Freescale Semiconductor
User's Guide

Document Number: KBTLDRDEMOUG
Rev. 2, 04/2016

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 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/<mcu>/<tool chain>/binaries`.

2.4 Host updater

The KinetisFlashTool.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 Flash Tool application, see the *Kinetis Flash Tool User's Guide* (document KFLASHTOOLUG).

2.5 Toolchain requirement

Firmware projects:

- IAR Embedded Workbench for ARM® v.7.50.1 or later
- Python v.2.7 (www.python.org)
- Kinetis Design Studio IDE (KDS) v.3.2.0
- Keil MDK v.5.18 with corresponding device packs

Host projects:

- Microsoft® Visual Studio® Professional 2015 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 v.7.3 (for the blhost and elftosb tools)
- GNU Compiler (GCC) v.4.8.1 (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 a 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-KL28Z, FRDM-K64F, FRDM-K82F, FRDM-KL82Z, TWR-K24F120M, TWR-KL28Z72M, TWR-K80F150M, and TWR-KL82Z72M:

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 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.

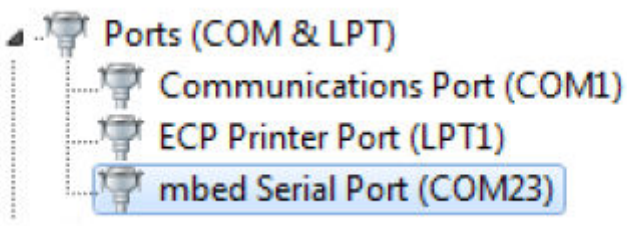


Figure 1. mbed serial port in Windows Device Manager

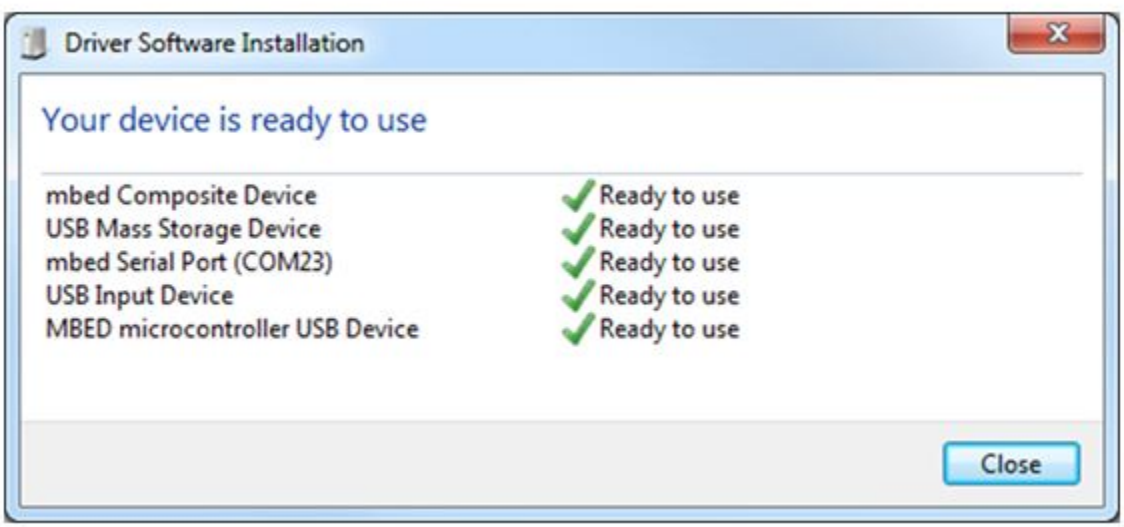


Figure 2. Driver software installation

FRDM-KL03Z, FRDM-KL25Z, FRDM-KV31F, FRDM-K66F, MAPS-KS22, TWR-KV11Z75M, TWR-K22F120M, TWR-KV30F100M, TWR-KV31F120M, TWR-KV46F150M, TWR-KV58F220M, 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, J2 for the TWR-K64F120M.
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.

