Freescale Semiconductor
User's Guide

Document Number: KBTLDRDEMOUG
Rev. 0, 07/2015

Kinetis Bootloader Demo Application User's Guide

Contents

1 Introduction

This document describes how to use the Kinetis bootloader to load a user application on a Kinetis MCU.

2 Overview

This guide describes the steps required to use the Freescale-provided Kinetis bootloader utilities to both load the Kinetis bootloader image and use the bootloader to update the user application section of flash. Upon reset, the bootloader detects the presence of the user application and launches it. The bootloader also provides a means to suppress the application launch and remain in the bootloader command processor in order to refresh the user application. This full-circle environment enables application developers to easily install new applications onto Kinetis devices, and provides manufacturers a way to update Kinetis devices in the field without the need for a debugger.

2.1 Kinetis bootloader

The Kinetis bootloader serves as the standard bootloader for all Kinetis devices. It provides a standard interface to the device via all of the available peripherals supported on a given Freescale Kinetis device. The Kinetis bootloader interface comes in several forms, ranging from ROM, serial flashloader, or a customized flash-resident bootloader. Some future Kinetis devices arrive with a ROM containing the Kinetis bootloader, while others arrive pre-programmed from the factory with a one-time-use serial flashloader. For a customized interface, customers can leverage the Kinetis bootloader source code to create a unique flash-resident bootloader that is both compatible with tools that understand the bootloader interface, and are capable of supporting application-specific features. Freescale provides utilities to demonstrate how to interface with the bootloader.

2.2 Host utility

The blhost.exe utility is a cross-platform host program used to interface with devices running the Kinetis bootloader. It can list and request execution of all of the commands supported by a given Kinetis device running the bootloader. For more information on the blhost utility, see the *Kinetis blhost User's Guide* (document KBLHOSTUG).

2.3 led_demo user application

The `led_demo_<platform>_<base_address>` binaries are example demo firmware applications used to demonstrate how the Kinetis bootloader can load and launch user applications. The demo binaries are found in `<install_dir>/apps/led_demo/binaries.`

2.4 Host updater

The KinetisUpdater.exe host application is a Windows® OS GUI program used to update the user application image on the device running the Kinetis bootloader firmware application. For more information on the Kinetis Updater application, see the *Kinetis Updater User's Guide* (document KUPDTRUG).

NOTE

The Kinetis Updater application requires Windows OS 7 or higher. Windows XP users should use the blhost utility instead.

2.5 Toolchain requirement

Firmware projects:

- IAR Embedded Workbench for ARM® v7.40.2 or later
- Python v2.7 (www.python.org)
- Kinetis Design Studio IDE (KDS) v.3.0.0

Host projects:

- Microsoft® Visual Studio® Professional 2013 for Windows ® OS Desktop
- Microsoft® .NET Framework 4.5 (included in Windows OS 8)
- Microsoft® Visual C++ Redistributable for Visual Studio 2013 (vcredist_x86.exe)
- Apple® Xcode v6.3.1 (for the blhost tool)
- GNU Compiler (GCC) v4.8.1 (for the blhost tool)

3 Kinetis bootloader application

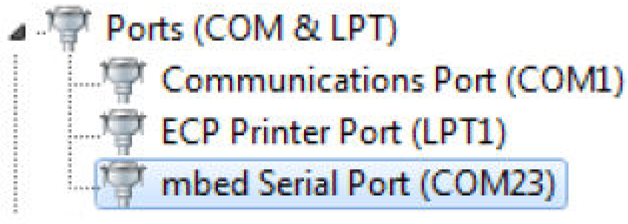
This section describes how to connect the platform to the computer and download the pre-built Kinetis bootloader application. For information about the configuration of a board, find the subsection dedicated to your specific board in Appendix B. All examples assume that the board is in its factory default configuration (jumpers, OpenSDA, etc).

3.1 Connect the platform

FRDM-K22F, FRDM-K64F, or TWR-K24F120M:

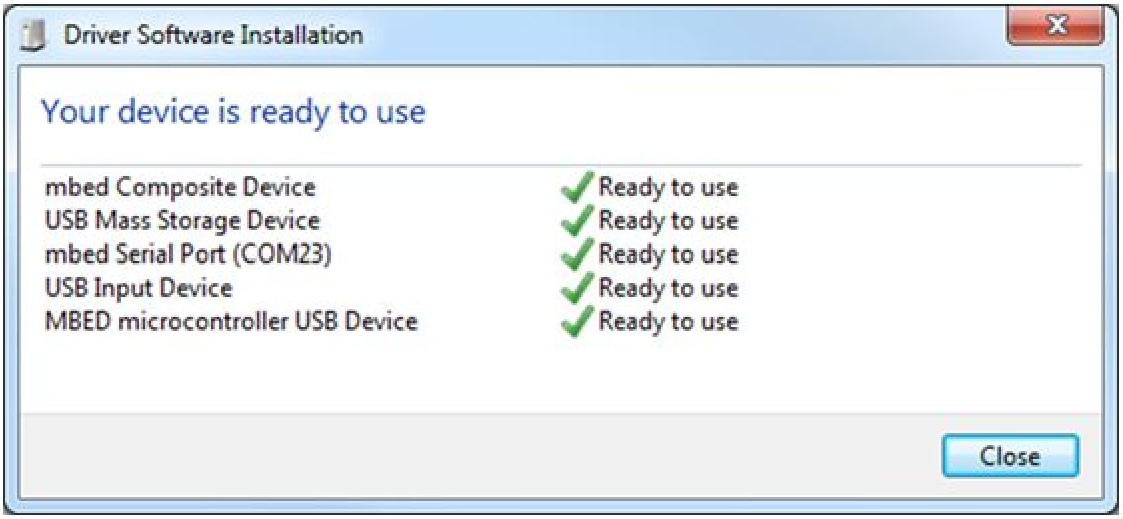
For Windows operating systems PCs, install the ARM® mbed™ serial port driver in order to communicate with the Kinetis device over a serial port.

1. Download and install the latest mbed Windows OS serial port driver from http://developer.mbed.org/handbook/Windows-serial-configuration.
2. Connect the OpenSDA USB connector to the USB port on a PC.
3. Install the mbed serial port driver.



Ports (COM & LPT)
- Communications Port (COM1)
- ECP Printer Port (LPT1)
- mbed Serial Port (COM23)

Figure 1. mbed serial port in Windows Device Manager



Driver Software Installation

Your device is ready to use

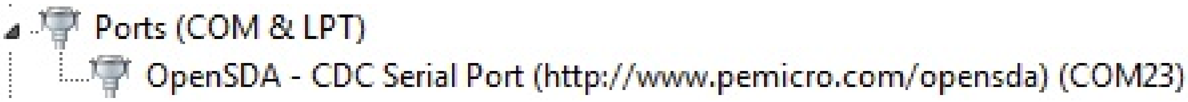
mbed Composite Device	✓ Ready to use
USB Mass Storage Device	✓ Ready to use
mbed Serial Port (COM23)	✓ Ready to use
USB Input Device	✓ Ready to use
MBED microcontroller USB Device	✓ Ready to use

Figure 2. Driver software installation

FRDM-KL03Z, FRDM-KL25Z, TWR-KV11Z75M, TWR-KV46F150M, TWR-KV31F120M, TWR-K22F120M, TWR-K64F120M, or TWR-K65F180M:

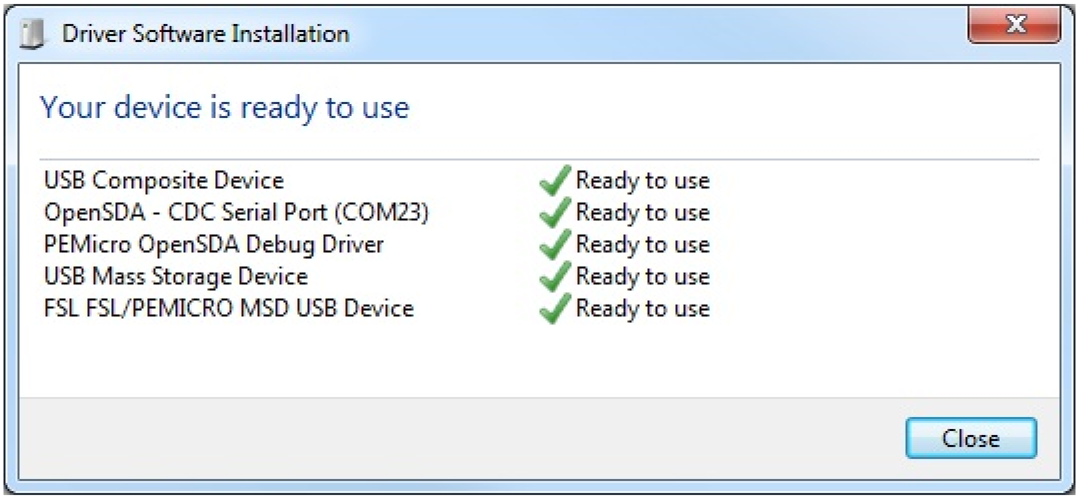
For PCs running on Windows OS, install the P&E Micro OpenSDA drivers in order to communicate with the Kinetis device over a serial port.

1. Connect the module to the USB port on a PC using the module's debug USB connector.
2. Download the driver package from the P&E Micro website (www.pemicro.com/opensda/) and run the installer.
3. After the installer is finished, plug in the module and open the Windows Device Manager to show the COM port number assigned to the virtual serial port.



Ports (COM & LPT)
OpenSDA - CDC Serial Port (http://www.pemicro.com/opensda) (COM23)

