NXP Semiconductors
User's Guide

Document Number: MCUXSDKLWIPUG
Rev. 8, 03/2018

lwIP TCP/IP Stack and MCUXpresso SDK Integration User’s Guide

Contents

1 Overview

This document describes how to compile and run the lwIP TCP/IP stack examples. This document also provides the board-specific information related to the jumper and hardware settings.

2 Release scope

2.1 Hardware

This release includes support for the TWR-K64F120, TWR-KV58F220M, TWR-K65F180M, and TWR-K35F180M Tower System modules, EVK-MIMXRT1050, FRDM-K64F and FRDM-K66F Freedom development platforms, and LPCXpresso54xxx derivatives.

3 Requirements for running lwIP demos

3.1 Supported hardware platforms

- Tower System development platforms:
  - TWR-SER serial peripheral board
  - TWR-ELEV boards (pair)
  - USB cable
  - Ethernet cable

NOTE

For Tower System boards, extra TWR-SER and TWR-ELEV modules are needed

- Freedom development platforms:
  - USB cable
  - Ethernet cable
- LPCXpresso development platforms:
  - USB cable
  - Ethernet cable

3.2 Software development tools

- MCUXpresso SDK release package that includes the lwIP TCP/IP package
- MCUXpresso IDE
- IAR Embedded Workbench for Arm® version 7.80.2
- Keil® µVision® 5 Integrated Development Environment Version 5.21a
- Kinetis Design Studio IDE Version: 3.2.0

NOTE

MCUXpresso targets only. KDS IDE does not support LPC boards.

- Makefiles support with GCC revision 5-2016-q3-update from GNU Arm Embedded Toolchain

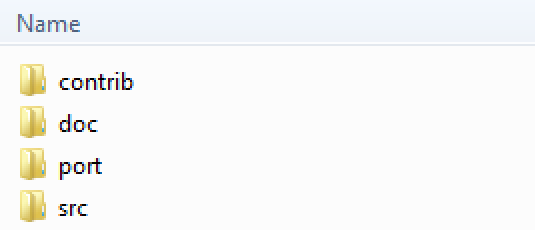
3.3 Board jumper settings

The Ethernet-related jumper settings are described in this document. For other jumper settings, see the board-specific user's guide. By default, the lwIP stack uses RMII mode. Follow the below hardware configuration:

- TWR-K64F120M
  - J32 1-2: Use the external clock from the CLOCKIN0 to keep the synchronization with the external PHY on TWR-SER Tower System module.
- TWR-K65F180M
  - No jumper specifications.
- TWR-SER
  - J2 3-4: Ethernet PHY Clock Select 50 MHz, RMII mode. Cut off other connections on this jumper.
  - J3 2-3: Route 50 MHz clock to CLOCKIN0. Cut off other connections on this jumper.
  - J12 9-10: Ethernet PHY Configuration, pull-up CONFIG0, RMII select. Cut off other connections on this jumper.
- Freedom FRDM-K64F platform
  - No jumper specifications.

4 lwIP code structure

The lwIP code is located in this folder: <install_dir>/middleware/lwip_2.0.0. The lwIP folder includes the source code. There are two subfolders in the lwip folder as shown in the figure.



Name

contrib
doc
port
src

Figure 1. lwIP folder structure

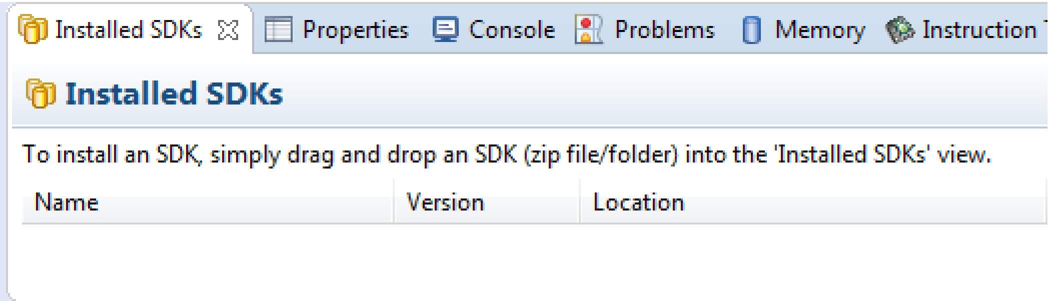
- contrib
  - This subfolder includes third party demo applications provided by the lwIP project (lwIP project has two separate downloads, lwIP and lwIP contrib).
- doc
  - This subfolder includes some readme files and some files for generating Doxygen documentation from lwIP sources.
- src
  - This subfolder includes the lwIP 2.0.0 source code which can be downloaded from this link: savannah.gnu.org
- port
  - This subfolder includes the adapter files which can make the lwIP stack run on the MCUXpresso SDK and different RTOSes.

5 Compiling or running the lwIP stack and demos

5.1 Step-by-step guide for MCUXpresso IDE

This section shows how to compile and run demos in the MCUXpresso IDE.

1. After opening an existing or creating a new workspace, drag and drop the SDK zip file into the "Installed SDKs" view to install an SDK. Next, import the lwIP demos.



Installed SDKs

To install an SDK, simply drag and drop an SDK (zip file/folder) into the 'Installed SDKs' view.