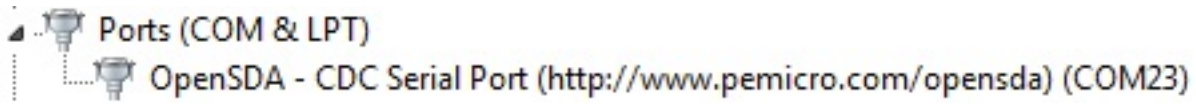


Figure 3. OpenSDA virtual comport in Windows Device Manager

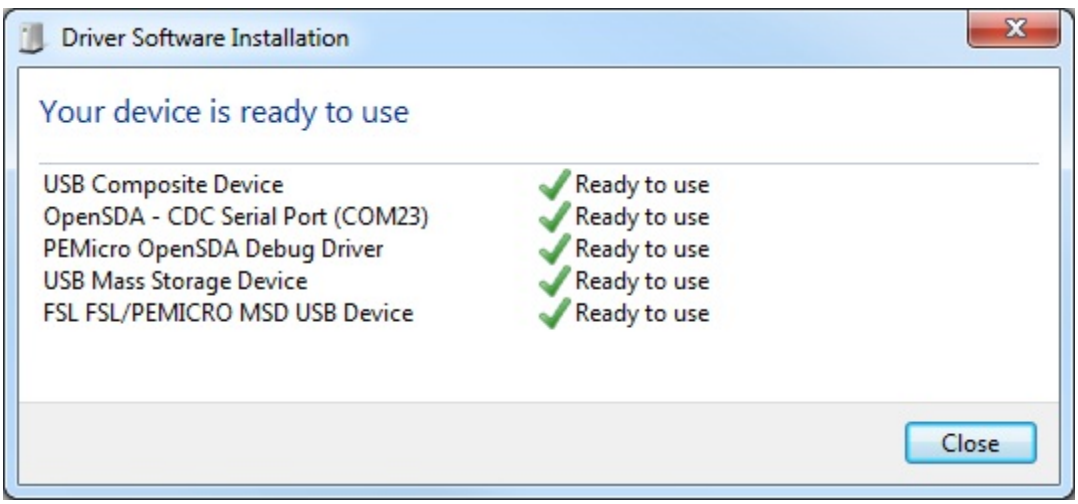
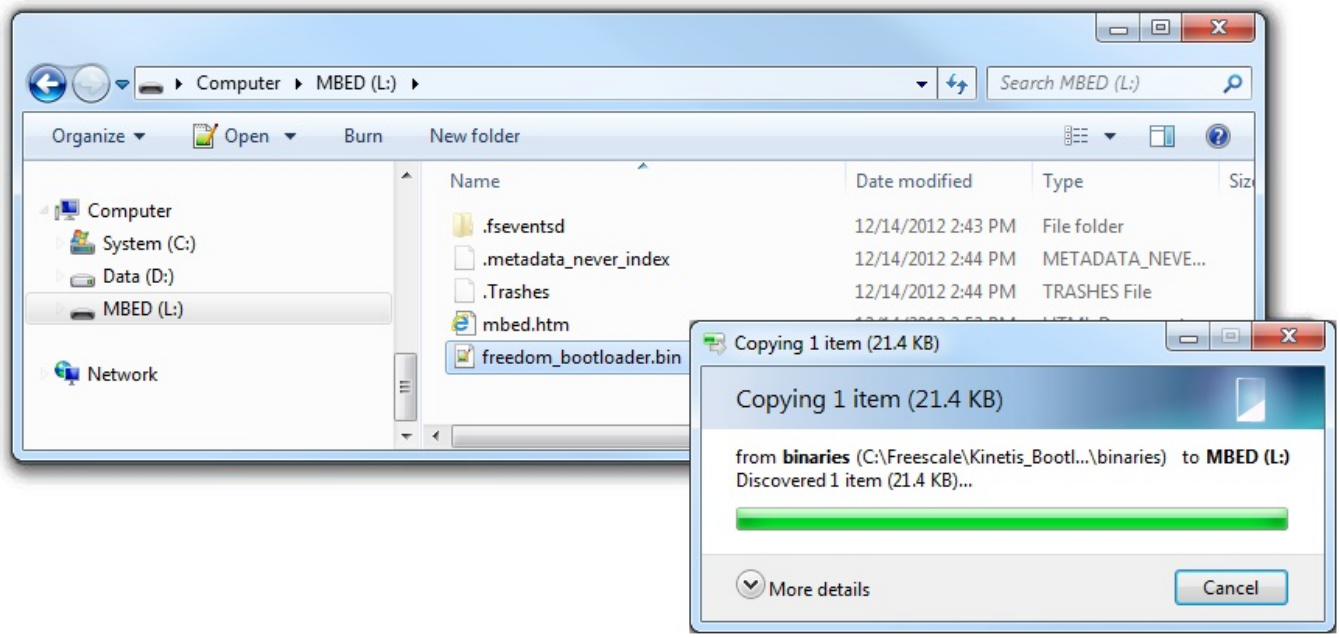