Figure 3. OpenSDA virtual comport in Windows Device Manager



Driver Software Installation

Your device is ready to use

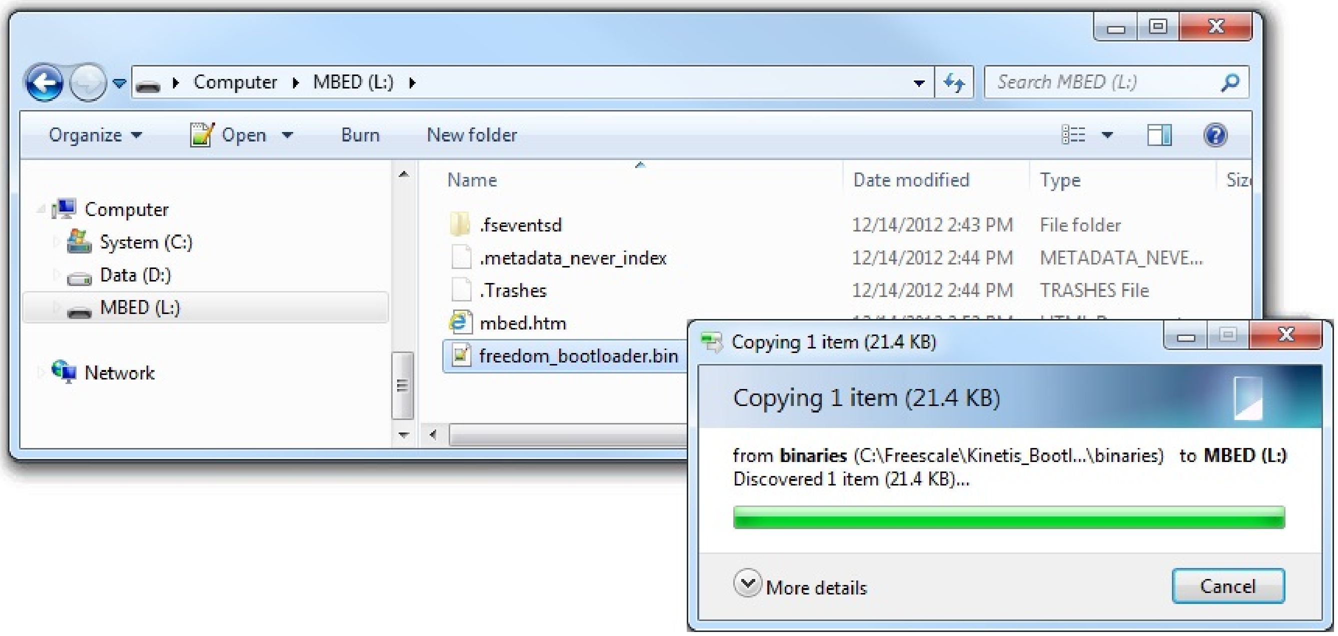
USB Composite Device	✓ Ready to use
OpenSDA - CDC Serial Port (COM23)	✓ Ready to use
PEMicro OpenSDA Debug Driver	✓ Ready to use
USB Mass Storage Device	✓ Ready to use
FSL FSL/PEMICRO MSD USB Device	✓ Ready to use

Figure 4. Driver software installation

3.1.1 Install the Kinetis bootloader

To install the bootloader application, drag or copy and paste the appropriate binary (Tower or Freedom) file from `<install_dir>/targets/<device>/binaries` onto the mass storage device, where <device> is the MCU family container folder. While it is possible to use the flashloader image (for boards that support it), peripheral pin mappings may not route to the Freedom or Tower ports that are easily accessible. For flashloader use, see Appendix C for pin mappings. The mass storage device appears on your computer as either a drive named "MBED" or the board name depending on the OpenSDA firmware loaded onto the board.

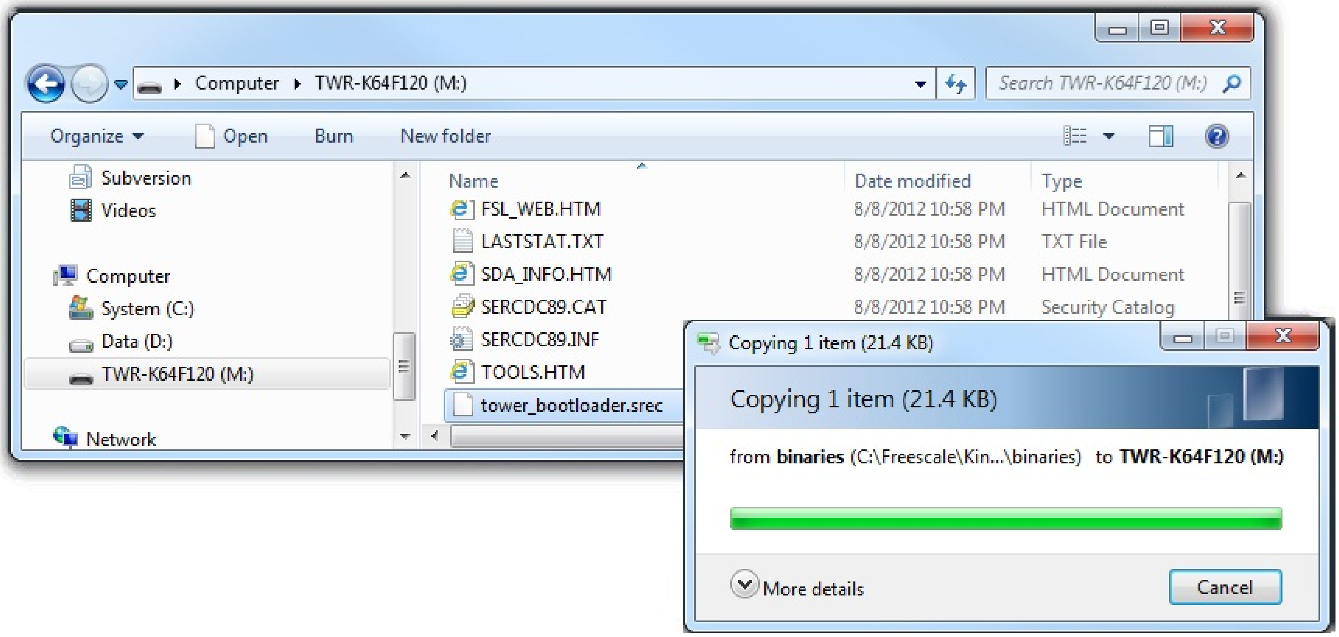
Choose the binary based on whether you have a Freescale Freedom platform or Tower System module. The bootloader srec format image should be used to download with openSDA and bootloader binary format image to download with mbed software.



Copying 1 item (21.4 KB)

from **binaries** (C:\Freescale\Kinetis_Bootl...\binaries) to **MBED (L:)**
Discovered 1 item (21.4 KB)...

Figure 5. Install the Kinetis bootloader on the MBED-labeled drive



Copying 1 item (21.4 KB)

from **binaries** (C:\Freescale\Kin...\binaries) to **TWR-K64F120 (M:)**

Figure 6. Install the Kinetis bootloader on the board-named drive

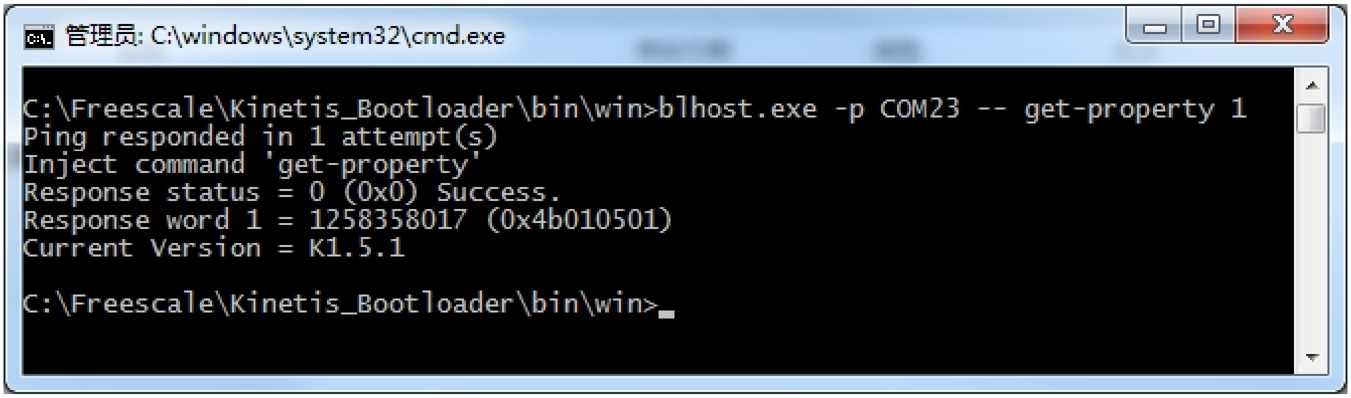
4 The host utility application

This section describes simple use of the blhost host utility program to demonstrate communication with the Kinetis bootloader.

- Open a command prompt in the directory containing blhost. For Windows OS, blhost is located in `<install_dir>/bin/win.` To open a command prompt, go to the Windows OS start menu and type "cmd" in the search box at the bottom of the window. Navigate to the blhost folder using change directory (CD) commands.
- Type *blhost --help* to see the complete usage of the blhost utility.

For this exercise, verify the Kinetis device is running the bootloader firmware application.

- Press the "Reset" button on the platform.
- Note what the COM port that the platform is connected to. Refer to step 3 of Section 3.1. For this guide, the device is connected to COM23.
- Type *blhost -p COM23 -- get-property 1* to get the bootloader version from the Kinetis bootloader.
- Something similar to the screen shot below indicates that blhost.exe is successfully communicating with the Kinetis bootloader on the platform.



```
C:\Freescale\Kinetis_Bootloader\bin\win>blhost.exe -p COM23 -- get-property 1
Ping responded in 1 attempt(s)
Inject command 'get-property'
Response status = 0 (0x0) Success.
Response word 1 = 1258358017 (0x4b010501)
Current Version = K1.5.1

C:\Freescale\Kinetis_Bootloader\bin\win>
```

Figure 7. Host communication with Kinetis bootloader

5 Windows GUI updater application

This section describes how to use the Windows GUI updater application, KinetisUpdater.exe, to install an example user application onto the platform.

5.1 Installing the user application

The FRDM-K22F platform is used in this example. Similar steps can be used for other development platforms.

1. Press the "Reset" button on the platform.
2. Navigate Windows Explorer to the `<install_dir>/bin/win/KinetisUpdater` directory.
3. Double-click the KinetisUpdater.exe file to launch the app.

NOTE

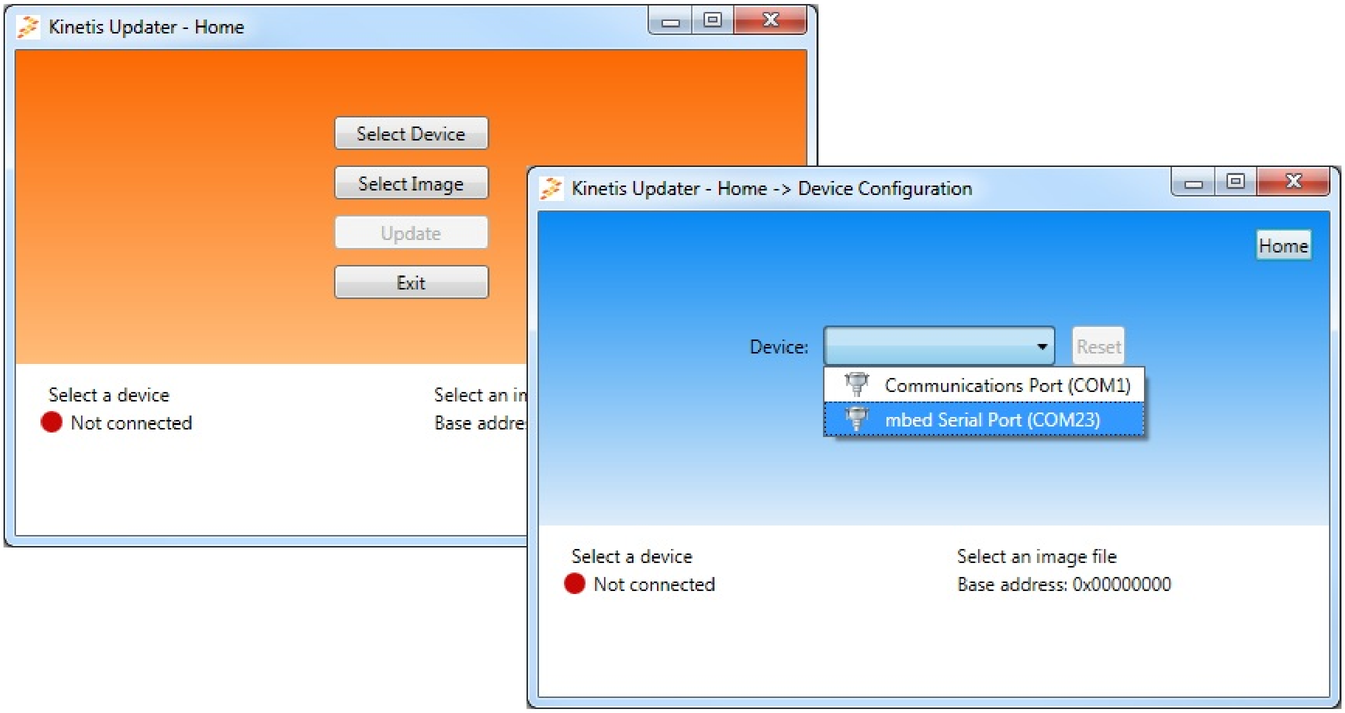
If the application fails to launch, check that the .NET Framework 4.5 and the Visual C++ Redistributable 2013 are installed as noted in Toolchain requirements.