Name	Version	Location

Figure 2. Install SDKs view

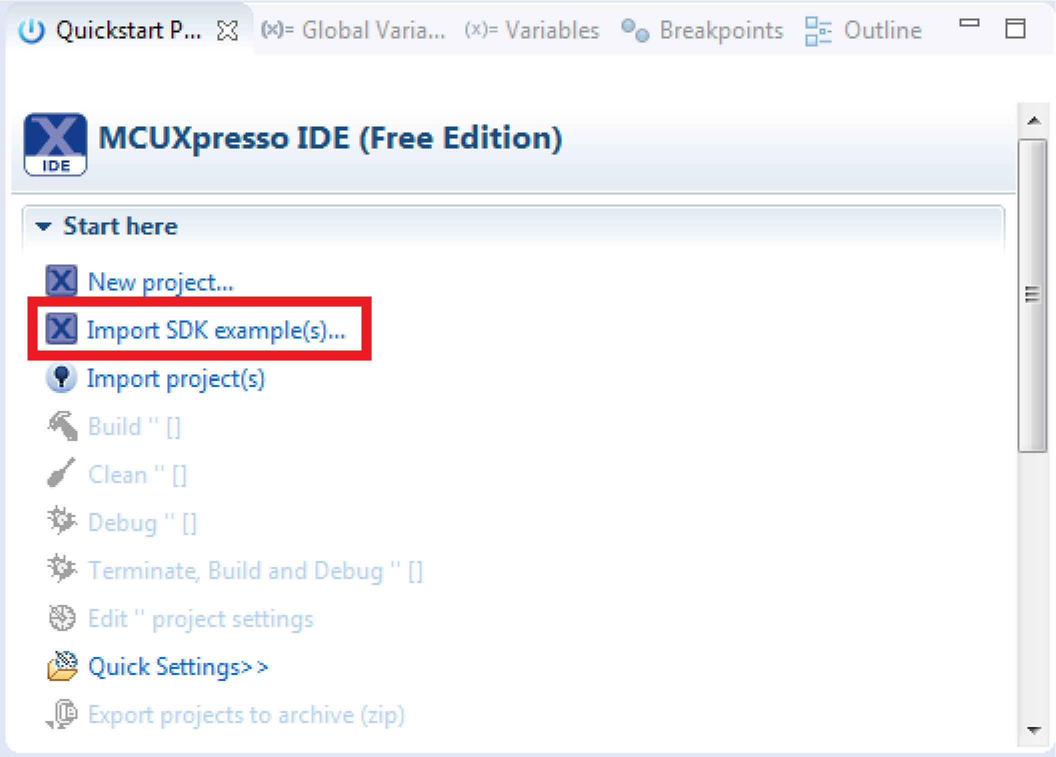
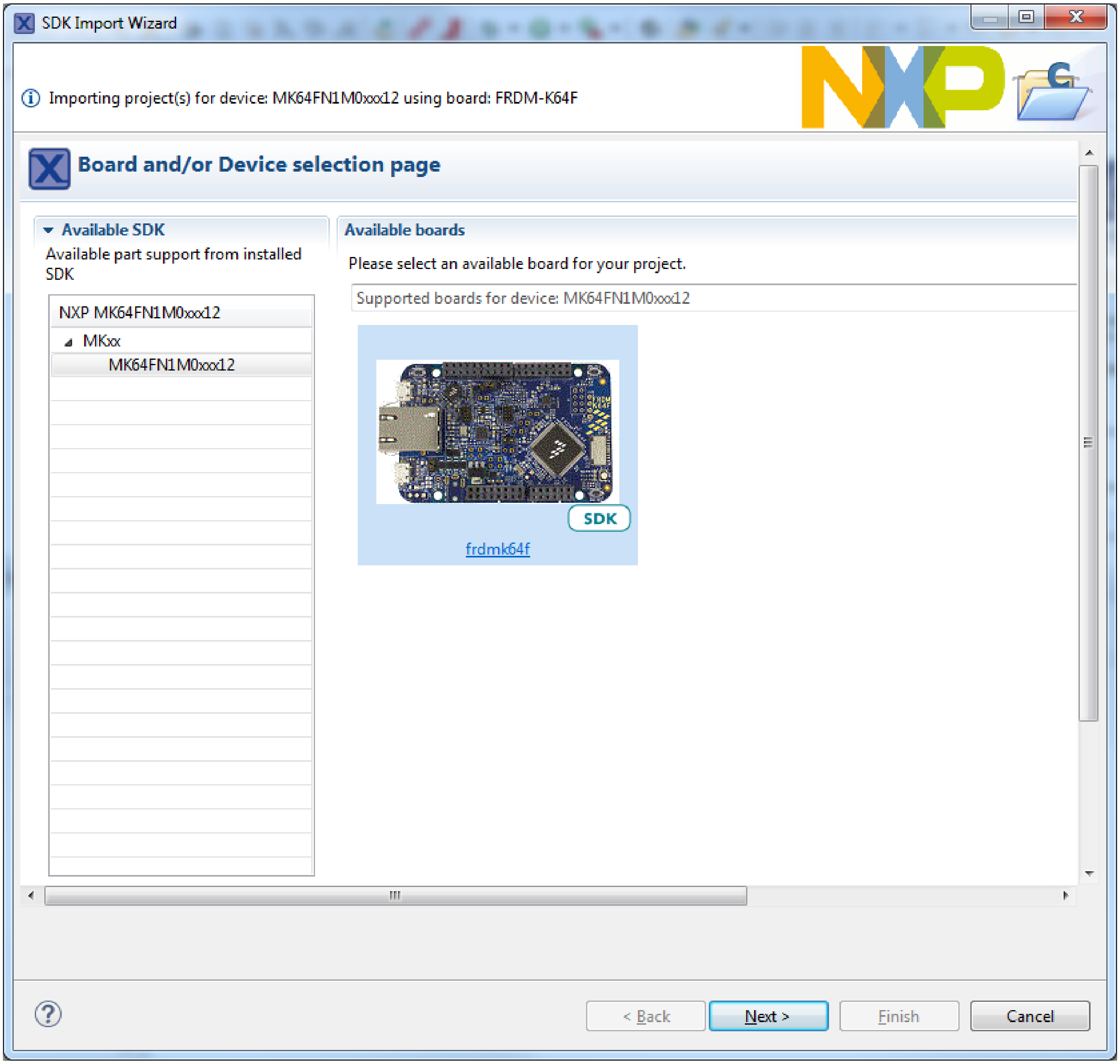


Figure 3. Import project



SDK Import Wizard

Importing project(s) for device: MK64FN1M0xxx12 using board: FRDM-K64F

Board and/or Device selection page

Available SDK

Available part support from installed SDK

NXP MK64FN1M0xxx12

MKxx

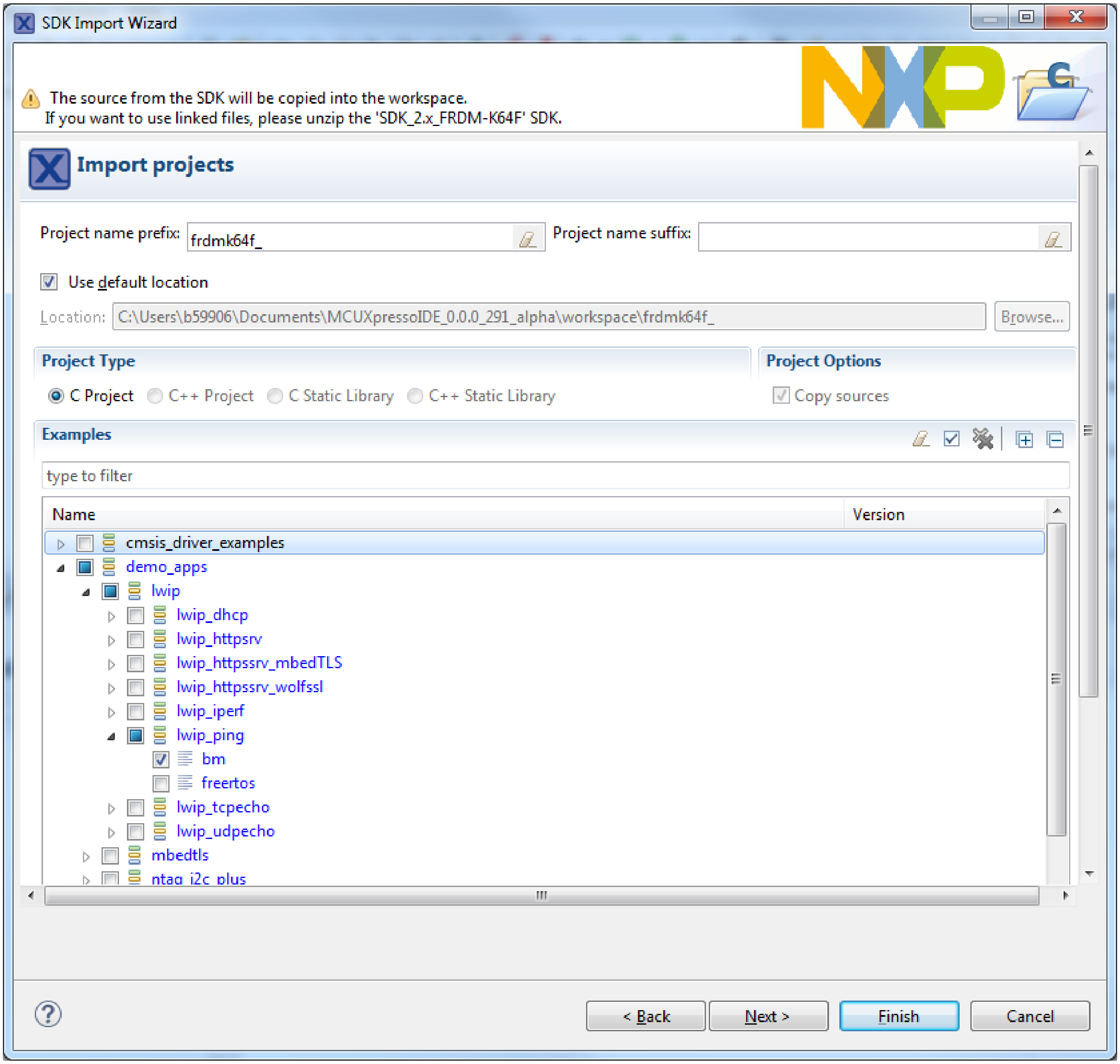
MK64FN1M0xxx12

Available boards

Please select an available board for your project.