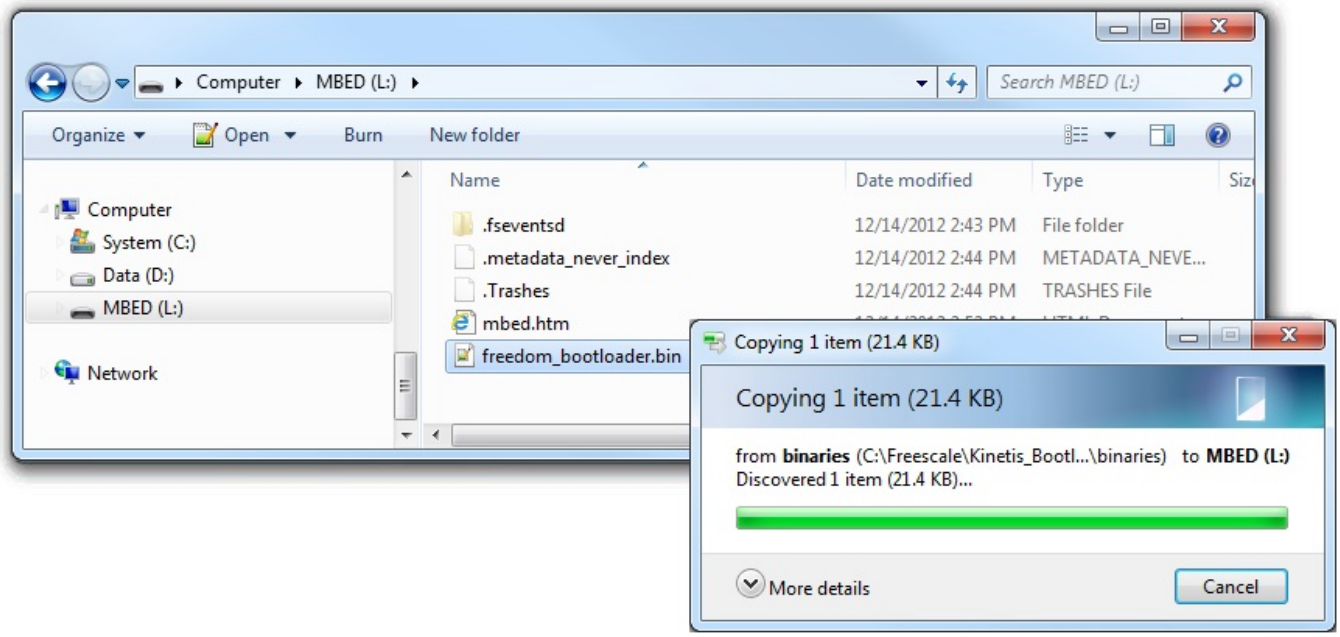
Figure 4. Driver software installation

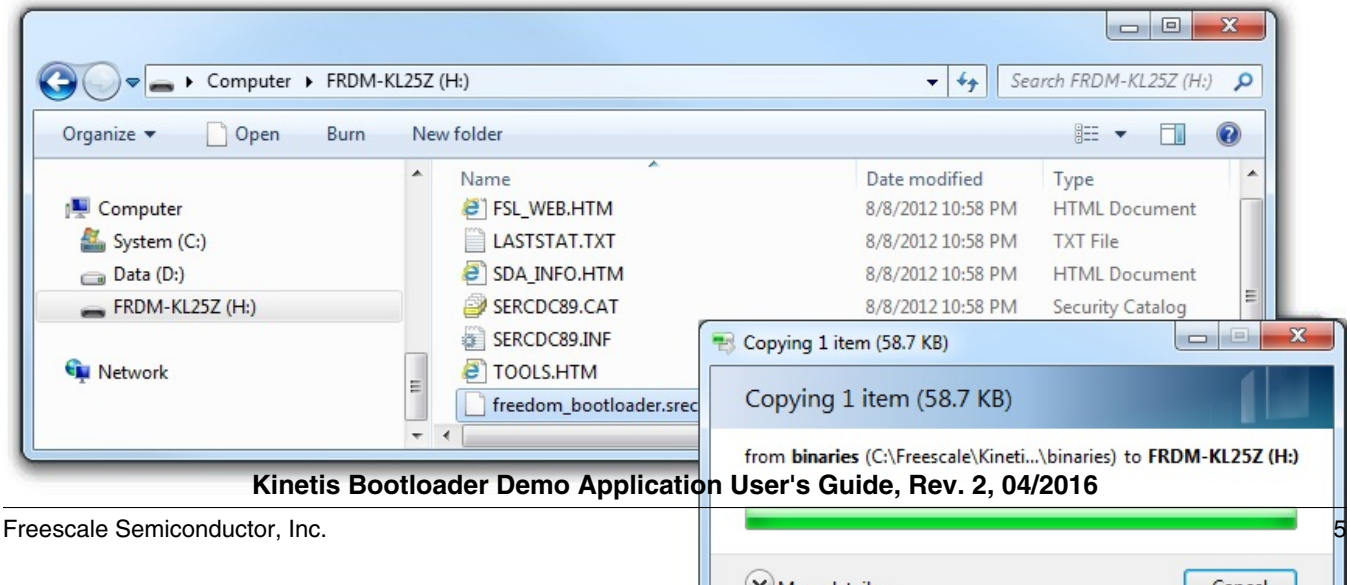
3.1.1 Install the Kinetis bootloader

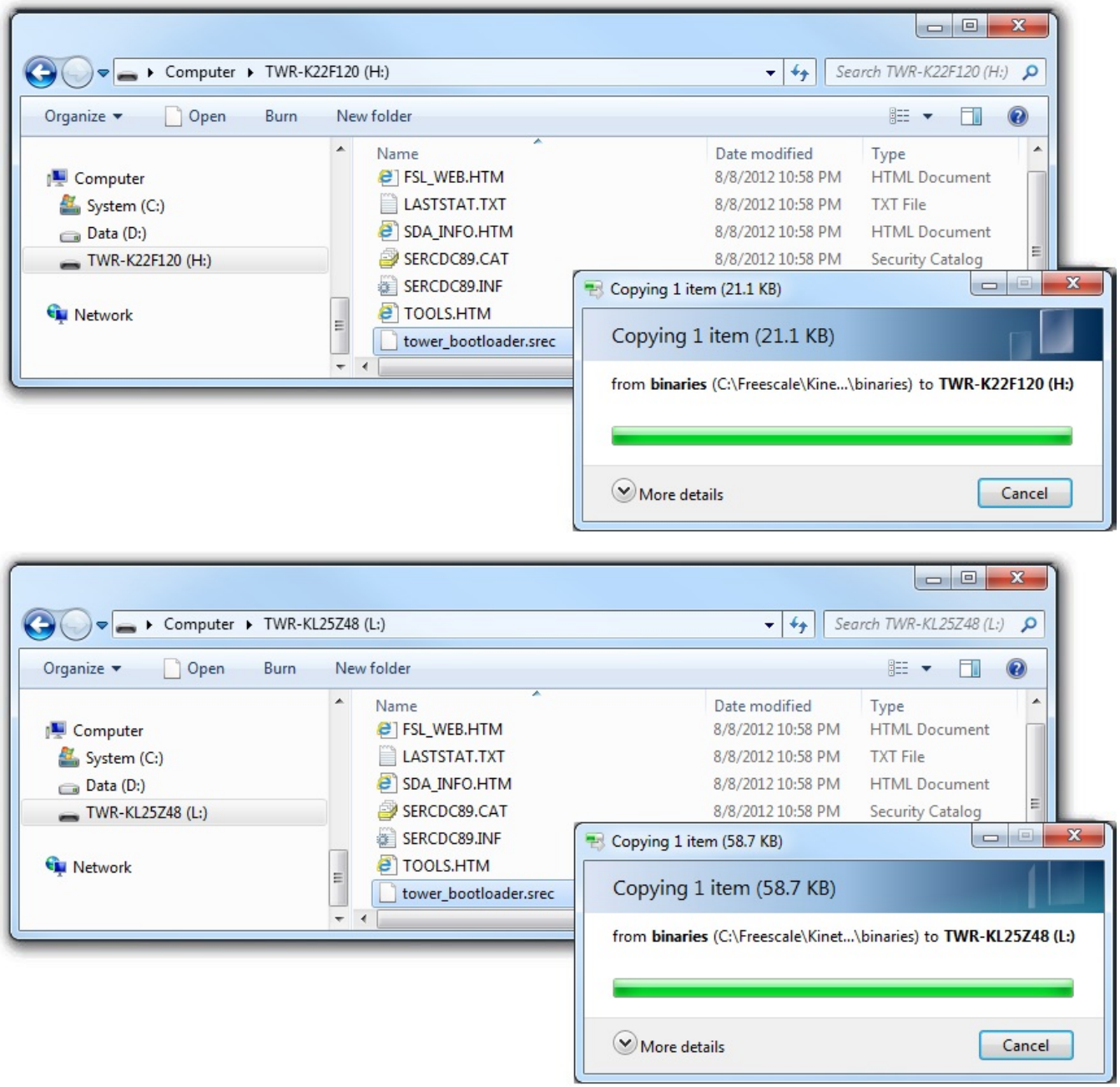
To install the bootloader application, drag or copy and paste the appropriate binary (Tower or Freedom) file from `<install_dir>/targets/<device>/<tool chain>/binaries` onto the mass storage device, where <device> is the MCU family container folder. While it is possible to use the flashloader image (for 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 the 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 the platform is a 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.