NOTE

Ensure that the x86 version of the package is installed, not x64 version. This even applies for 64-bit systems.

4. Start at the orange home screen. Click "Select Device".
   - The blue device configuration page is displayed.

- Select the COM23 device from the drop-down box.
- Click "Home" to return to the home screen.

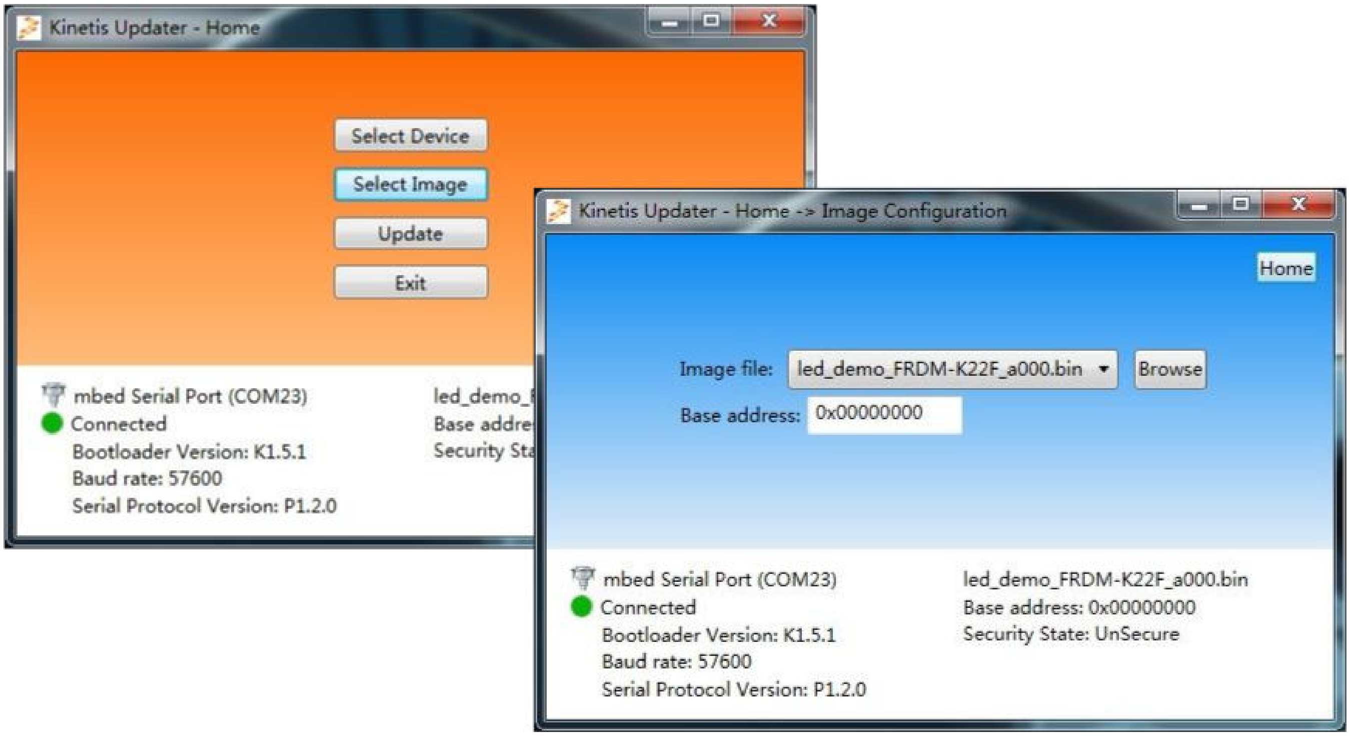


Select a device
● Not connected

Select an image file
Base address: 0x00000000

Figure 8. Select the COM port

5. Click "Select Image".
   - The blue image configuration page is displayed.
   - Select the led_demo_FRDM-K22F_a000.bin application image from the `<install_dir>/apps/led_demo/binaries` directory using the "Browse" button.
   - Set the base address to 0xA000.
   - Click "Home" to return to the home screen.

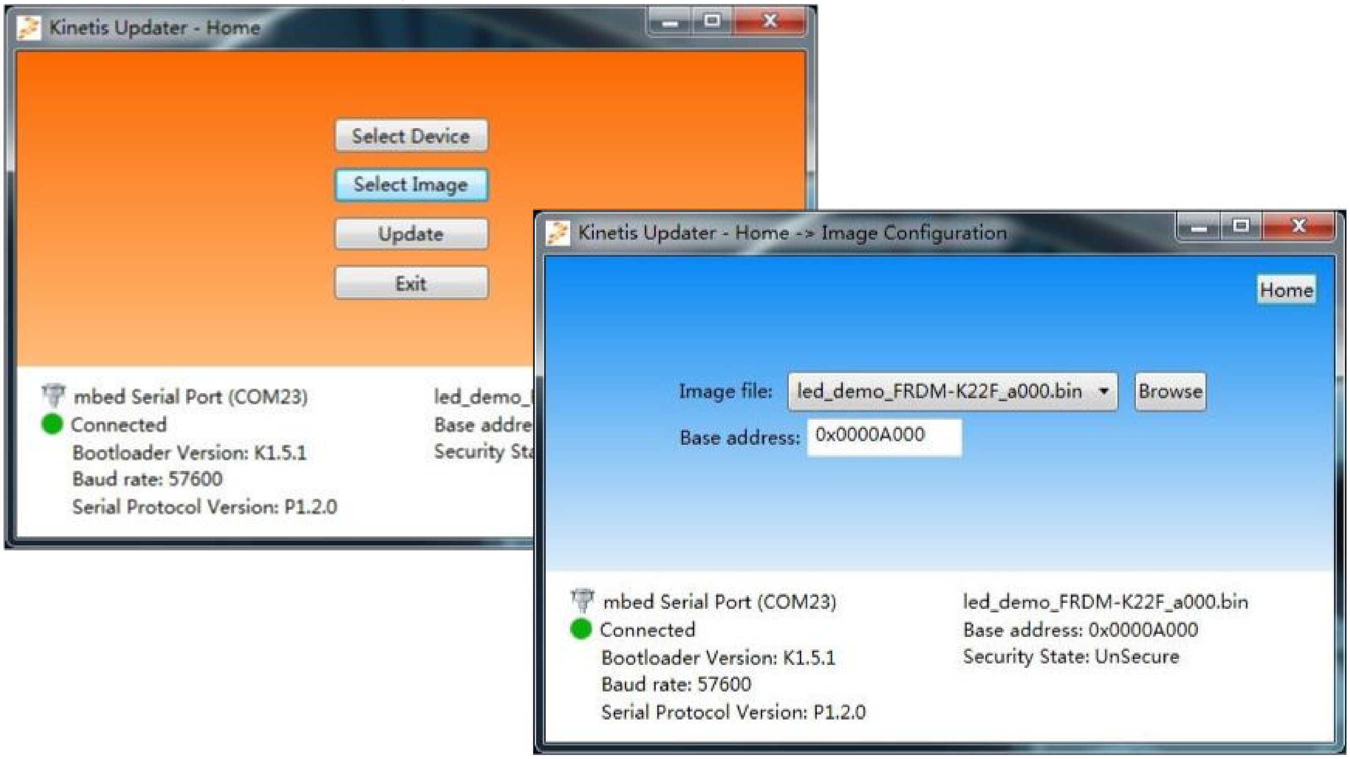


mbed Serial Port (COM23)
Connected
Bootloader Version: K1.5.1
Baud rate: 57600
Serial Protocol Version: P1.2.0

Image file: led_demo_FRDM-K22F_a000.bin
Base address: 0x00000000

led_demo_FRDM-K22F_a000.bin
Base address: 0x00000000
Security State: UnSecure

Figure 9. Browse for the user application



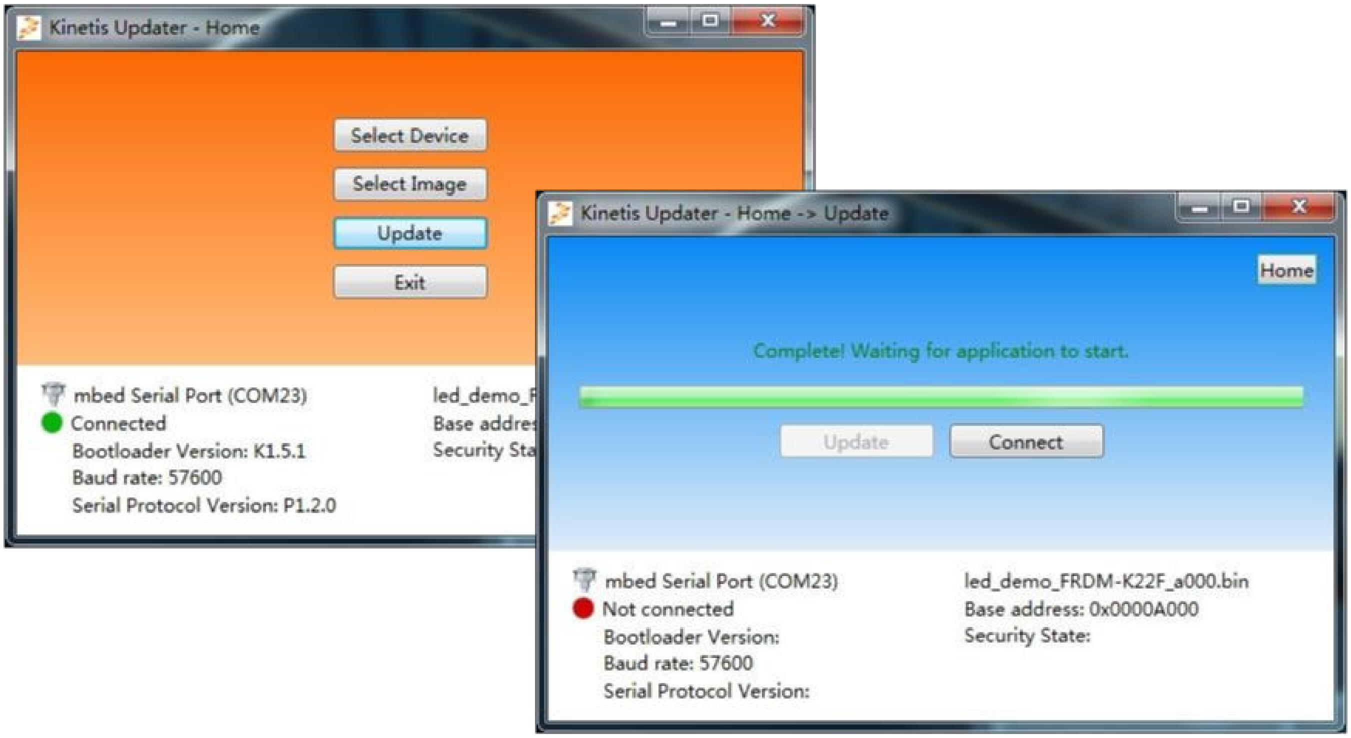
mbed Serial Port (COM23)
Connected
Bootloader Version: K1.5.1
Baud rate: 57600
Serial Protocol Version: P1.2.0