Supported boards for device: MK64FN1M0xxx12

Figure 4. Select the board



The source from the SDK will be copied into the workspace.
If you want to use linked files, please unzip the 'SDK_2.x_FRDM-K64F' SDK.

Project name prefix: frdmk64f_

Project name suffix:

☑ Use default location

Location: C:\Users\b59906\Documents\MCUXpressoIDE_0.0.0_291_alpha\workspace\frdmk64f_

Project Type

● C Project ○ C++ Project ○ C Static Library ○ C++ Static Library

Project Options

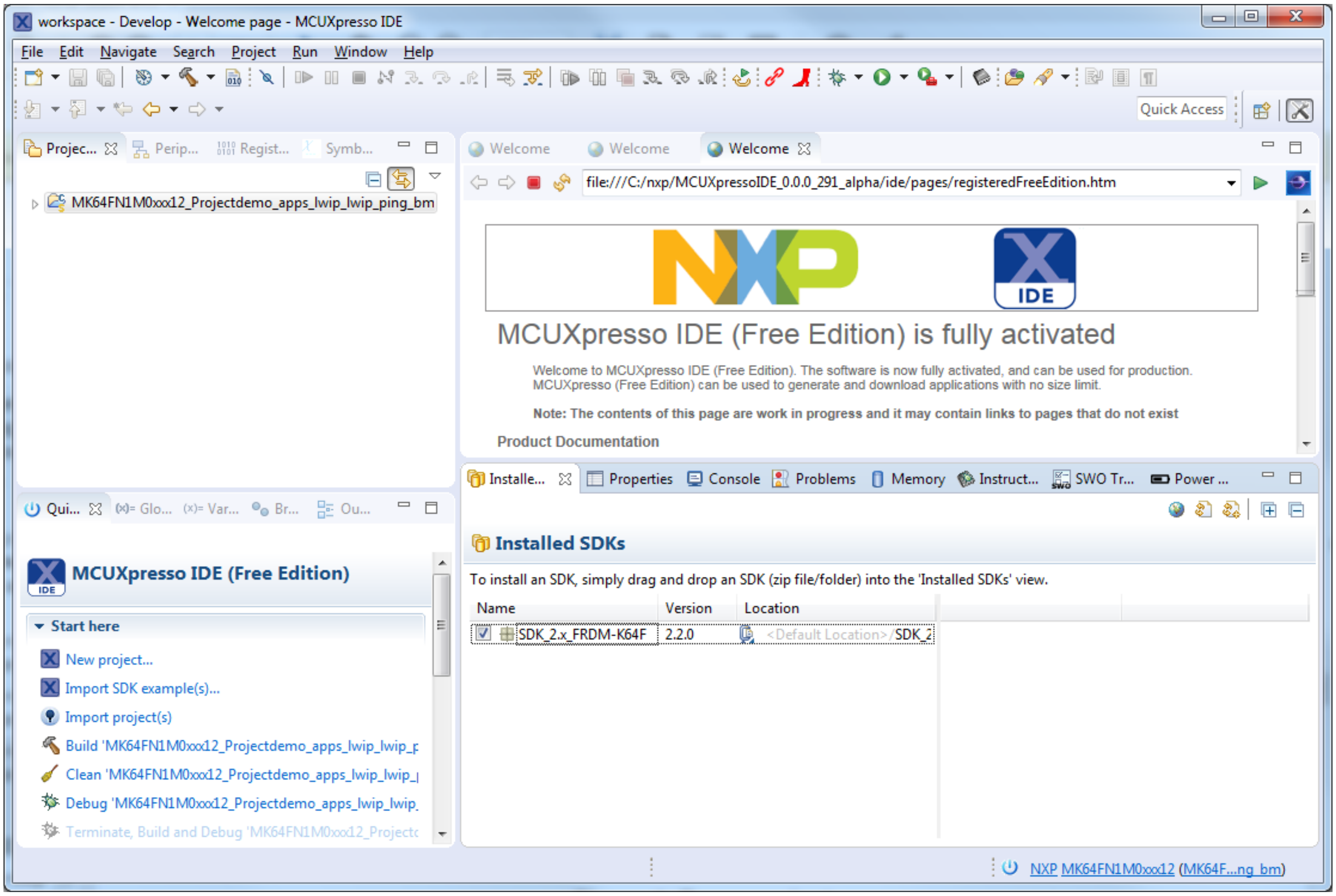
☑ Copy sources

Examples

type to filter

Name	Version
▷ □ cmsis_driver_examples	
◢ ■ demo_apps	
◢ ■ lwip	
▷ □ lwip_dhcp	
▷ □ lwip_httpsrv	
▷ □ lwip_httpssrv_mbedTLS	
▷ □ lwip_httpssrv_wolfssl	
▷ □ lwip_iperf	
◢ ■ lwip_ping	
☑ bm	
□ freertos	
▷ □ lwip_tcpecho	
▷ □ lwip_udpecho	
▷ □ mbedtls	
▷ □ ntag_i2c_plus	

Figure 5. Select the path of lwip_ping_bm


workspace - Develop - Welcome page - MCUXpresso IDE
File Edit Navigate Search Project Run Window Help
Quick Access
Projec... Perip... Regist... Symb...
MK64FN1M0xxx12_Projectdemo_apps_lwip_lwip_ping_bm
Welcome Welcome Welcome
file:///C:/nxp/MCUXpressoIDE_0.0.0_291_alpha/ide/pages/registeredFreeEdition.htm
NXP
IDE
Qui... Glo... Var... Br... Ou...
MCUXpresso IDE (Free Edition)
Start here
New project...
Import SDK example(s)...
Import project(s)
Build 'MK64FN1M0xxx12_Projectdemo_apps_lwip_lwip_p
Clean 'MK64FN1M0xxx12_Projectdemo_apps_lwip_lwip_
Debug 'MK64FN1M0xxx12_Projectdemo_apps_lwip_lwip_
Terminate, Build and Debug 'MK64FN1M0xxx12_Projectc
Installe... Properties Console Problems Memory Instruct... SWO Tr... Power ...
NXP MK64FN1M0xxx12 (MK64F...ng_bm)

MCUXpresso IDE (Free Edition) is fully activated

Welcome to MCUXpresso IDE (Free Edition). The software is now fully activated, and can be used for production. MCUXpresso (Free Edition) can be used to generate and download applications with no size limit.

