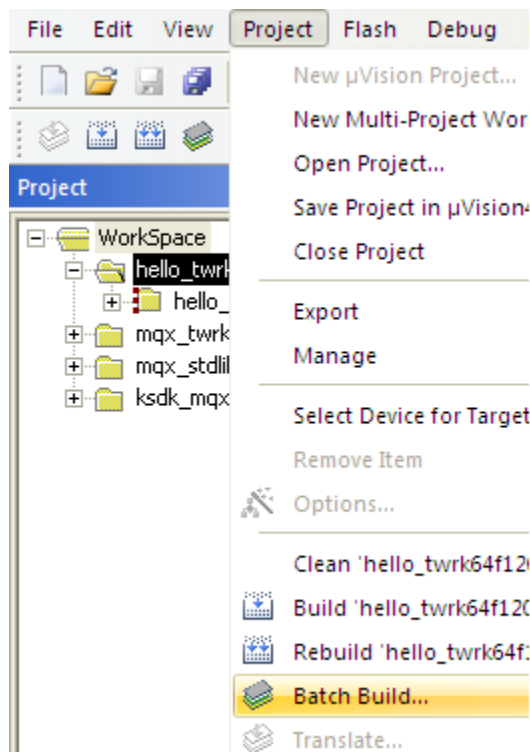


Figure 2: Start batch build

2. In the Batch Build dialog window, check the boxes next to the library with the desired build target that you want to build.

- Click the “Build” button to build the libraries for the first time, or the “Rebuild” button (in red) to build the libraries again.

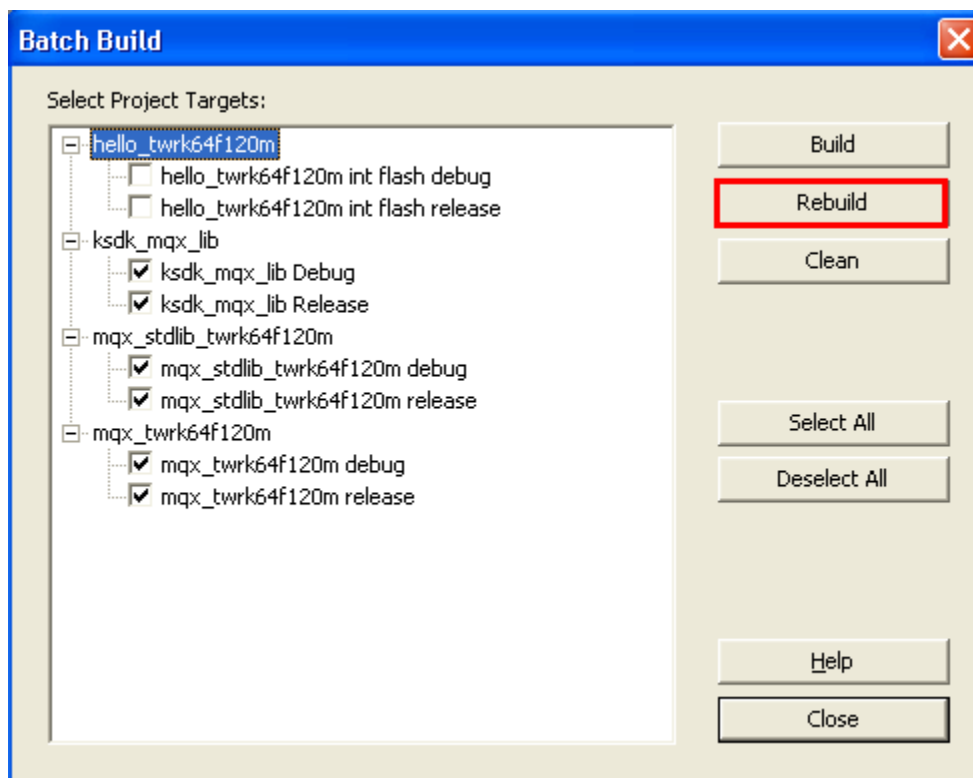


Figure 3: Select projects to build

When complete, check that the library files are generated in folders

<install_dir>/rtos/mqx/lib/<board_name>.mdk/<build_target>/mqx/<lib_mqx>.lib

<install_dir>/rtos/mqx/lib/<board_name>.mdk/<build_target>/mqx_stdlib/<lib_mqx_stdlib>.lib

<install_dir>/lib/ksdk_mqx_lib/mdk/<device_name>/<build_target>/<libksdk_platform_mqx>.lib

For example with the hello example of MQX RTOS for KSDK for TWR-K64F120M and Debug target, the following library files are generated.