Image file: led_demo_FRDM-K22F_a000.bin
Base address: 0x0000A000

led_demo_FRDM-K22F_a000.bin
Base address: 0x0000A000
Security State: UnSecure

Figure 10. Set base address for application file

7. The "Update" button should now be enabled. Click "Update".
    - The blue update page is now displayed.
    - Click the "Update" button to write the application image to the device flash.
    - Wait for the application to start. The waiting time is determined by the timeout parameter.



mbed Serial Port (COM23)
● Connected
Bootloader Version: K1.5.1
Baud rate: 57600
Serial Protocol Version: P1.2.0

led_demo_F
Base addres
Security Sta

Complete! Waiting for application to start.

mbed Serial Port (COM23)
● Not connected
Bootloader Version:
Baud rate: 57600
Serial Protocol Version:

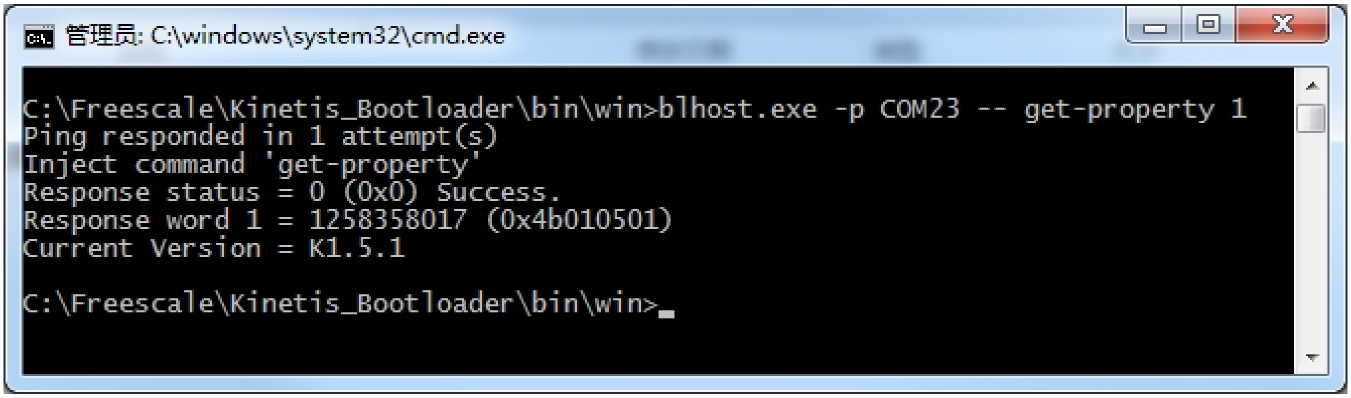
led_demo_FRDM-K22F_a000.bin
Base address: 0x0000A000
Security State:

Figure 11. Perform the update

8. At this point, the LED(s) on the target board should be noticably blinking indicating that the Kinetis Bootloader successfully installed the led_demo user application.
9. You can reprogram the device without exiting the application if you re-enter the bootloader by pressing the boot pin button (see Appendix B to determine if your platform has a boot pin button) and resetting the board.
10. Click the "Exit" button when finished.

6 Returning to Flash-resident bootloader

Some Freescale development platforms support re-entry of the bootloader from a user application. See Appendix B to determine if your board has a "Boot Pin" button listed. To return to the Kinetis bootloader interface, simply hold the "Boot Pin" button and press and release the "Reset" button on the target board. When the device resets, the Kinetis bootloader detects the press on the boot pin and does not jump to the user application. Verify you are in bootloader mode by again running the blhost.exe tool as done earlier.



```
C:\Freescale\Kinetis_Bootloader\bin\win>blhost.exe -p COM23 -- get-property 1
Ping responded in 1 attempt(s)
Inject command 'get-property'
Response status = 0 (0x0) Success.
Response word 1 = 1258358017 (0x4b010501)
Current Version = K1.5.1

C:\Freescale\Kinetis_Bootloader\bin\win>
```

Figure 12. Back to the Kinetis bootloader interface

Pressing the "Reset" button alone allows the Kinetis Bootloader to again launch the led_demo application.

7 Appendix A - Kinetis flash-resident bootloader operation

This section describes the linkage between the Kinetis flash-resident bootloader and the user application. The demonstration described above illustrates a fairly simple collaboration between the Kinetis bootloader and the led_demo application. The considerations are:

- The flash-resident bootloader is located in flash at address 0.
- The user application is located in flash above the bootloader at BL_APP_VECTOR_TABLE_ADDRESS as defined in `<install_dir>/apps/targets/<mcu>/src/bootloader_config.h`
- The vector table for the User Application must be placed at the beginning of the application image.
- The Bootloader Configuration Area (BCA) must be placed at 0x3C0 from the beginning of the image.

NOTE

The base address of a user application for use with a flash-resident bootloader is different than the application base address when using a ROM-based bootloader. The application linker file must be updated to link teh image to the correct base address. In addition, application vector table must be updated based on the correct application location.