Note: The contents of this page are work in progress and it may contain links to pages that do not exist

Product Documentation

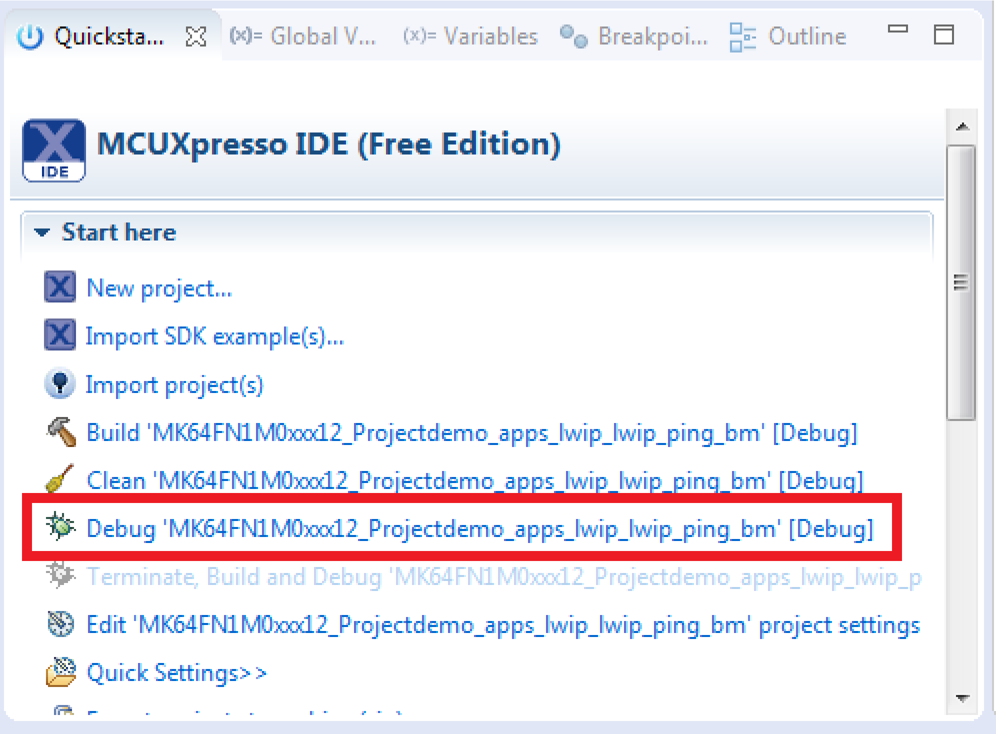
Installed SDKs

To install an SDK, simply drag and drop an SDK (zip file/folder) into the 'Installed SDKs' view.

Name	Version	Location
☑ SDK_2.x_FRDM-K64F	2.2.0	<Default Location>/SDK_2

Figure 6. Demo project

2. Build the lwip_ping_bm using [Build] on the Quickstart panel.
3. Select “Debug ‘MK64FN1M0xxx12_Projectdemo_apps_lwip_ping_bm’ [Debug]” on the Quickstart Panel. Wait for the download to finish.



Quicksta... | Global V... | Variables | Breakpoi... | Outline

MCUXpresso IDE (Free Edition)

Start here

- New project...
- Import SDK example(s)...
- Import project(s)
- Build 'MK64FN1M0xxx12_Projectdemo_apps_lwip_lwip_ping_bm' [Debug]
- Clean 'MK64FN1M0xxx12_Projectdemo_apps_lwip_lwip_ping_bm' [Debug]
- Debug 'MK64FN1M0xxx12_Projectdemo_apps_lwip_lwip_ping_bm' [Debug]
- Terminate, Build and Debug 'MK64FN1M0xxx12_Projectdemo_apps_lwip_lwip_p
- Edit 'MK64FN1M0xxx12_Projectdemo_apps_lwip_lwip_ping_bm' project settings
- Quick Settings>>

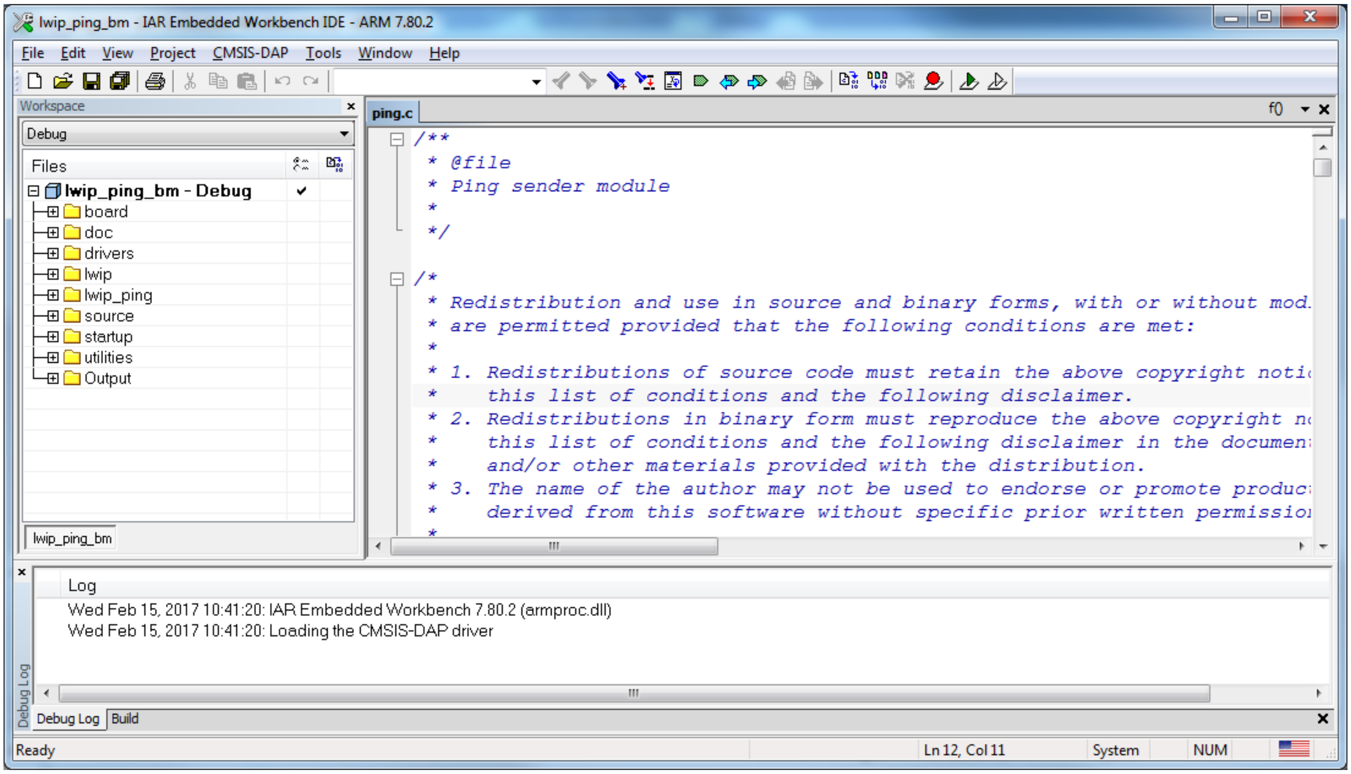
Figure 7. Debug the demo

4. Click the “Resume” button to run the demo.

5.2 Step-by-step guide for IAR Embedded Workbench for Arm

This section shows how to compile and run demos in IAR Embedded Workbench for Arm.