<install_dir>/rtos/mqx/lib/twrk64f120m.mdk/debug/mqx/<lib_mqx>.lib

<install_dir>/rtos/mqx/lib/twrk64f120m.mdk/debug/mqx_stdlib/<lib_mqx_stdlib>.lib

<install_dir>/lib/ksdk_mqx_lib/mdk/K64F12/debug/<libksdk_platform_mqx>.lib

4 Build the application project file

After building the libraries, build the application project file. The steps to build the hello example of MQX RTOS for KSDK for TWR-K64F120M are described here.

1. In the workspace, set the hello_twrk64f120m as the active project by right clicking the project and select “Set as Active Project”.
2. Select the appropriate build target.

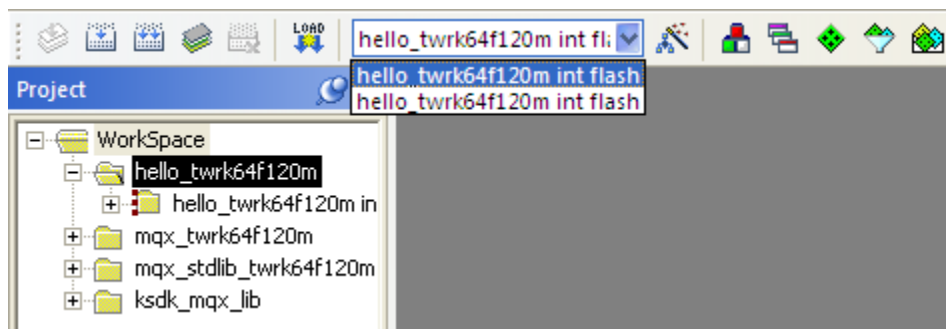


Figure 4: Select build target for hello project

3. Click the build button (in red) to build the project.

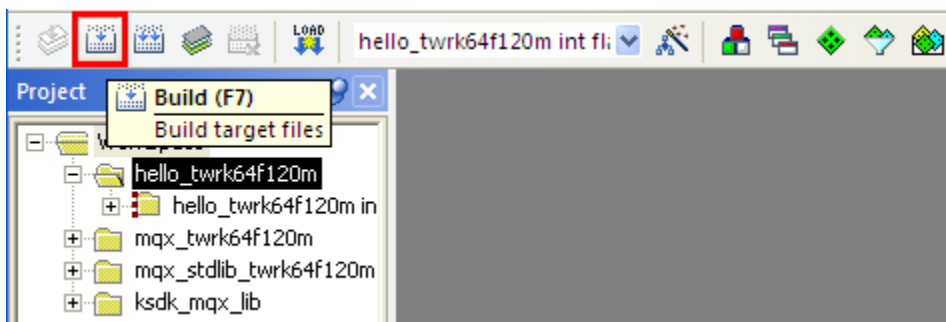


Figure 5: Build hello example

4. When the build of the hello finishes, the build output message is similar to this image. Any error or warning message will be in the “Build Output” tab view. Note that a warning message will not affect the linking process and the application image is generated normally.

```

Build Output
Build Project 'hello_twrk64f120m' - Target 'hello_twrk64
compiling hello.c...
compiling init_hardware.c...
compiling init_bsp.c...
compiling mqx_init.c...
compiling mqx_main.c...
compiling fsl_misc_utilities.c...
compiling fsl_debug_console.c...
compiling print_prv.c...
compiling scan_prv.c...
compiling fp_prv.c...
compiling gpio_pins.c...
compiling pin_mux.c...
compiling hardware_init.c...
linking...
int flash debug\hello.axf: Warning: L6319W: Ignoring --b
int flash debug\hello.axf: Warning: L6803W: Relocation #
Program Size: Code=27984 RO-data=1808 RW-data=184 ZI-dat
Finished: 0 information, 2 warning and 0 error messages.
FromELF: creating hex file...
"int flash debug\hello.axf" - 0 Error(s), 2 Warning(s).

```

Figure 6: Build output message

5 Run a demo application

To download and run the application, follow these steps.