Computer ▸ MBED (L:) ▸
Search MBED (L:)
Organize
Open
Burn
New folder
Computer
System (C:)
Data (D:)
MBED (L:)
Network
Name
Date modified
Type
.fseventsd
12/14/2012 2:43 PM
File folder
.metadata_never_index
12/14/2012 2:44 PM
METADATA_NEVE...
.Trashes
12/14/2012 2:44 PM
TRASHES File
mbed.htm
freedom_bootloader.bin
Copying 1 item (21.4 KB)
Copying 1 item (21.4 KB)
from binaries (C:\Freescale\Kinetis_Bootl...\binaries) to MBED (L:)
Discovered 1 item (21.4 KB)...
More details
Cancel


Computer ▸ MBED (L:) ▸
Search MBED (L:)
Organize
Open
Burn
New folder
Computer
System (C:)
Data (D:)
MBED (L:)
Network
Name
Date modified
Type
.fseventsd
12/14/2012 2:43 PM
File folder
.metadata_never_index
12/14/2012 2:44 PM
METADATA_NEVE...
.Trashes
12/14/2012 2:44 PM
TRASHES File
mbed.htm
freedom_bootloader.bin
Copying 1 item (21.4 KB)
Copying 1 item (21.4 KB)
from binaries (C:\Freescale\Kinetis_Bootl...\binaries) to MBED (L:)
Discovered 1 item (21.4 KB)...
More details
Cancel


Computer ▸ FRDM-KL25Z (H:)
Search FRDM-KL25Z (H:)
Organize
Open
Burn
New folder
Computer
System (C:)
Data (D:)
FRDM-KL25Z (H:)
Network
Name
Date modified
Type
FSL_WEB.HTM
8/8/2012 10:58 PM
HTML Document
LASTSTAT.TXT
8/8/2012 10:58 PM
TXT File
SDA_INFO.HTM
8/8/2012 10:58 PM
HTML Document
SERCDC89.CAT
8/8/2012 10:58 PM
Security Catalog
SERCDC89.INF
TOOLS.HTM
freedom_bootloader.srec
Copying 1 item (58.7 KB)
Copying 1 item (58.7 KB)
from binaries (C:\Freescale\Kineti...\binaries) to FRDM-KL25Z (H:)


Computer ▸ TWR-K22F120 (H:)
Search TWR-K22F120 (H:)
Organize
Open
Burn
New folder
Computer
System (C:)
Data (D:)
TWR-K22F120 (H:)
Network
Name
Date modified
Type
FSL_WEB.HTM 8/8/2012 10:58 PM HTML Document
LASTSTAT.TXT 8/8/2012 10:58 PM TXT File
SDA_INFO.HTM 8/8/2012 10:58 PM HTML Document
SERCDC89.CAT 8/8/2012 10:58 PM Security Catalog
SERCDC89.INF
TOOLS.HTM
tower_bootloader.srec
Copying 1 item (21.1 KB)
Copying 1 item (21.1 KB)
from binaries (C:\Freescale\Kine...\binaries) to TWR-K22F120 (H:)
More details
Cancel
Computer ▸ TWR-KL25Z48 (L:)
Search TWR-KL25Z48 (L:)
Organize
Open
Burn
New folder
Computer
System (C:)
Data (D:)
TWR-KL25Z48 (L:)
Network
Name
Date modified
Type
FSL_WEB.HTM 8/8/2012 10:58 PM HTML Document
LASTSTAT.TXT 8/8/2012 10:58 PM TXT File
SDA_INFO.HTM 8/8/2012 10:58 PM HTML Document
SERCDC89.CAT 8/8/2012 10:58 PM Security Catalog
SERCDC89.INF
TOOLS.HTM
tower_bootloader.srec
Copying 1 item (58.7 KB)
Copying 1 item (58.7 KB)
from binaries (C:\Freescale\Kinet...\binaries) to TWR-KL25Z48 (L:)
More details
Cancel