1. Open the workspace corresponding to different demos and different boards. For example, the lwip_ping_bm.eww on the Freedom FRDM-K64F Platform under <install_dir>/boards/frdmk64f/demo_apps/lwip/lwip_ping/bm/iar/ or the lwip_ping_demo_freertos.eww on the Freedom FRDM-K64F platform under <install_dir>/boards/frdmk64f/demo_apps/lwip/lwip_ping/freertos/iar/. These steps use lwip_ping_demo.eww on FRDM-K64F as an example.



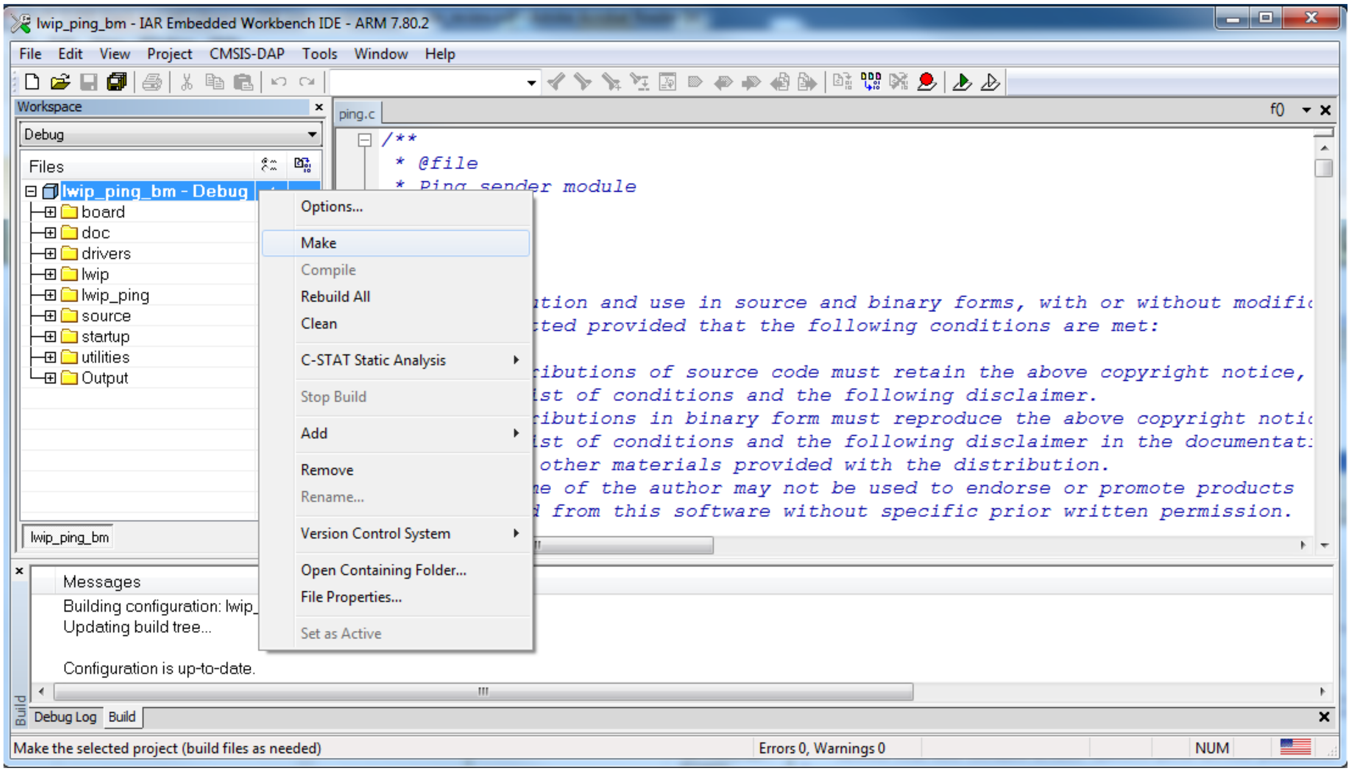
```
/**
 * @file
 * Ping sender module
 *
 */

/*
 * Redistribution and use in source and binary forms, with or without mod
 * are permitted provided that the following conditions are met:
 *
 * 1. Redistributions of source code must retain the above copyright notic
 *    this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright no
 *    this list of conditions and the following disclaimer in the documen
 *    and/or other materials provided with the distribution.
 * 3. The name of the author may not be used to endorse or promote produc
 *    derived from this software without specific prior written permissio
 *
```

```
Wed Feb 15, 2017 10:41:20: IAR Embedded Workbench 7.80.2 (armproc.dll)
Wed Feb 15, 2017 10:41:20: Loading the CMSIS-DAP driver
```

Figure 8. Workspace

2. Build the lwip_ping_bm.



```
/**
 * @file
 * Ping sender module
tion and use in source and binary forms, with or without modific
ted provided that the following conditions are met:

ibutions of source code must retain the above copyright notice,
ist of conditions and the following disclaimer.
ibutions in binary form must reproduce the above copyright notic
ist of conditions and the following disclaimer in the documentat
other materials provided with the distribution.
e of the author may not be used to endorse or promote products
 from this software without specific prior written permission.
```

```
Building configuration: lwip_
Updating build tree...

Configuration is up-to-date.
```

Make the selected project (build files as needed)

Errors 0, Warnings 0

Figure 9. lwip_ping_bm

3. Click the “Download and Debug” button. Wait for the download to finish.
4. Click the “Go” button to run the demo.

5.3 Step-by-step guide for Keil µVision 5 IDE

This section shows how to compile and run demos in Keil µVision 5 IDE.

1. Open the workspace corresponding to different demos and different boards. For example, the lwip_ping_bm.uvmpw on the Freedom FRDM-K64F platform under <install_dir>/boards/frdmk64f/demo_apps/lwip/lwip_ping/bm/mdk/ or the lwip_ping_freertos.uvmpw on the Freedom FRDM-K64F platform under <install_dir>/boards/frdmk64f/demo_apps/lwip/lwip_ping/freertos/mdk/. These steps take lwip_ping_bm.uvmpw on the Freedom FRDM-K64F platform for an example.