1. Connect the development platform to your PC via USB cable between the OpenSDA USB connector and the PC USB connector.
2. Open the terminal application on the PC, such as PuTTY or TeraTerm, and connect to the OpenSDA serial port number. Configure the terminal with these settings:
 - a) 115200 baud rate
 - b) No parity
 - c) 8 data bits
 - d) 1 stop bit

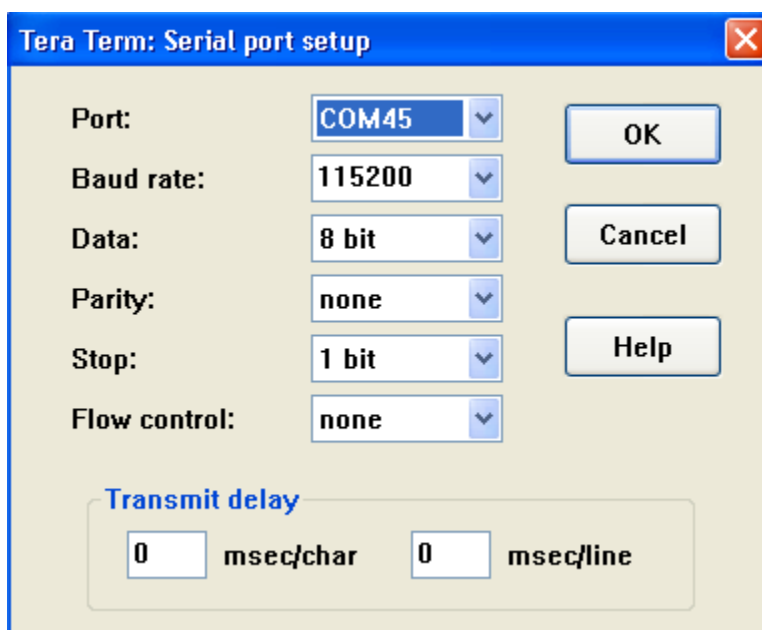


Figure 7: Set up serial terminal

- Right click the hello project and select “Options for <demo_name>” or press Alt+F7 to open Options dialog window for the project. For the hello example of MQX RTOS for KSDK for TWR-K64F120M, open the specific window.

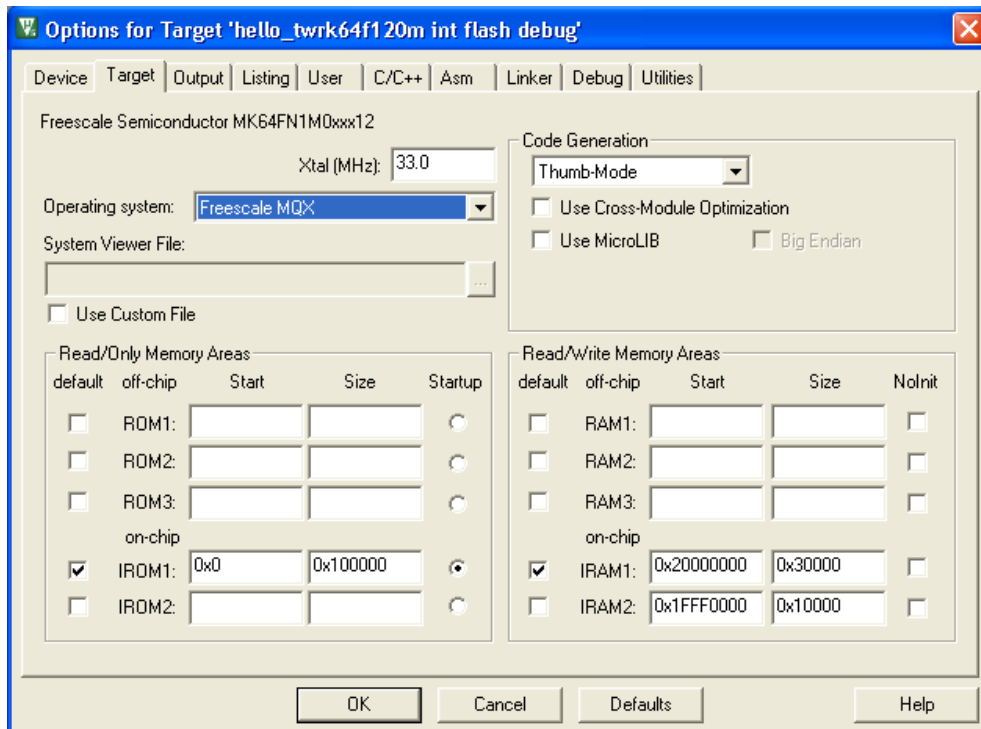


Figure 8: Project options

Note

In case “Freescale MQX RTOS” option is not available in your Keil MDK IDE, install the plug-in can by running
 <KSDK_DIR>/tools/mqx_plugins/keil_extensions/MDK_MQXViewer_AddOn.exe executable file.