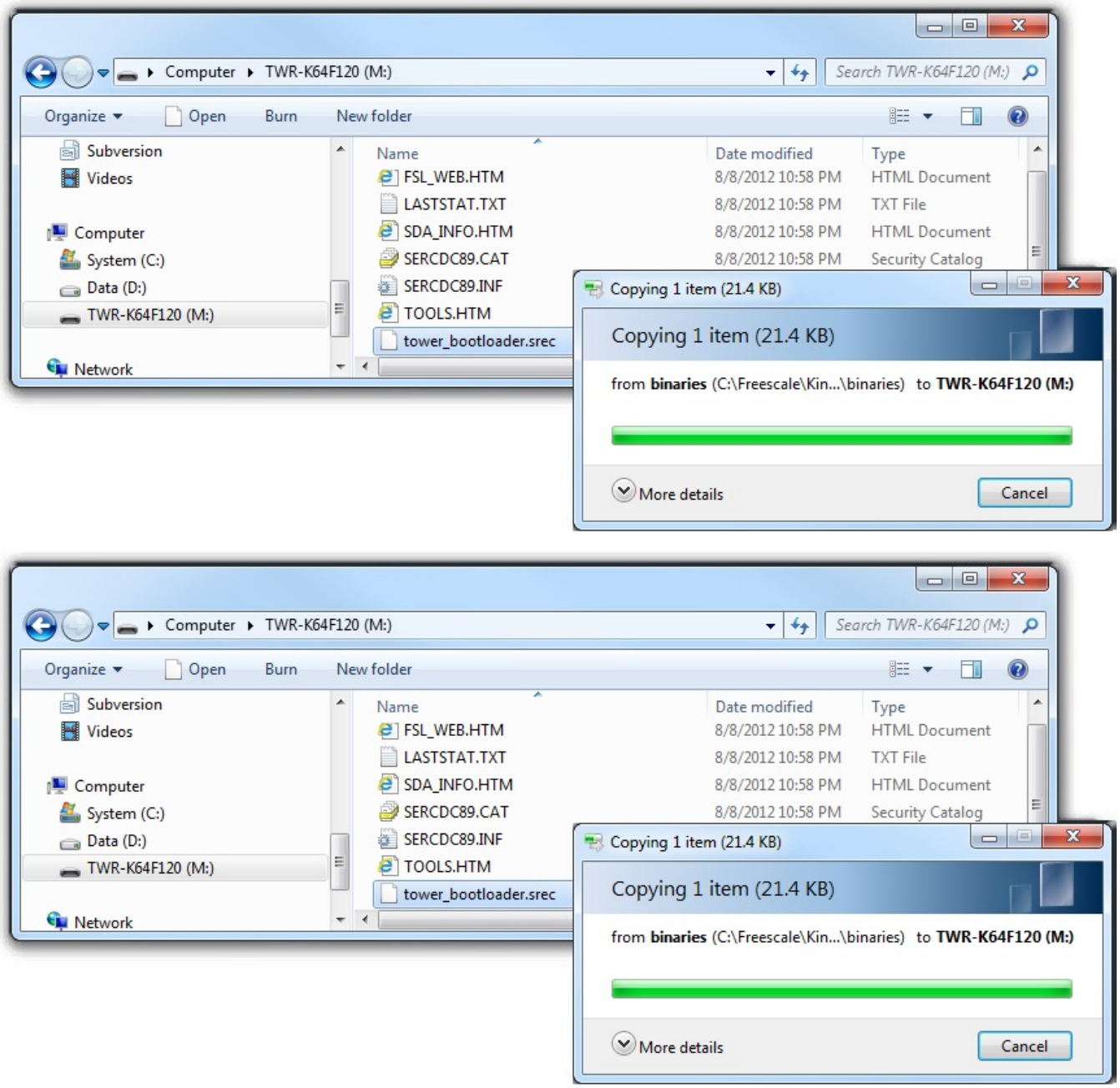


Figure 6. Install the Kinetis bootloader on the board-named drive

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/Tools/blhost/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. See step 3 of Section 3.1, "Connect the platform". 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.

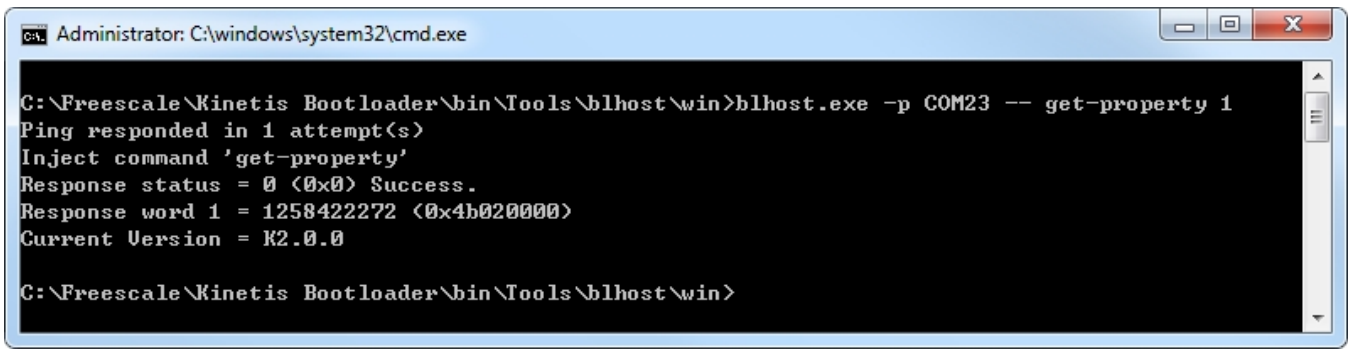


Figure 7. Host communication with Kinetis bootloader using the UART0 peripheral

For the TWR-K64F120M module:

- Press the "Reset" button on the platform.
- Type *blhost -u -- 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.

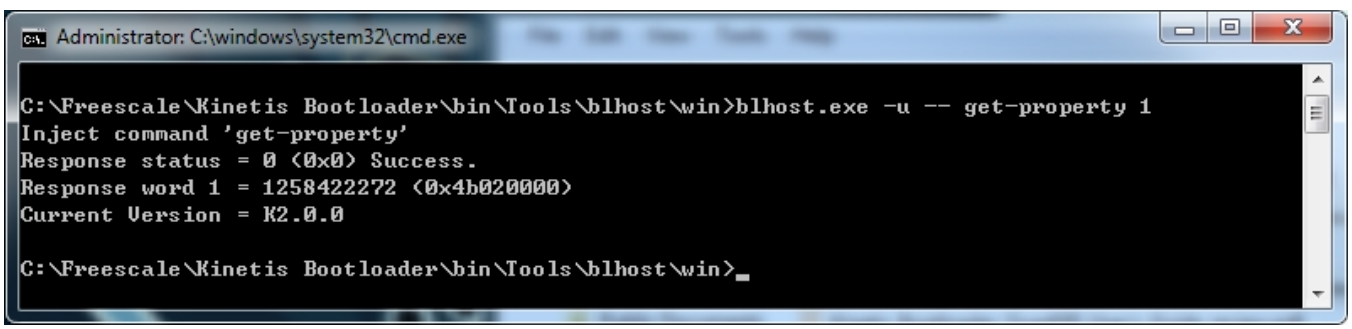


Figure 8. Host communication with Kinetis bootloader using the USB peripheral

5 Windows GUI updater application

This section describes how to use the Windows GUI updater application, KinetisFlashTool.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/Tools/KinetisFlashTool/win` directory.
3. Double-click the KinetisFlashTool.exe file to launch the app.
4. Connect the device.
   - Select the COM23 device from the "Port" drop-down box.
   - Click the "Connect" button.