7.1 Memory map overview

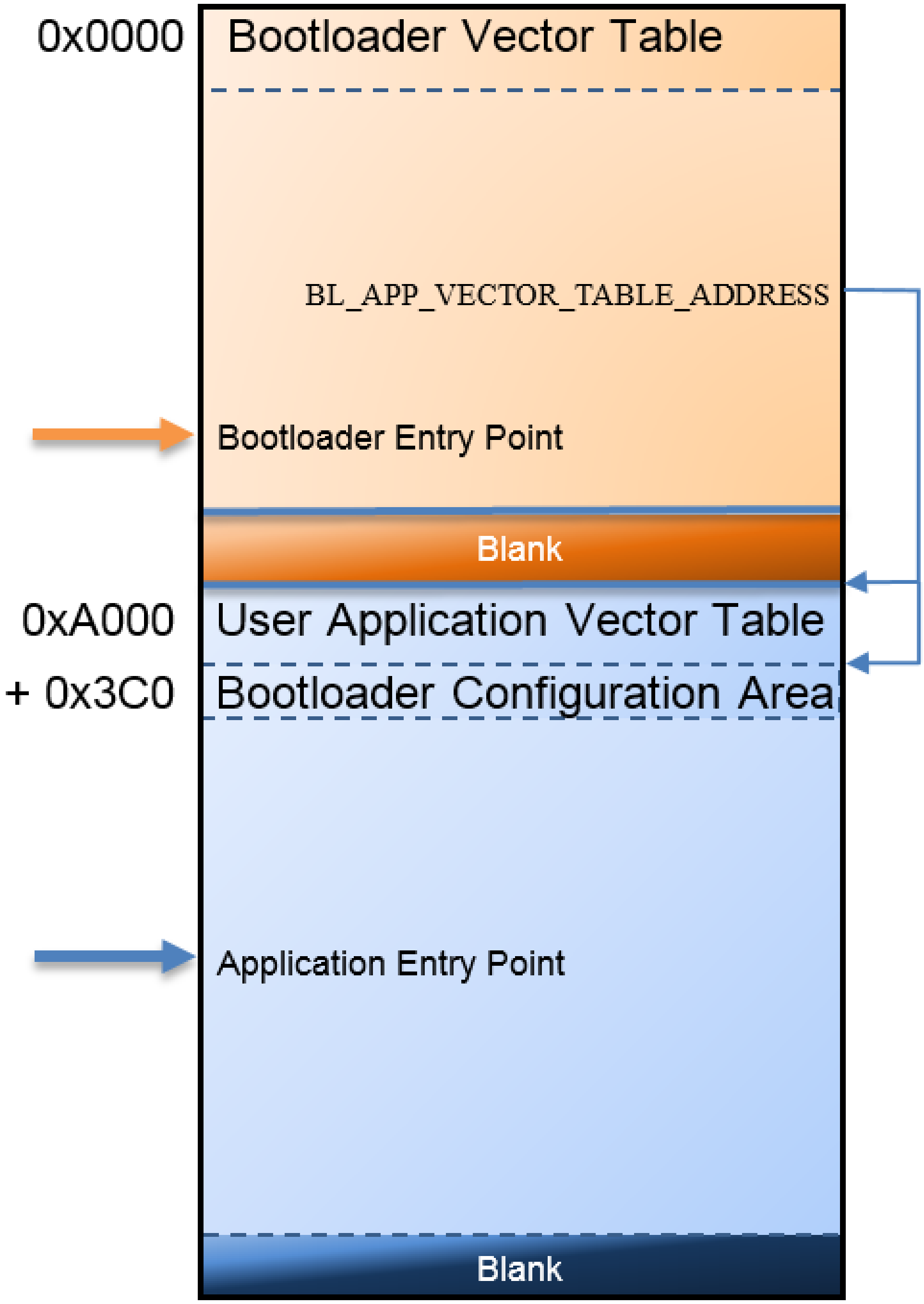


Figure 13. Device memory map

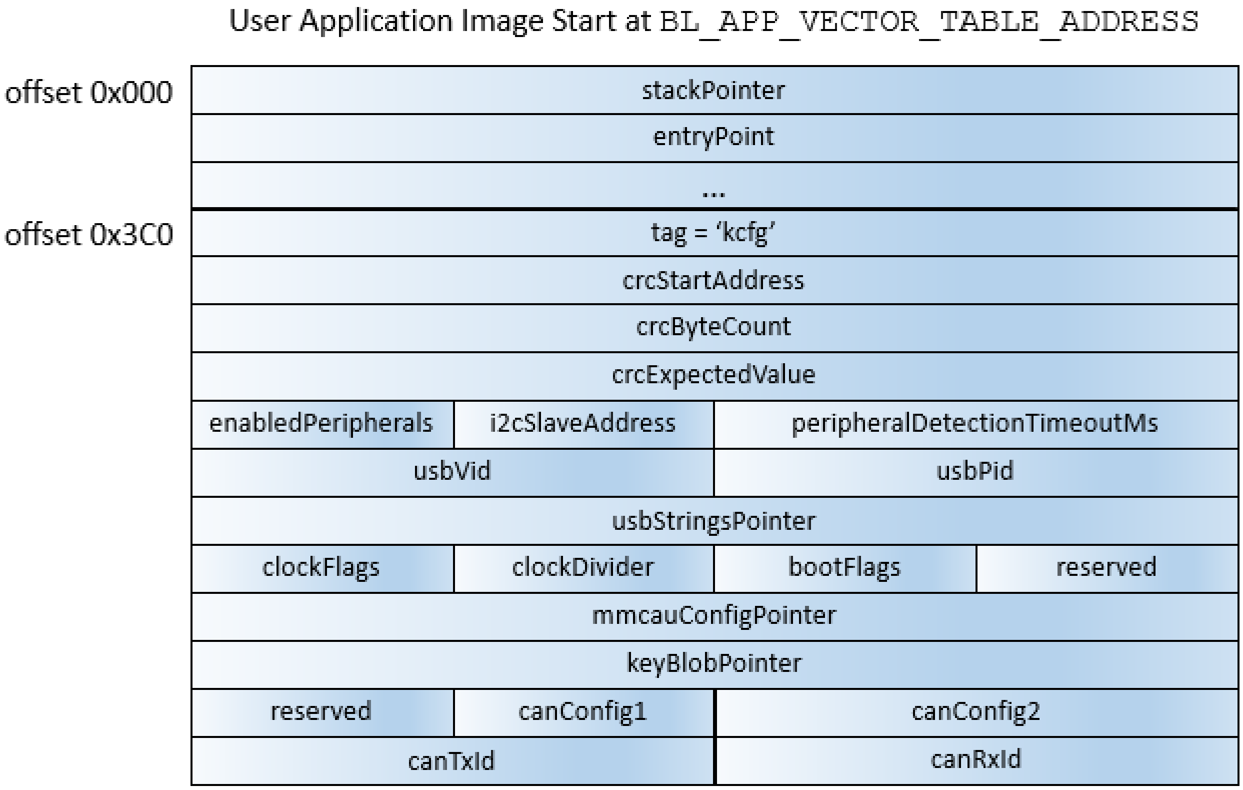


Figure 14. User application vector table and Bootloader Configuration Area (BCA)

7.2 User application vector table

The Kinetis bootloader checks `BL_APP_VECTOR_TABLE_ADDRESS+0` for the User Application stack pointer and `BL_APP_VECTOR_TABLE_ADDRESS+4` for the User Application entry point. Initially, this area is expected to be erased (0xFF) and the bootloader remains in its command interface.

After a user application is installed to `BL_APP_VECTOR_TABLE_ADDRESS`, the bootloader jumps to the application after a period specified by `peripheralDetectionTimeoutMs` in the Bootloader Configuration Area (BCA).

7.3 Bootloader Configuration Area (BCA)

The Bootloader Configuration Area is located at offset 0x3C0 from the beginning of the User Application image. This information is read by the Kinetis bootloader early during the bootloader initialization in order to set up clocks and gather other information relevant to detecting active peripherals. If the first four bytes of the BCA are not `'kcfg'`, the bootloader does not use any information from the BCA on flash.

For this tutorial, the led_demo application set the tag field to `'kcfg'` and the `peripheralDetectionTimeoutMs` to 500 so that the bootloader would only wait 500 milliseconds before launching the led_demo application instead of the default 5 second wait period.

8 Appendix B - Kinetis Bootloader Development platforms

All boards must be in their default factory state for jumper settings.