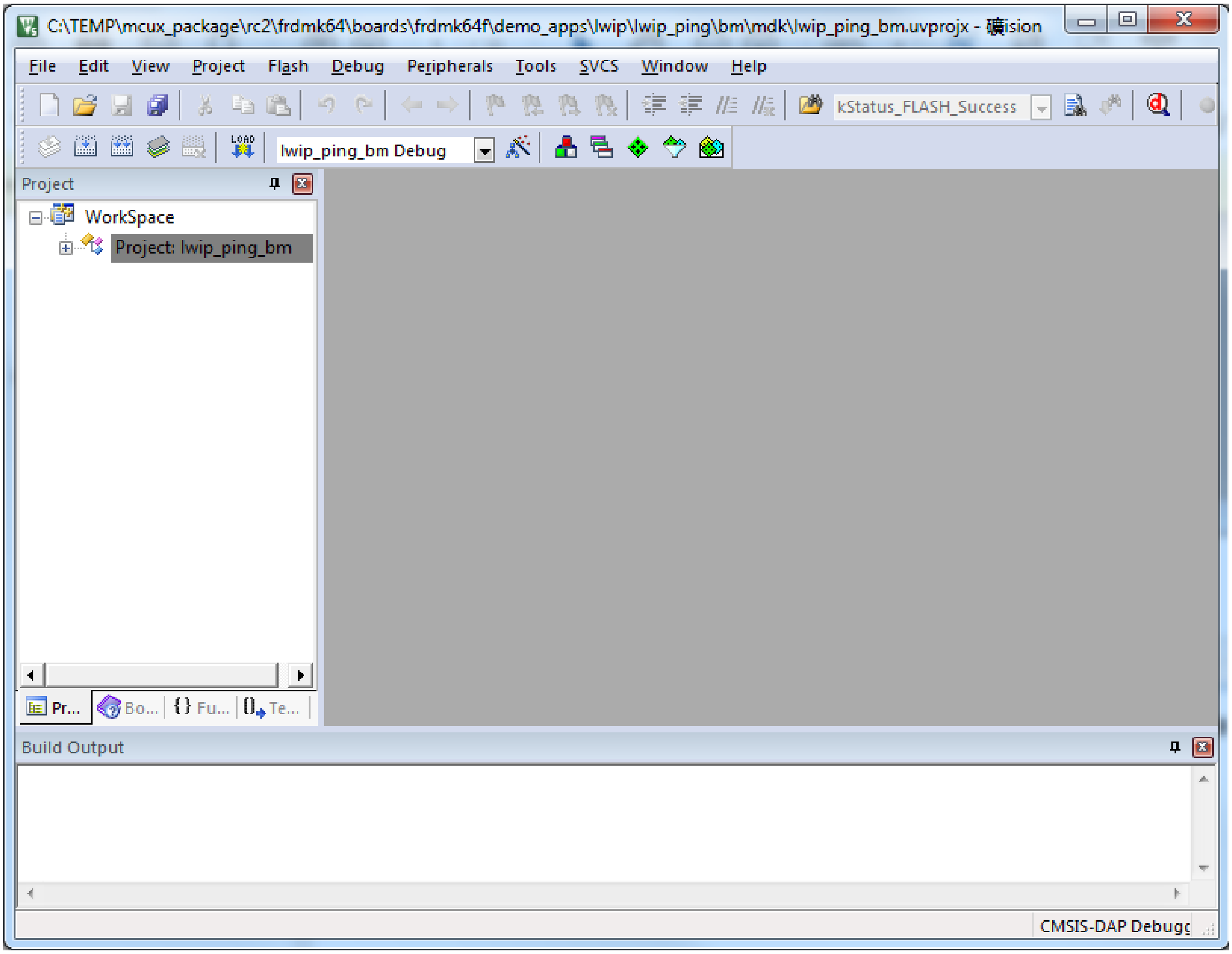


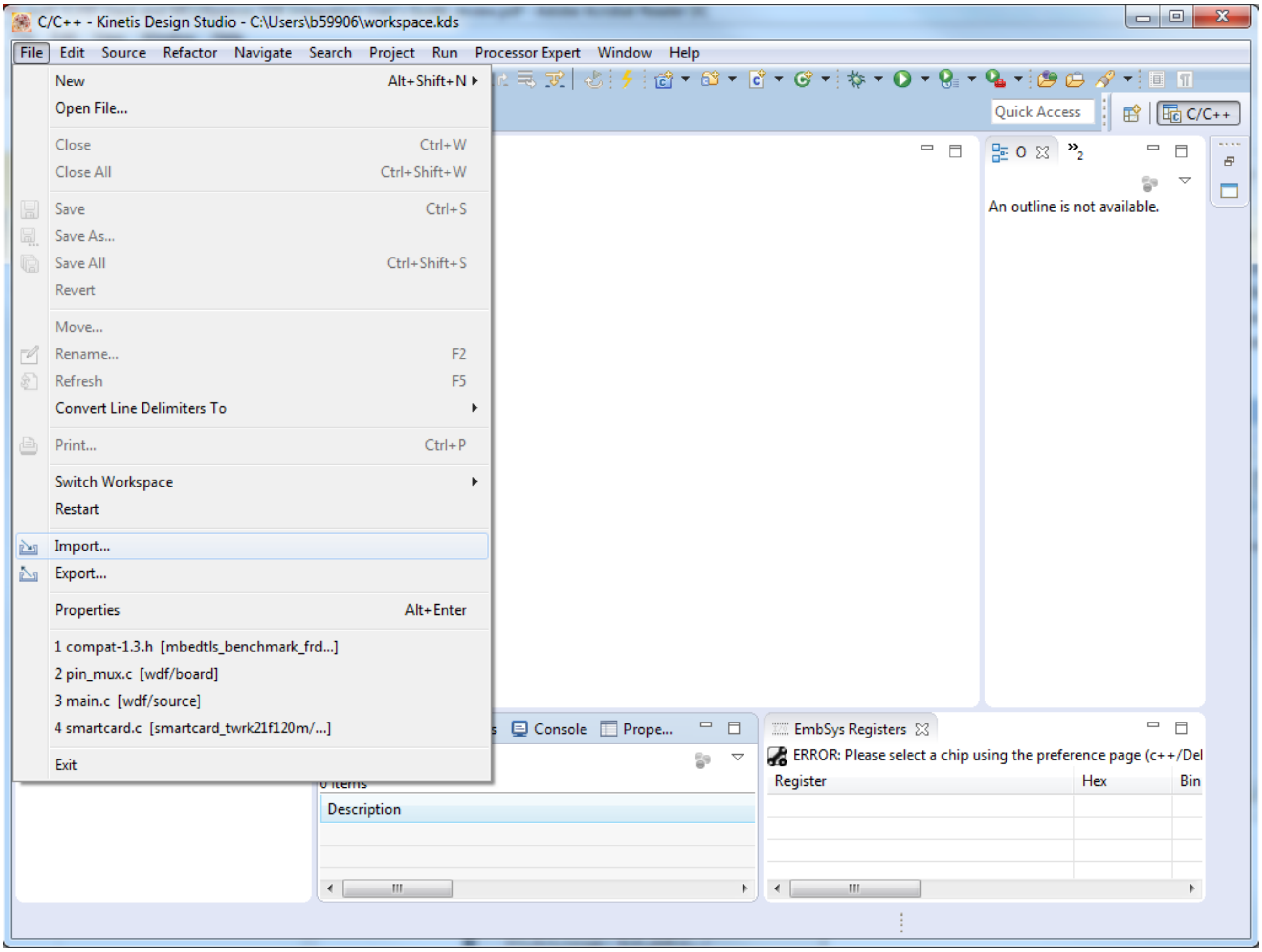
Figure 10. Workspace

2. Build the lwip_ping_bm.
3. Click Start/Stop Debug Session. Wait for the download to finish.
4. Click the “Run” button to run the demo.