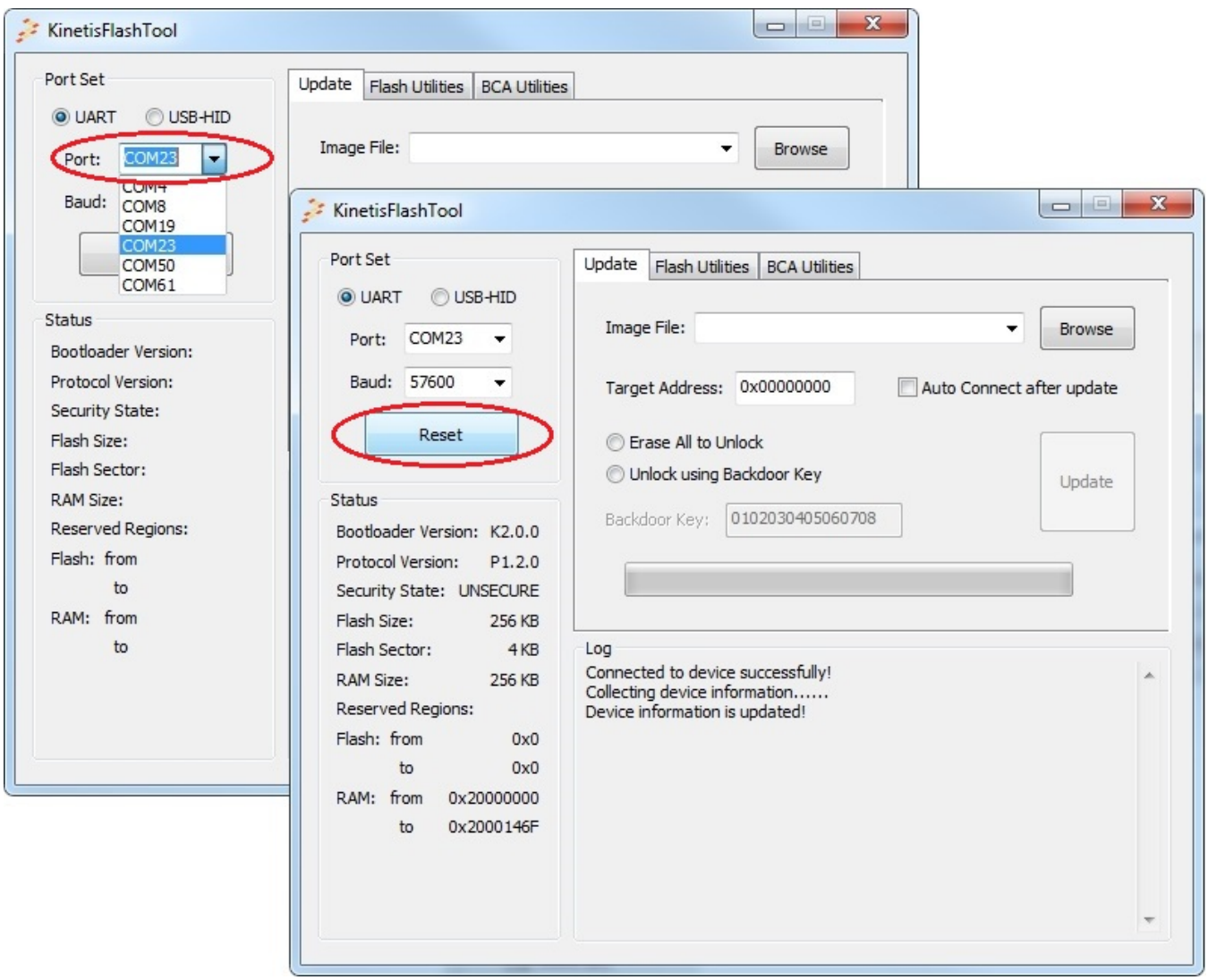


Figure 9. Connect the device

5. Select the image file.
   - Select the led_demo_FRDM-K22F_a000.bin application image from the `<install_dir>/apps/led_demo/<mcu>/<tool chain>/binaries` directory using the "Browse" button.
   - Set the target address to 0xA000.