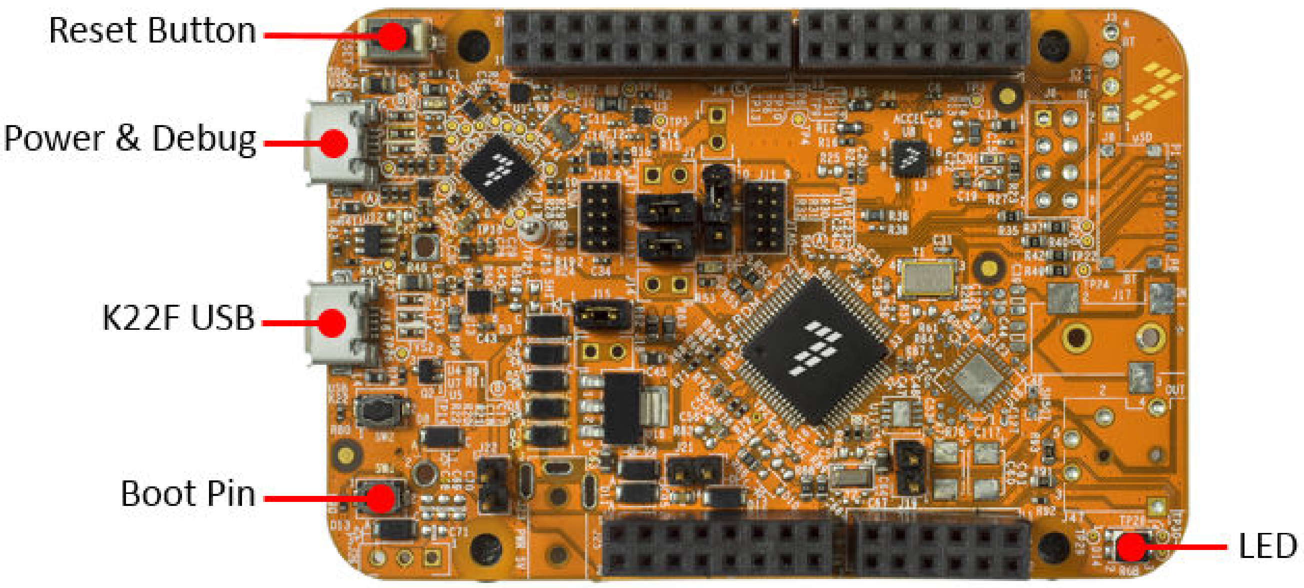


Figure 15. FRDM-K22F platform

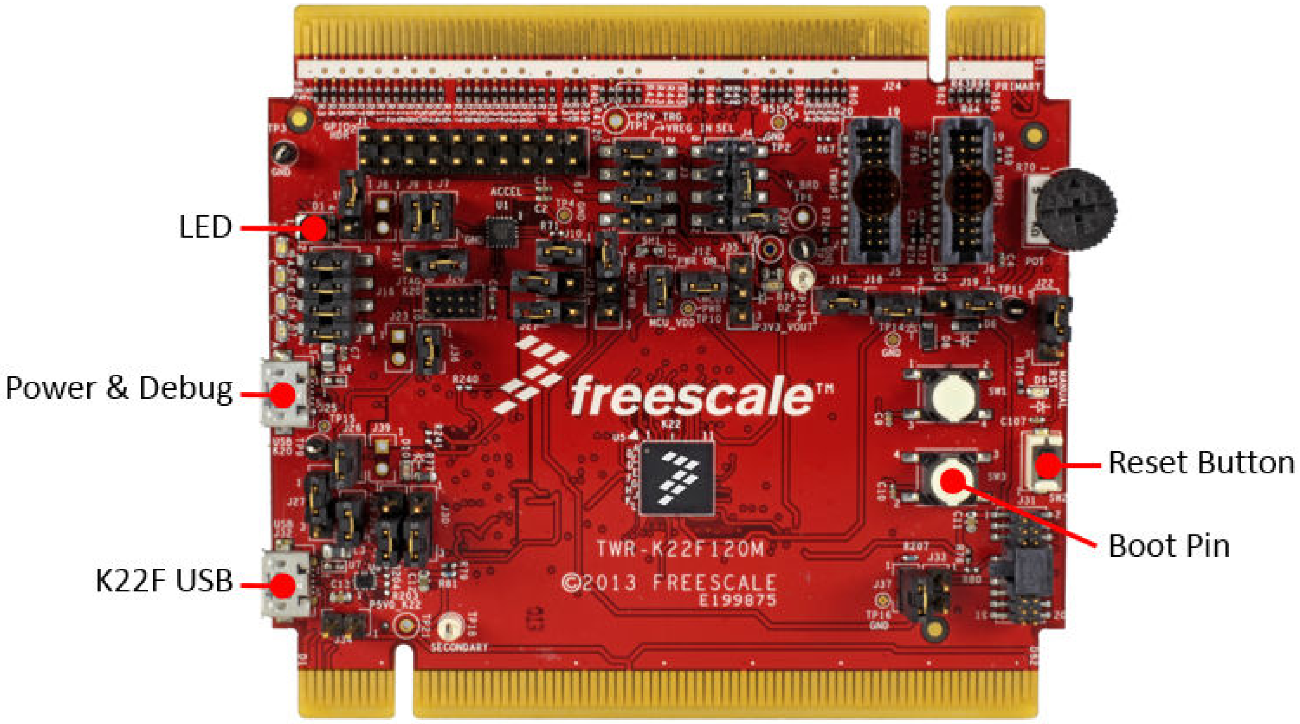


Figure 16. TWR-K22F120M platform

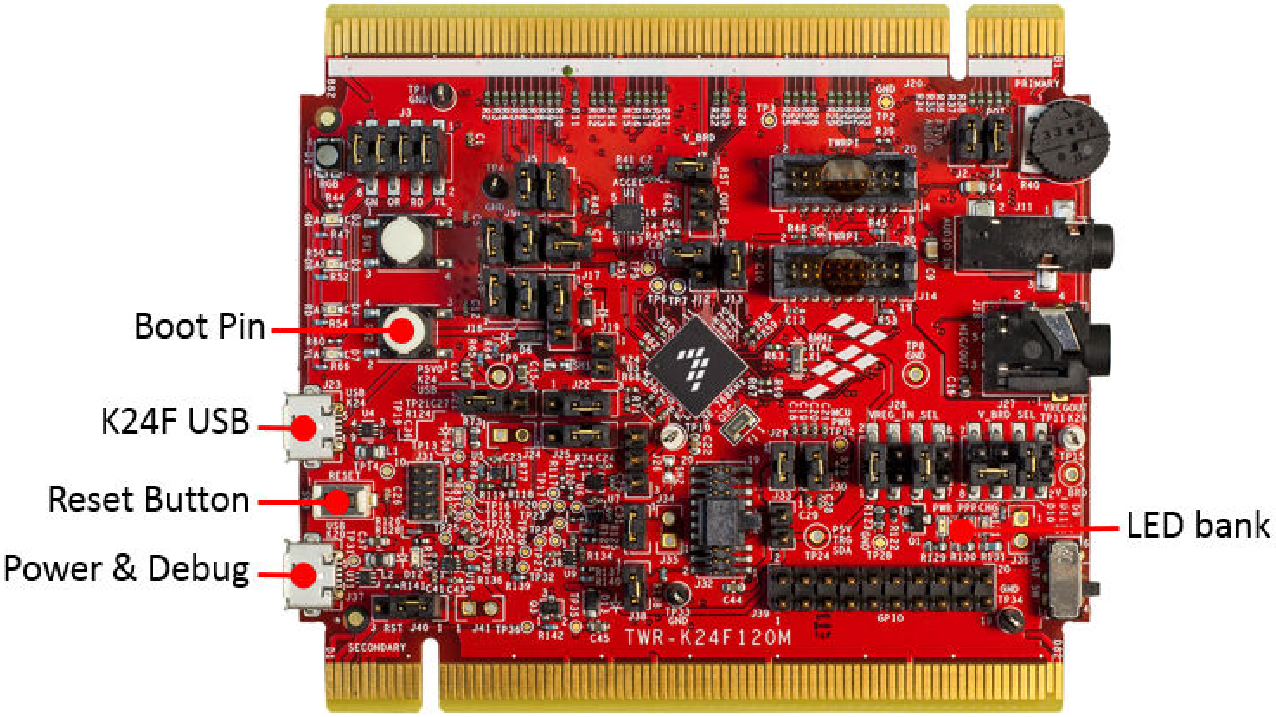


Figure 17. TWR-K24F120M platform

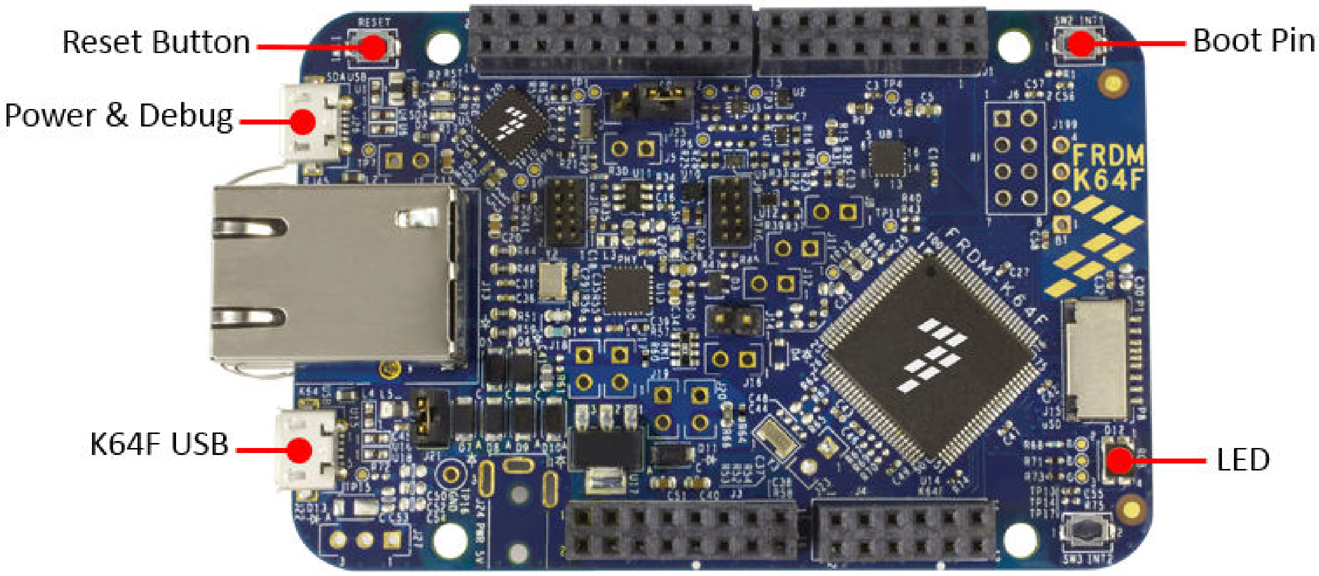


Figure 18. FRDM-K64F platform

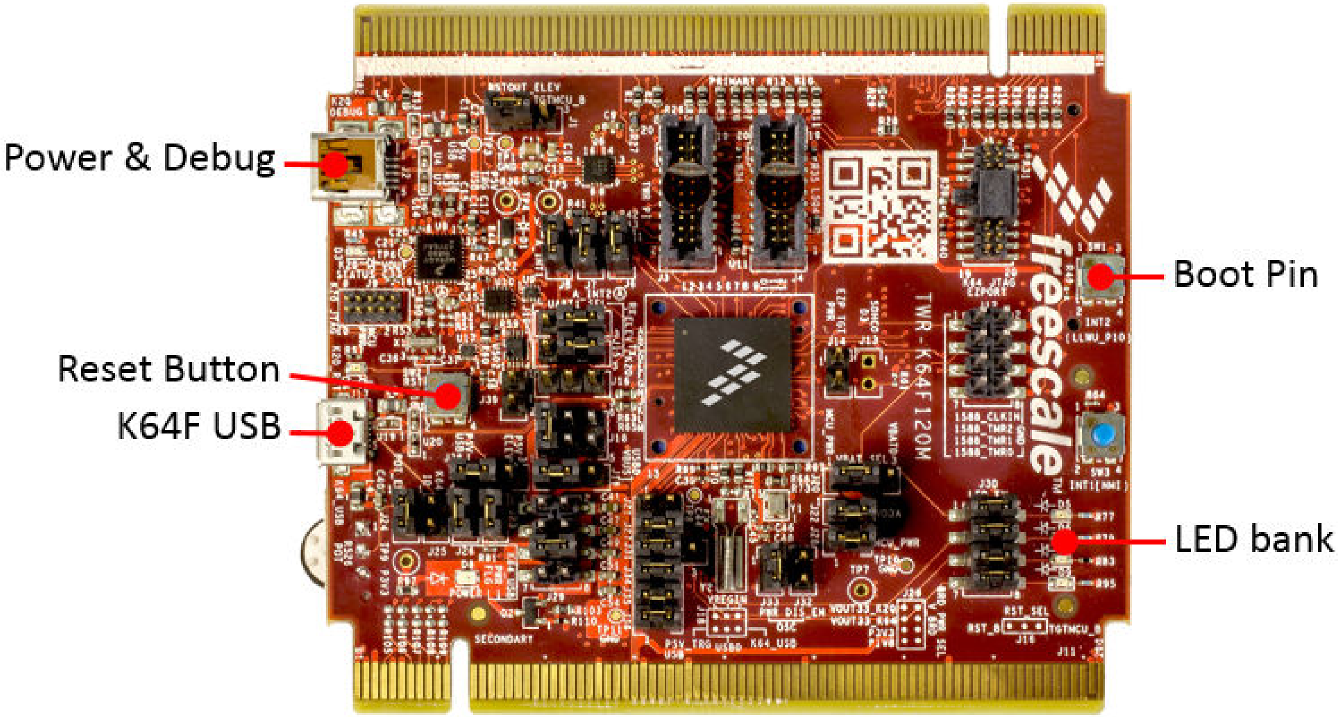


Figure 19. TWR-K64F120M platform

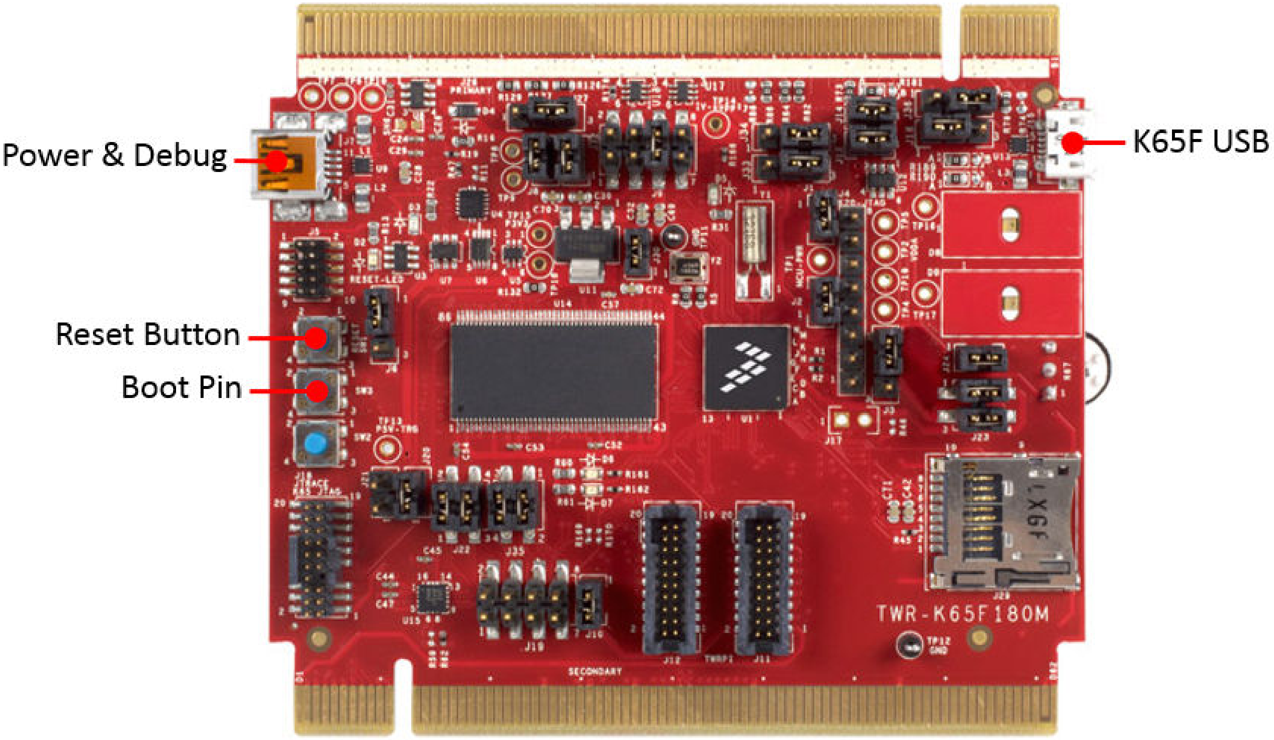


Figure 20. TWR-K65F180M platform

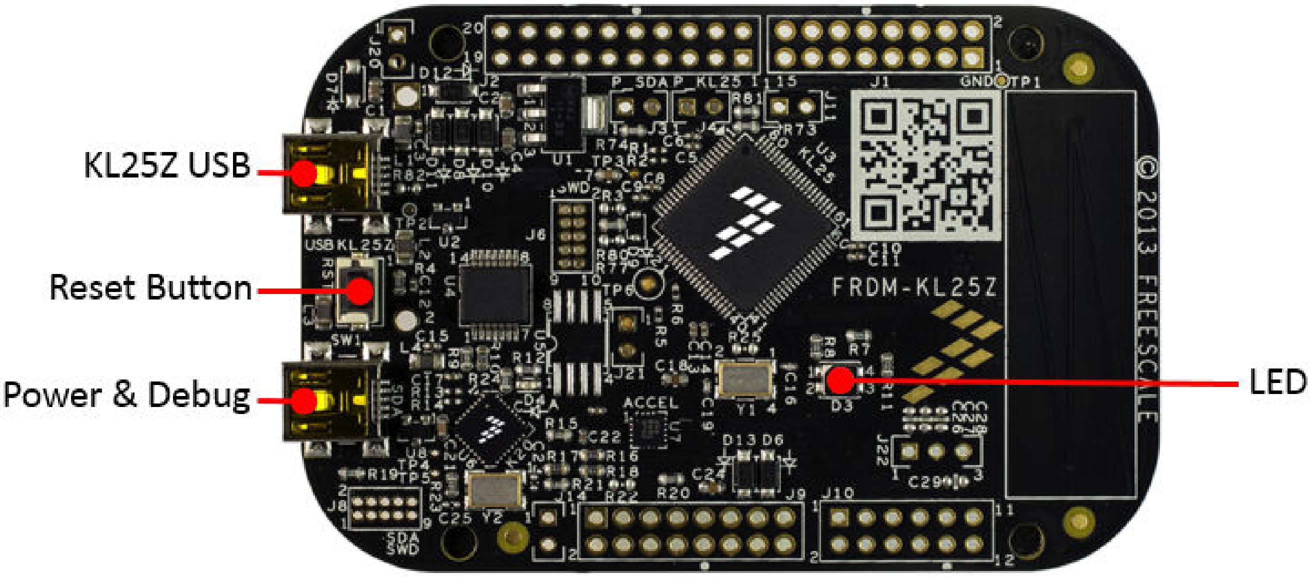


Figure 21. FRDM-KL25Z platform

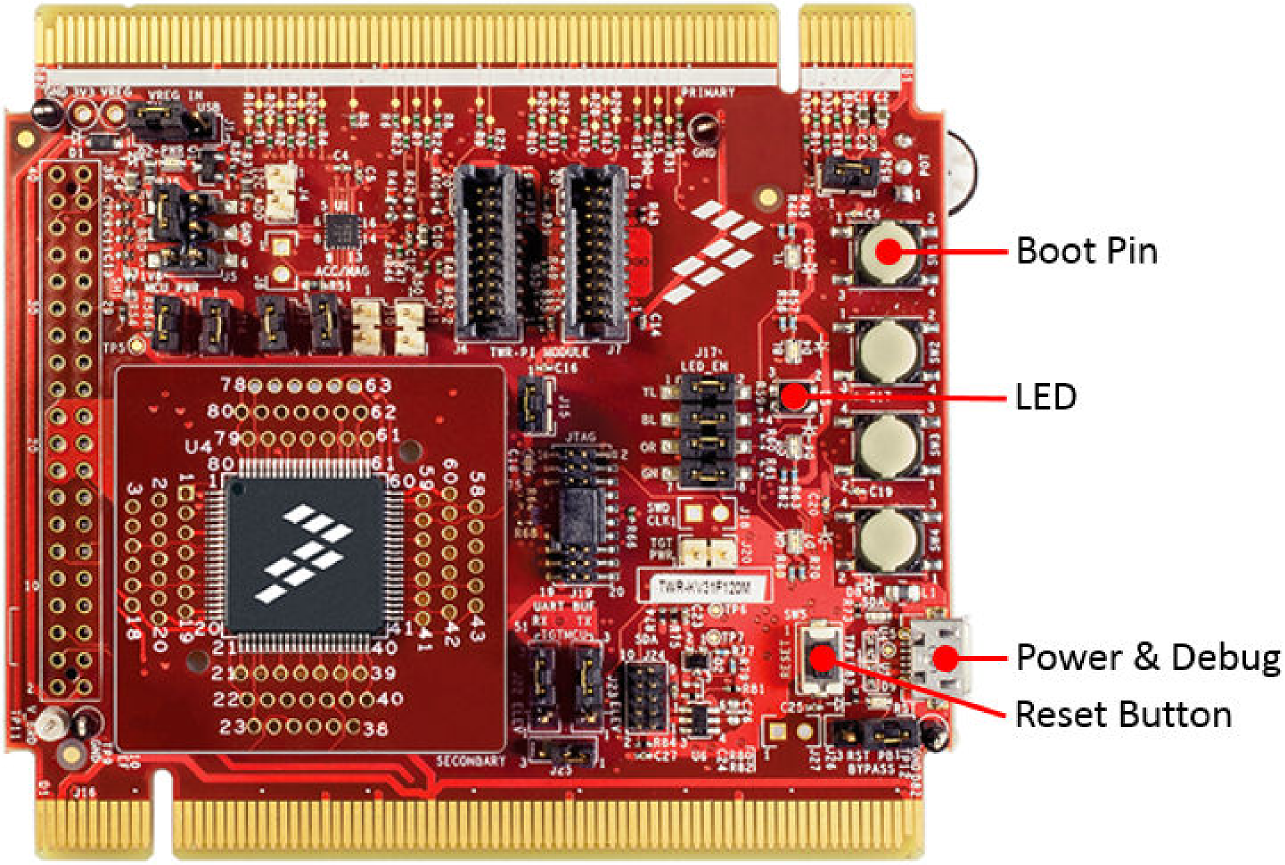


Figure 22. TWR-KV31F120M platform

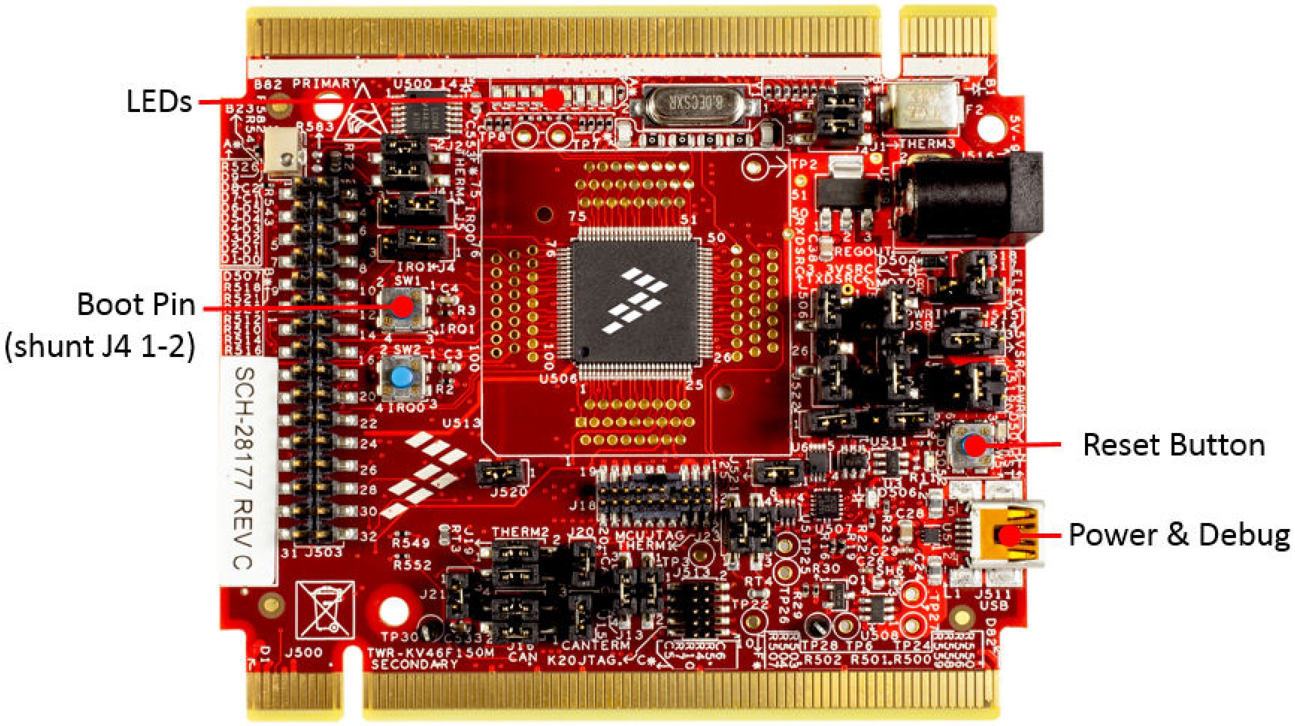