4. Navigate to the “Debug” option and select the other debugging tool to make sure the Debugger you use is suitable for the development platform.

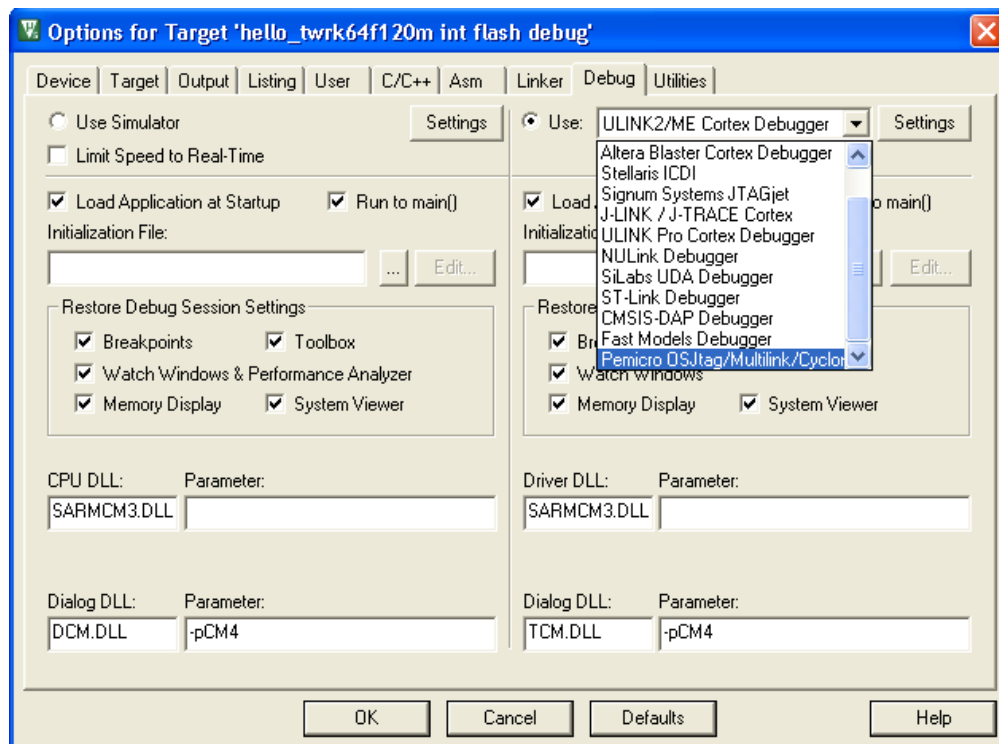


Figure 9: Select Debugger

- After selecting of appropriate Debugging tool, click the “Settings” button, next to the drop down menu of the Debugger selection, to configure the Debugger. This allows the Keil μ Vision5 to be able to connect to the board via the Debugger. This figure shows the P&E Micro debugger as an example, but you may use different debugger. In the open window, select the right interface (in red) and Port (in pink). Click the “Refresh List” button (in orange) to update the status of the board. In some cases, the target CPU also needs to be changed (in blue). Check the SWD option (in yellow). Click the “OK” button (in green) when done.