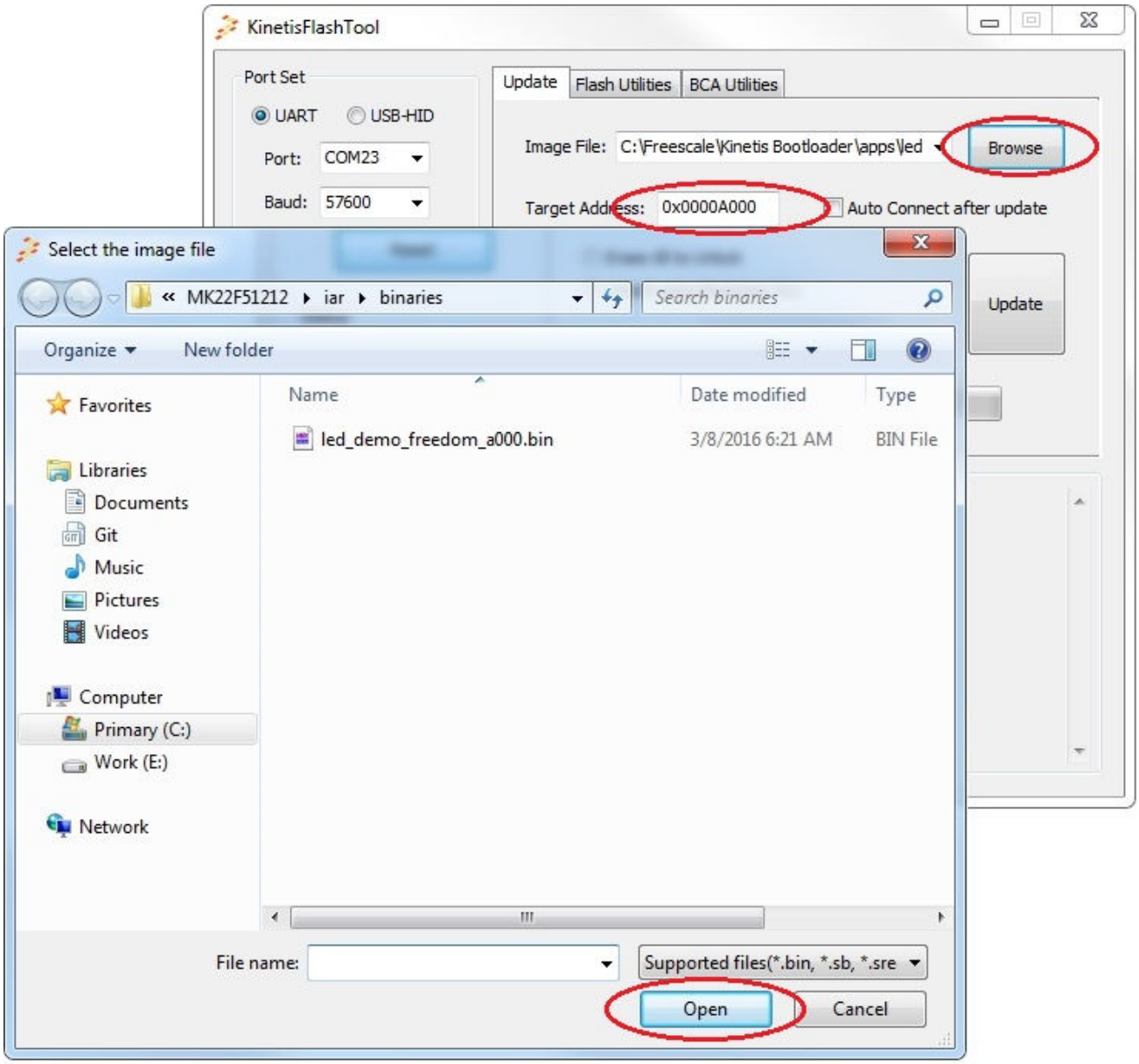


Figure 10. Browse for the user application

6. Update the image.
   - 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.

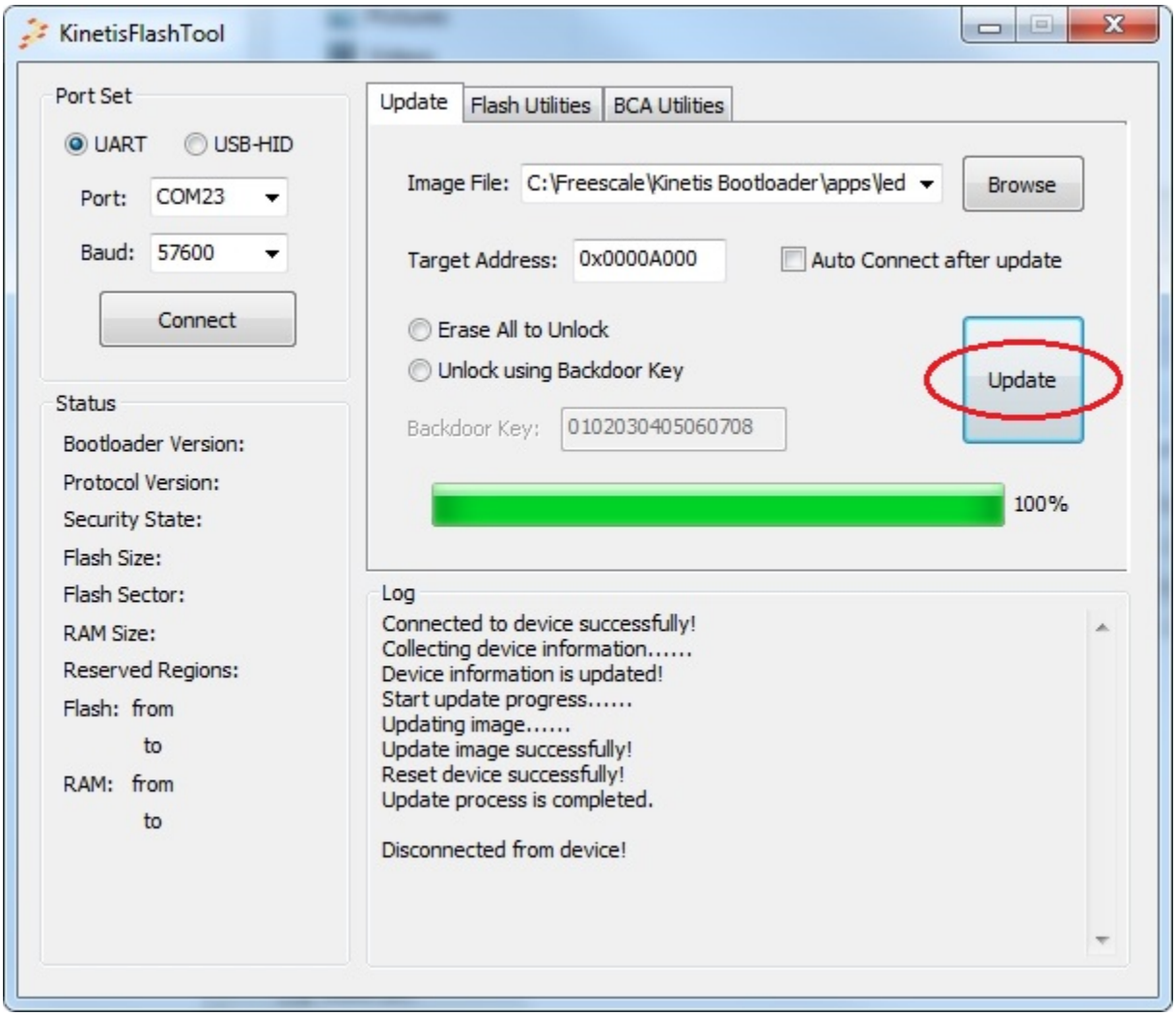


Figure 11. Perform the update

7. 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.
8. 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 the platform has a boot pin button) and resetting the board.
9. Click the "Close" button at the top right corner when finished.

6 Returning to Flash-resident bootloader

Some development platforms support re-entry of the bootloader from a user application. See Appendix B to determine if the board has a "Boot Pin" button listed.

NOTE

If the MCU on the platform has a built-in bootloader in ROM, the boot pin returns to the ROM, not the flash-resident bootloader. The following platforms have the bootloader in ROM: MK80, MK82, MKL28, and MKL82.

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 the bootloader mode by again running the blhost.exe tool as done earlier.