5.4 Step-by-step guide for the Kinetis Design Studio IDE

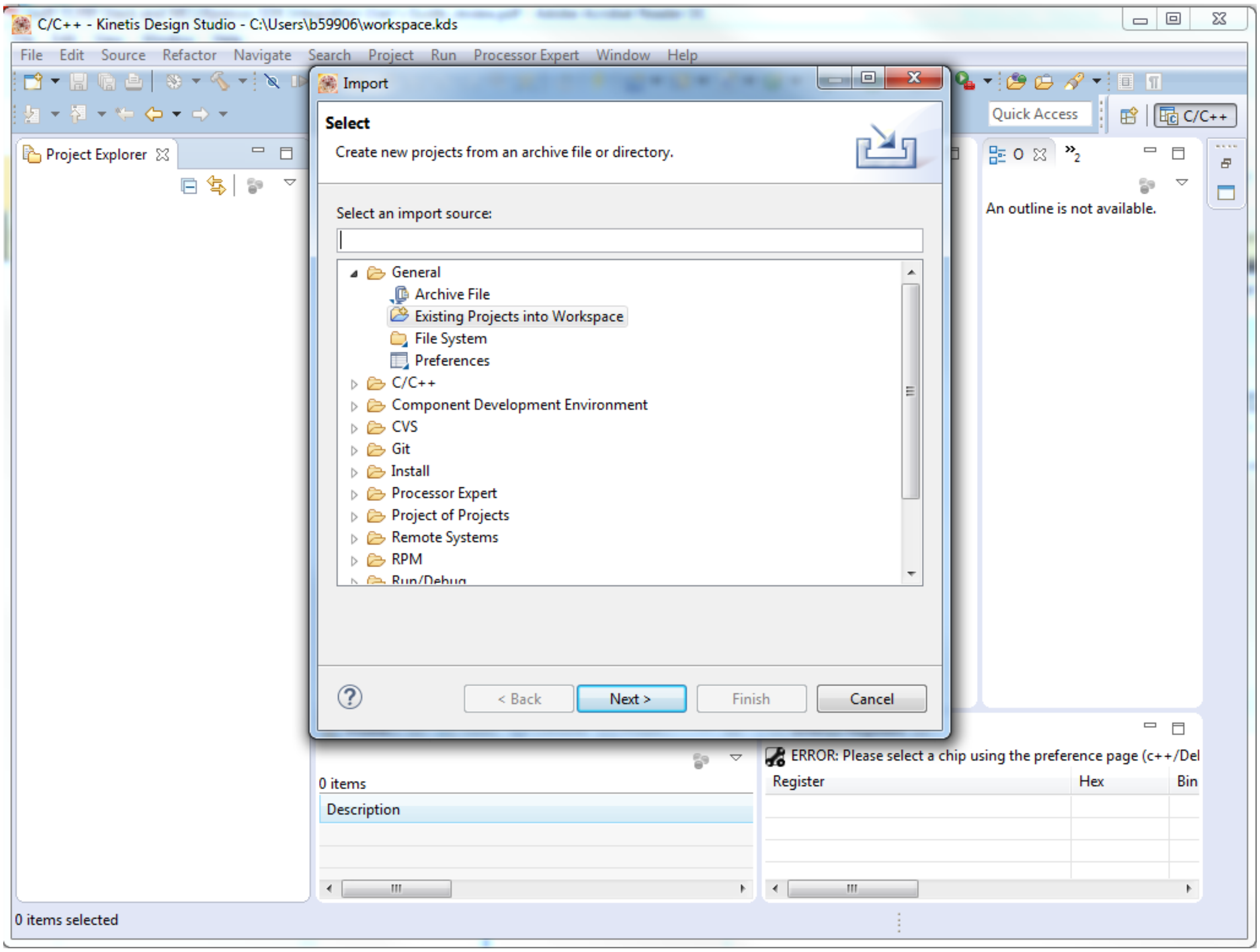
This section shows how to compile and run demos in the Kinetis Design Studio IDE.

1. The Kinetis Design Studio IDE does not have a workspace. Create a workspace and import the lwIP demos. For example, import the .cproject for lwip_ping_bm on Freedom FRDM-K64F platform under <install_dir>/boards/frdmk64f/demo_apps/lwip/lwip_ping/bm/kds/; or import the lwip_ping_freertos on the Freedom FRDM-K64F platform under <install_dir>/boards/frdmk64f/demo_apps/lwip/lwip_ping/freertos/kds/.



ERROR: Please select a chip using the preference page (c++/Del

Figure 11. Import project with KDS IDE



C/C++ - Kinetis Design Studio - C:\Users\b59906\workspace.kds

Import

Select

Create new projects from an archive file or directory.

Select an import source:

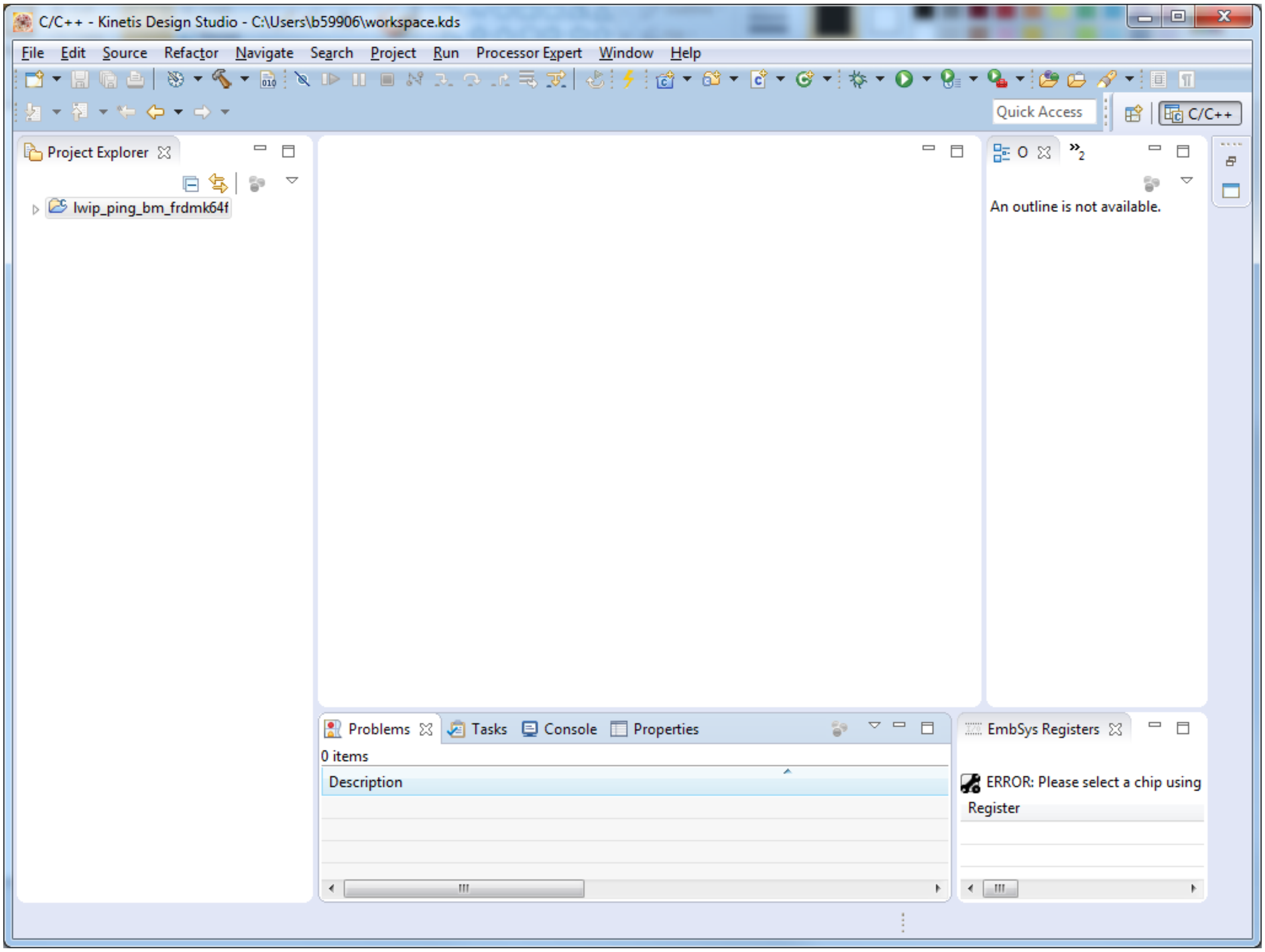
An outline is not available.