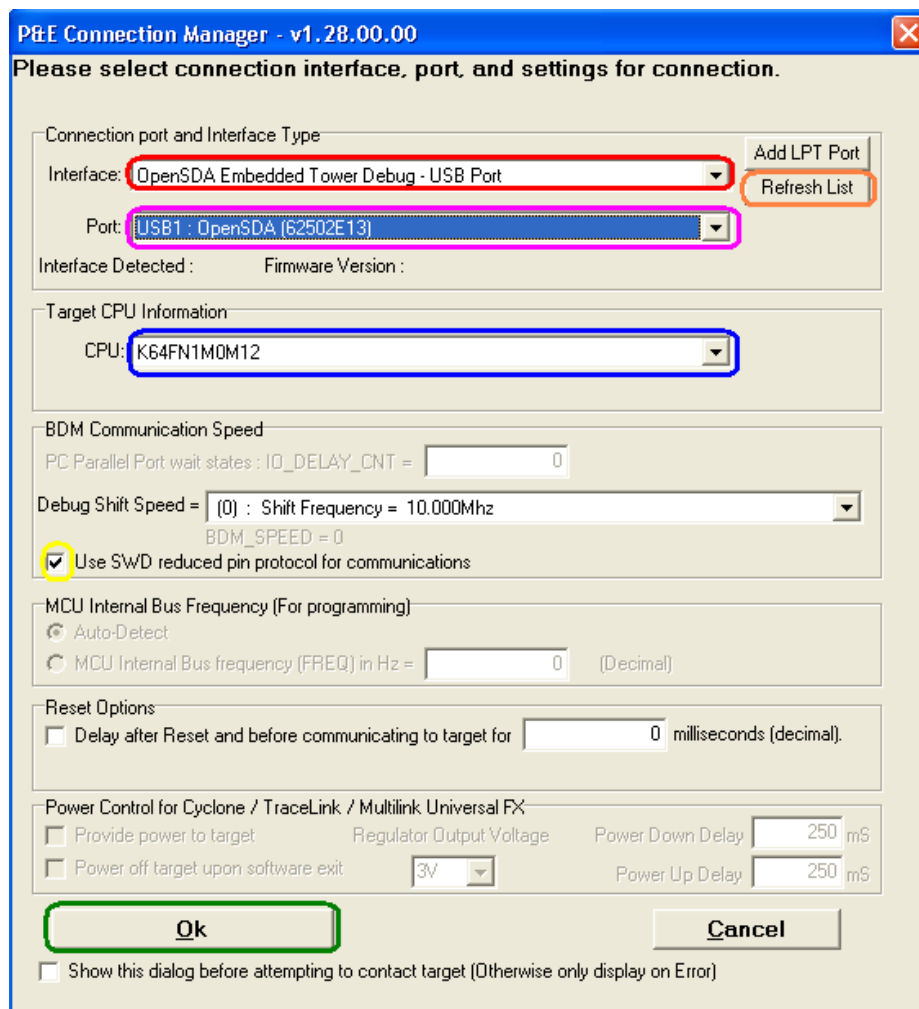


Figure 10: Debugger configurations

1. Click the “Download” button (in red) to download the application to the target platform.

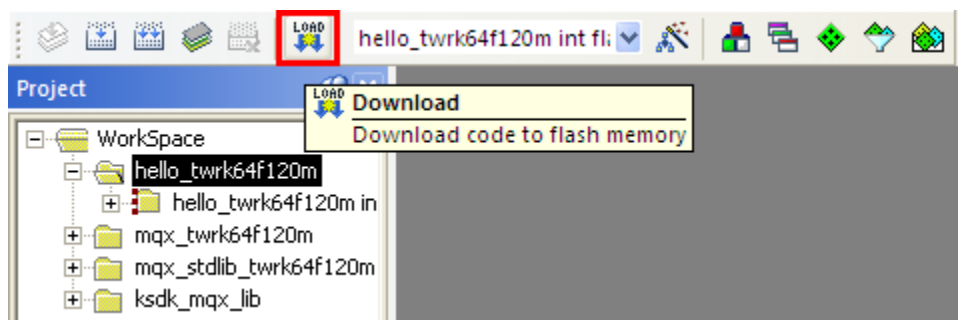


Figure 11: Download application image

2. Click the “Start/Stop Debug Session” button (in red) to debug the application.

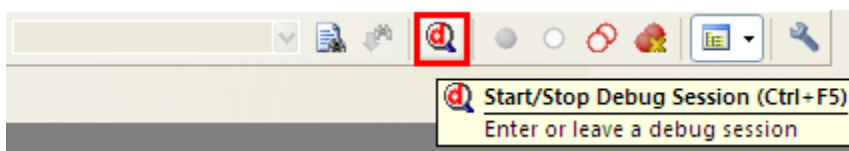


Figure 12: Start debug session

3. The application runs and stops to the main() function.

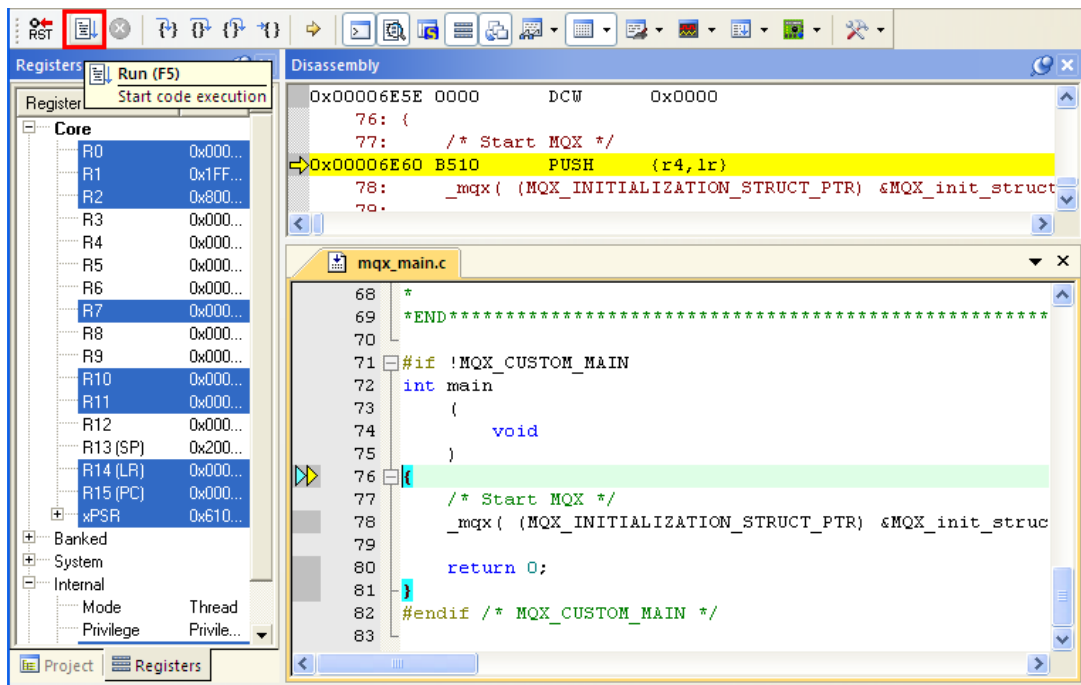


Figure 13: Run the application

4. Click the “Run” button (in red) to run the application.

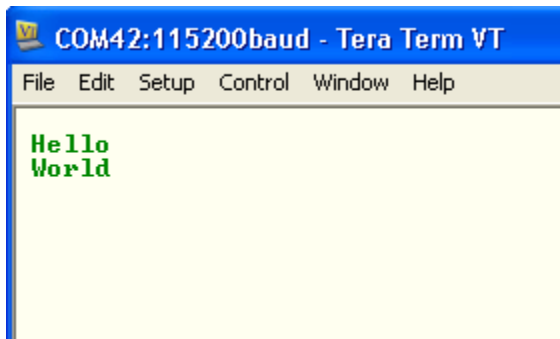


Figure 14: Output of hello example of MQX RTOS for KSDK

6 Revision History

This table summarizes revisions to this document.

Table 1 Revision History		
Revision number	Date	Substantial changes
1	04/2015	Kinetis SDK 1.2.0 release
0	12/2014	Kinetis SDK 1.1.0 release

How to Reach Us:**Home Page:**

freescale.com

Web Support:

freescale.com/support

Information in this document is provided solely to enable system and software implementers to use Freescale products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document.

Freescale reserves the right to make changes without further notice to any products herein. Freescale makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in Freescale data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including "typicals," must be validated for each customer application by customer's technical experts. Freescale does not convey any license under its patent rights nor the rights of others. Freescale sells products pursuant to standard terms and conditions of sale, which can be found at the following address: freescale.com/SalesTermsandConditions.

Freescale, the Freescale logo, and Kinetis are trademarks of Freescale Semiconductor, Inc., Reg. U.S. Pat. & Tm. Off. All other product or service names are the property of their respective owners. ARM, ARM powered logo, Keil, µvision, and Cortex are registered trademarks of ARM Limited (or its subsidiaries) in the EU and/or elsewhere. All rights reserved.

© 2015 Freescale Semiconductor, Inc.