Figure 23. TWR-KV46F150M platform

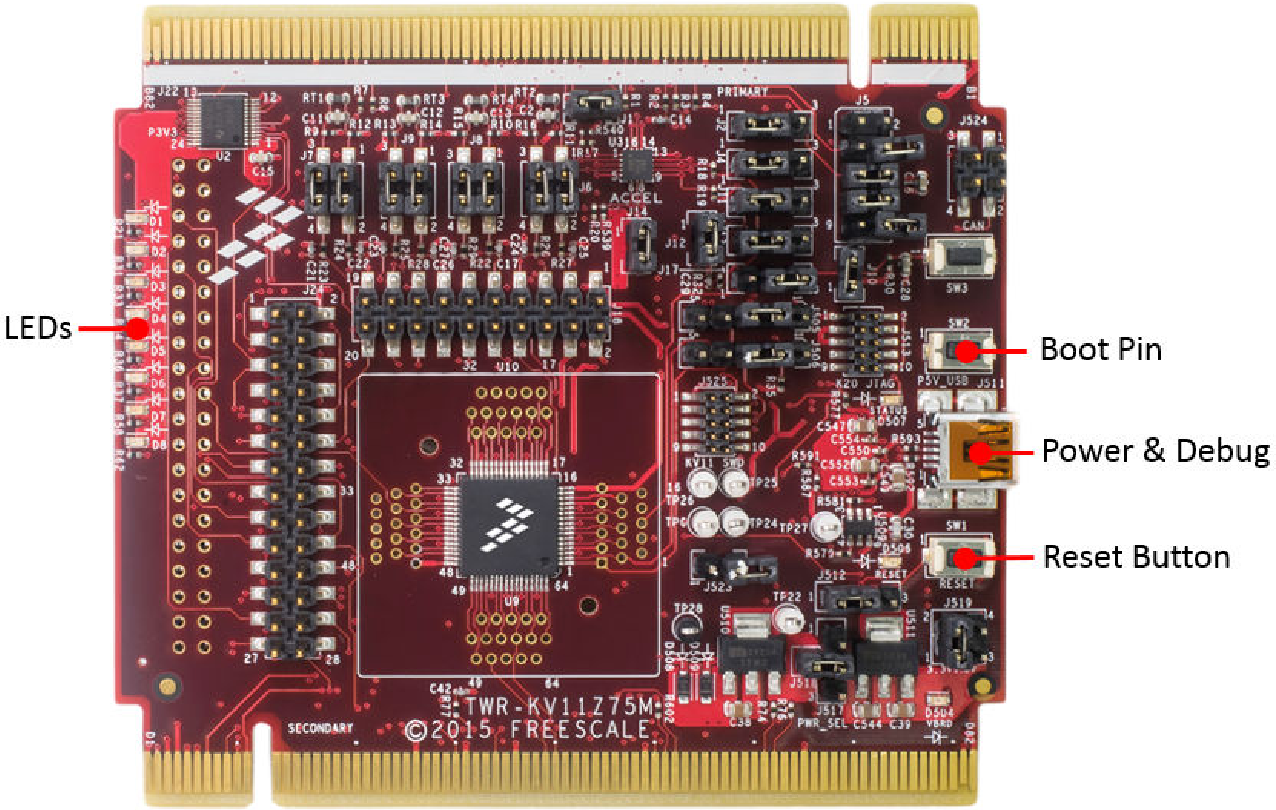


Figure 24. TWR-KV11Z75M platform

9 Appendix C - Kinetis Bootloader Pin mappings

Table 1. MK22F128R/256R/512R bootloader/flashloader – TWR-K22F120M/ FRDM-K22F

Peripheral	Instance	Port	Signal	AltMode	TWR-K22F120M Test points	FRDM-K22F Test points
UART	1	PTE0	UART1_TX	3	OpenSDA port J25	mbed J5
		PTE1	UART1_RX			
I2C	1	PTC10	I2C1_SCL	2	J9 pin 1	J1 pin 13
		PTC11	I2C1_SDA		J7 pin 1	J2 pin 7
SPI	1	PTD4	SPI1_PCS0	7	J16 pin 1	J2 pin 6
		PTD5	SPI1_SCK		J16 pin 3	J2 pin 12
		PTD6	SPI1_SOUT		J16 pin 5	J2 pin 8
		PTD7	SPI1_SIN		J16 pin 7	J2 pin 10
USB			USB0_DP	Default	USB J32	USB J16
			USB0_DM			