ERROR: Please select a chip using the preference page (c++/Del

0 items

0 items selected

Figure 12. Import project select



An outline is not available.

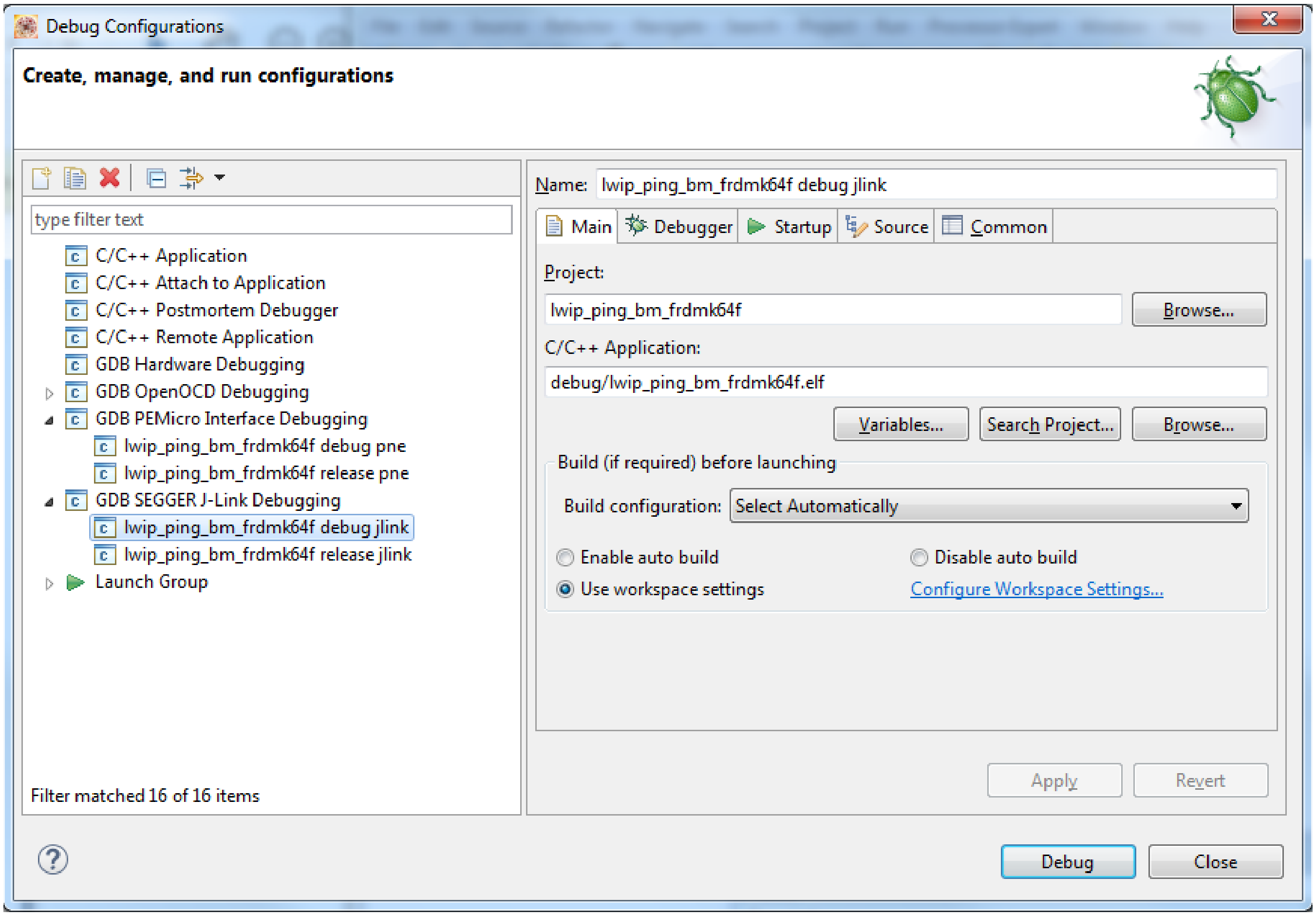
0 items

Description

ERROR: Please select a chip using Register

Figure 13. Demo project

2. Build the lwip_ping_bm.
3. Open debug configurations and choose J-Link Debugging.



Debug Configurations

Create, manage, and run configurations

Name: lwip_ping_bm_frdmk64f debug jlink

Project: lwip_ping_bm_frdmk64f

C/C++ Application: debug/lwip_ping_bm_frdmk64f.elf

Build (if required) before launching

Build configuration: Select Automatically

○ Enable auto build ○ Disable auto build

● Use workspace settings

Filter matched 16 of 16 items

type filter text
C/C++ Application
C/C++ Attach to Application
C/C++ Postmortem Debugger
C/C++ Remote Application
GDB Hardware Debugging
GDB OpenOCD Debugging
GDB PEMicro Interface Debugging
lwip_ping_bm_frdmk64f debug pne
lwip_ping_bm_frdmk64f release pne
GDB SEGGER J-Link Debugging
lwip_ping_bm_frdmk64f debug jlink
lwip_ping_bm_frdmk64f release jlink
Launch Group
Main
Debugger
Startup
Source
Common
Browse...
Variables...
Search Project...
Browse...
Configure Workspace Settings...
Apply
Revert
Debug
Close

Figure 14. Debug Configurations

4. Click the “Debug” button. Wait for the download to finish.
5. Click the “Resume” button to run the demo.

5.5 Step-by-step guide for ARMGCC and KDSGCC

1. ARMGCC and KDSGCC both use cMake to generate makefiles. Run the batch file (in the Windows® operating system) or sh file (in Linux® operating system) to build projects. These steps use ARMGCC as an example.
2. Change to the demo directory. For example: <install_dir>/boards/frdmk64f/demo_apps/lwip/lwip_ping/bm/armgcc
3. Run build_all.bat to build both debug and release projects.
4. Go to the debug/release directory to download and run the elf file using gdb.

6 Revision history

This table summarizes revisions to this document.

Table 1. Revision History

Revision number	Date	Substantive changes
2	09/2015	Updated tool versions in Section 3.2 and updated Section 5.2
3	11/2015	Updated Section 1
4	01/2016	Updated Section 1 and Section 3.2
5	10/2016	Updated Section 2.1 and Section 3.1
6	03/2017	Updated for MCUXpresso SDK
7	11/2017	• Removed unsupported board for MCUXpresso SDK v2.3.0 release • Added EVK-MIMXRT1050 board support
8	03/2018	Editorial updates for MCUXpresso SDK v2.3.1 and v2.4.0

How to Reach Us:

Home Page:
nxp.com

Web Support:
nxp.com/support

Information in this document is provided solely to enable system and software implementers to use NXP 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. NXP reserves the right to make changes without further notice to any products herein.

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP 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 NXP 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. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address: nxp.com/SalesTermsandConditions.

Document Number MCUXSDKLWIPUG
Revision 8, 03/2018