Table 2. MK24F25612 bootloader – TWR-K24F120M

Peripheral	Instance	Port	Signal	AltMode	TWR-K24F120M Test points
UART	1	PTE0	UART1_TX	3	mbed port J37
		PTE1	UART1_RX		
I2C	1	PTC10	I2C1_SCL	2	Elev B50
		PTC11	I2C1_SDA		Elev B51
SPI	1	PTD4	SPI1_PCS0	7	J3 pin 1
		PTD5	SPI1_SCK		J3 pin 3
		PTD6	SPI1_SOUT		J3 pin 5
		PTD7	SPI1_SIN		J3 pin 7
USB			USB0_DP	Default	USB J23
			USB0_DM		

NOTE

If testing the UART interface on mbed port J37, add shunts on J25 pin 2-3. If testing the UART interface on TWR-SER, add shunts on J25 pin 1-2 and J22 pin 1-2.

Table 3. MKL25Z4 bootloader – FRDM-KL25Z

Peripheral	Instance	Port	Signal	AltMode	FRDM-KL25Z Test points
UART	0	PTA2	UART0_TX	2	OpenSDA
		PTA1	UART0_RX		
I2C	0	PTC8	I2C0_SCL	2	J1 pin 14
		PTC9	I2C0_SDA		J1 pin 16
SPI	0	PTD0	SPI0_PCS0	2	J2 pin 6
		PTD1	SPI0_SCK		J2 pin 12
		PTD2	SPI0_SOUT		J2 pin 8
		PTD3	SPI0_SIN		J2 pin 10
USB			USB0_DP	Default	USB connector KL25Z
			USB0_DM		

Table 4. MK64F12 bootloader – TWR-K64F120M

Peripheral	Instance	Port	Signal	AltMode	TWR-K64F120M Test points
UART	1	PTC4	UART1_TX	3	OpenSDA J2
		PTC3	UART1_RX		
I2C	1	PTC10	I2C1_SCL	2	Elev A75 or J3 pin 3
		PTC11	I2C1_SDA		Elev B71 or J3 pin 4

Table continues on the next page...

Table 4. MK64F12 bootloader – TWR-K64F120M (continued)

Peripheral	Instance	Port	Signal	AltMode	TWR-K64F120M Test points
SPI	0	PTD0	SPI1_PCS0	2	Elev B46
		PTD1	SPI1_SCK		Elev B48
		PTD2	SPI1_SOUT		Elev B45
		PTD3	SPI1_SIN		Elev B44
USB			USB0_DP	Default	USB J17
			USB0_DM		

Table 5. MK64F12 flashloader – TWR-K64F120M

Peripheral	Instance	Port	Signal	AltMode	TWR-K64F120M Test points
UART	0	PTB17	UART0_TX	3	Elev B11
		PTB16	UART0_RX		Elev B10
I2C	1	PTC10	I2C1_SCL	2	Elev A75 or J3 pin 3
		PTC11	I2C1_SDA		Elev B71 or J3 pin 4
SPI	0	PTD0	SPI1_PCS0	2	Elev B46
		PTD1	SPI1_SCK		Elev B48
		PTD2	SPI1_SOUT		Elev B45
		PTD3	SPI1_SIN		Elev B44
USB			USB0_DP	Default	USB J17
			USB0_DM		

Table 6. MK64F12 bootloader/flashloader – FRDM-K64F

Peripheral	Instance	Port	Signal	AltMode	FRDM-K64F Test points
UART	0	PTB17	UART0_TX	3	mbed port J26
		PTB16	UART0_RX		
I2C	1	PTC10	I2C1_SCL	2	J4 pin 12
		PTC11	I2C1_SDA		J4 pin 10
SPI	0	PTD0	SPI1_PCS0	2	J2 pin 6
		PTD1	SPI1_SCK		J2 pin 12
		PTD2	SPI1_SOUT		J2 pin 8
		PTD3	SPI1_SIN		J6 pin 10
USB			USB0_DP	Default	USB J22
			USB0_DM		

Table 7. MK65F18 bootloader – TWR-K65F180M

Peripheral	Instance	Port	Signal	AltMode	TWR-K65F180M Test points
UART	2	PTE16	UART2_TX	3	mbed port J7
		PTE17	UART2_RX		
I2C	0	PTC10	I2C0_SCL	2	J13 pin 2
		PTC11	I2C0_SDA		J14 pin 1
SPI	2	PTD11	SPI2_PCS0	2	Elev B46
		PTD12	SPI2_SCK		Elev B48
		PTD13	SPI2_SOUT		Elev B45
		PTD14	SPI2_SIN		Elev B44
USB			USB0_DP	Default	TWR_SER USB J14
			USB0_DM		
HS USB			USB1_DM	Default	USB J15
			USB1_DP		

Table 8. MK65F18 flashloader – TWR-K65F180M

Peripheral	Instance	Port	Signal	AltMode	TWR-K65F180M Test points
UART	4	PTE24	UART4_TX	3	Elev A48
		PTE25	UART4_RX		Elev A47
I2C	0	PTC10	I2C0_SCL	2	J13 pin 2
		PTC11	I2C0_SDA		J14 pin 1
SPI	2	PTD11	SPI2_PCS0	2	Elev B46
		PTD12	SPI2_SCK		Elev B48
		PTD13	SPI2_SOUT		Elev B45
		PTD14	SPI2_SIN		Elev B44
USB			USB0_DP	Default	TWR_SER USB J14
			USB0_DM		
HS USB			USB1_DM	Default	USB J15
			USB1_DP		

Table 9. MKV31F128/256/512 bootloader – TWR-KV31F120M

Peripheral	Instance	Port	Signal	AltMode	TWR-KV31F120M Test points
UART	0	PTB16	UART0_TX	3	OpenSDA port
		PTB17	UART0_RX		
I2C	0	PTD2	I2C0_SCL	7	J9 pin 2
		PTD3	I2C0_SDA		J12 pin 1

Table continues on the next page...

Table 9. MKV31F128/256/512 bootloader – TWR-KV31F120M (continued)

Peripheral	Instance	Port	Signal	AltMode	TWR-KV31F120M Test points
SPI	0	PTE16	SPI0_PCS0	2	Elev B46
		PTE17	SPI0_SCK		Elev B48
		PTE18	SPI0_SOUT		Elev B45
		PTE19	SPI0_SIN		Elev B44

Table 10. MKV46F15 bootloader – TWR-KV46F150M

Peripheral	Instance	Port	Signal	AltMode	TWR-KV46F150M Test points
UART	0	PTD6	UART0_TX	3	OpenSDA port
		PTD7	UART0_RX		
I2C	0	PTB0	I2C0_SCL	2	Elev B28 or J501 pin 22
		PTB1	I2C0_SDA		Elev B27 or J501 pin 33
SPI	0	PTE16	SPI0_PCS0	2	Elev A27 or J501 pin 10
		PTE17	SPI0_SCK		Elev A28 or J501 pin 12
		PTE18	SPI0_SOUT		Elev B29 or J501 pin 18
		PTE19	SPI0_SIN		Elev B30 or J501 pin 20
FlexCAN	0	PTA12	CAN0_TX	2	J13 pin 1
		PTA13	CAN0_RX		J13 pin 2

Table 11. MKV11Z7 bootloader – TWR-KV11Z75M

Peripheral	Instance	Port	Signal	AltMode	TWR-KV11Z75M Test points
UART	0	PTD17	UART0_TX	3	OpenSDA J511
		PTD16	UART0_RX		
I2C	0	PTB0	I2C0_SCL	2	J18 pin 17
		PTB1	I2C0_SDA		J18 pin 18
SPI	0	PTE16	SPI0_PCS0	2	J18 pin 5
		PTE17	SPI0_SCK		J18 pin 6
		PTE18	SPI0_SOUT		J18 pin 7
		PTE19	SPI0_SIN		J18 pin 8
FlexCAN	0	PTA24	CAN0_TX	2	J24 pin 13
		PTA25	CAN0_RX		J24 pin 14

NOTE

If testing the UART interface on OpenSDA port J511, add shunts on J505 pin 2-3 and J506 pin 2-3.

Table 12. MKV11Z7 flashloader – TWR-KV11Z75M

Peripheral	Instance	Port	Signal	AltMode	TWR-KV11Z75M Test points
UART	0	PTD6	UART0_TX	3	J24 pin 27
		PTD7	UART0_RX		J24 pin 28
I2C	0	PTB0	I2C0_SCL	2	J18 pin 17
		PTB1	I2C0_SDA		J18 pin 18
SPI	0	PTE16	SPI0_PCS0	2	J18 pin 5
		PTE17	SPI0_SCK		J18 pin 6
		PTE18	SPI0_SOUT		J18 pin 7
		PTE19	SPI0_SIN		J18 pin 8
FlexCAN	0	PTA24	CAN0_TX	2	J24 pin 13
		PTA25	CAN0_RX		J24 pin 14

NOTE

CAN connection – option 1, use CAN transceiver on board:

1. Put jumper on J5 pin 1-2, and keep default jumpers on J5 5-6, and 7-8
2. Connect TWR-KV11Z75M J524 pin 2 to BusPal (KV46) J13 pin 2
3. Connect TWR-KV11Z75M J524 pin 1 to BusPal (KV46) J13 pin 1

CAN connection - option 2, use CAN transceiver on TWR-SER board:

1. Remove the jumpers on TWR-SER J5 pins 5-6 and pins 7-8
2. Wring CAN0_TX

   Wire CAN0_TX on TWR-KV11Z75M J24 pin 13 to TWR-SER J5 pin 8 - signal name C_TXD

3. Wring CAN0_RX

   Wire CAN0_TX on TWR-KV11Z75M J24 pin 14 to TWR-SER J5 pin 6 - signal name C_RXD

4. Connect to BusPal (KV46)

   Connect TWR-SER CANH, J7 pin 1 to BusPal KV46 J13 pin 2

   Connect TWR-SER CANH, J7 pin 3 to BusPal KV46 J13 pin 1

10 Revision history

This table summarizes revisions to this document.

Table 13. Revision history

Revision number	Date	Substantive changes
0	07/2015	Kinetis Bootloader 1.2.0 initial release

How to Reach Us:

Home Page:
freescale.com

Web Support:
freescale.com/support