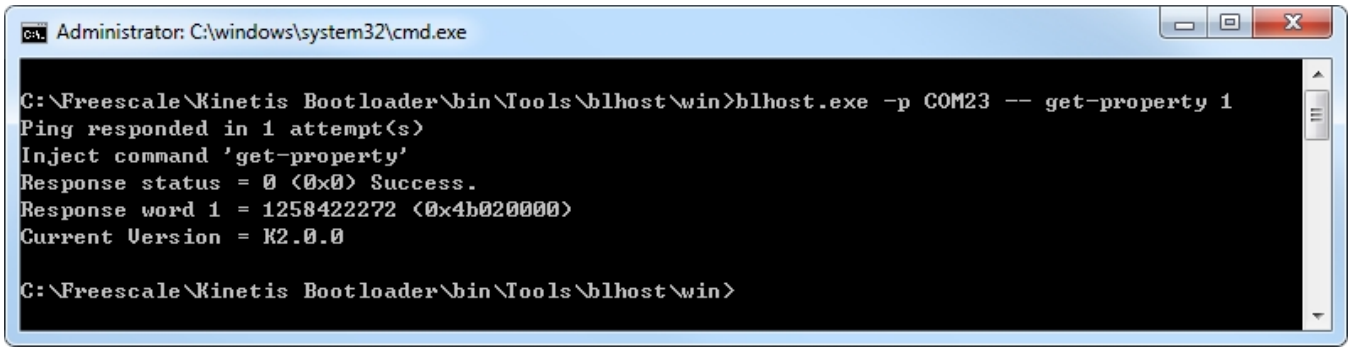


Figure 12. Back to the Kinetis bootloader interface

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

7 Appendix A - Kinetis flash-resident bootloader operation

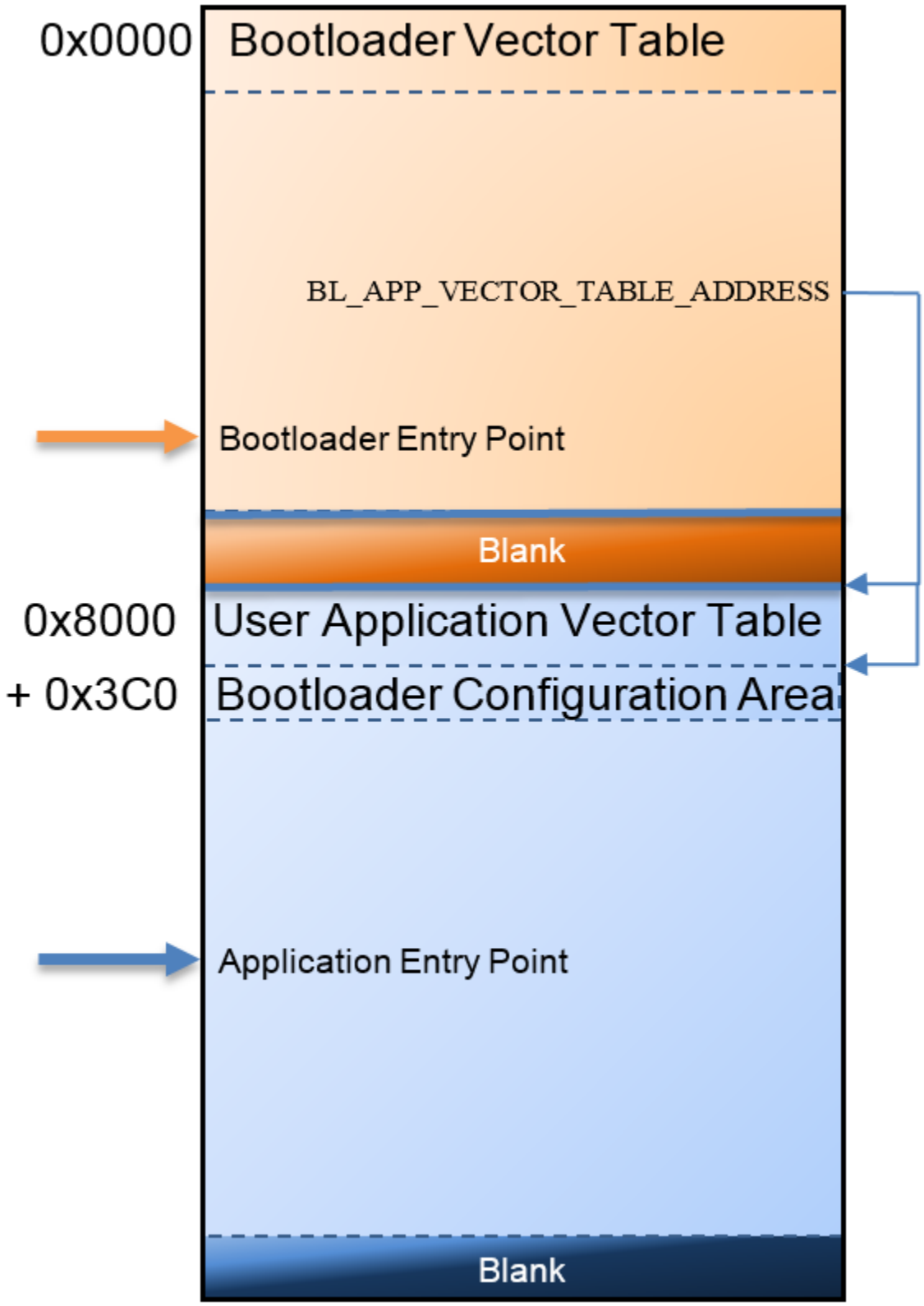
This section describes the linkage between the Kinetis flash-resident bootloader and the user application. The demonstration described above illustrates a fairly simple collaboration between the Kinetis bootloader and the led_demo application. The considerations are:

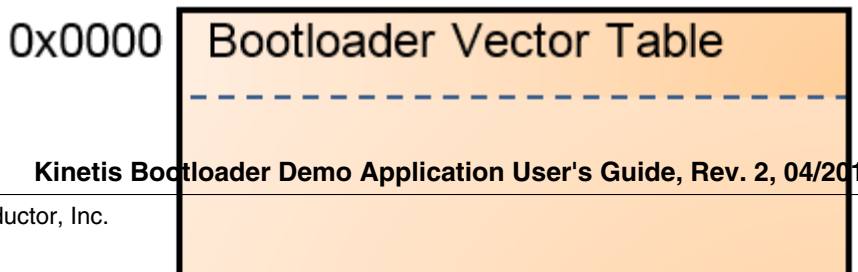
- The flash-resident bootloader is located in flash at address 0.
- The user application is located in flash above the bootloader at BL_APP_VECTOR_TABLE_ADDRESS as defined in `<install_dir>/apps/targets/<mcu>/src/bootloader_config.h`
- The vector table for the User Application must be placed at the beginning 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 the image to the correct base address. In addition, the application vector table must be updated based on the correct application location.

7.1 Memory map overview


0x0000
Bootloader Vector Table
BL_APP_VECTOR_TABLE_ADDRESS
Bootloader Entry Point
Blank
0x8000
User Application Vector Table
+ 0x3C0
Bootloader Configuration Area
Application Entry Point
Blank


0x0000
Bootloader Vector Table

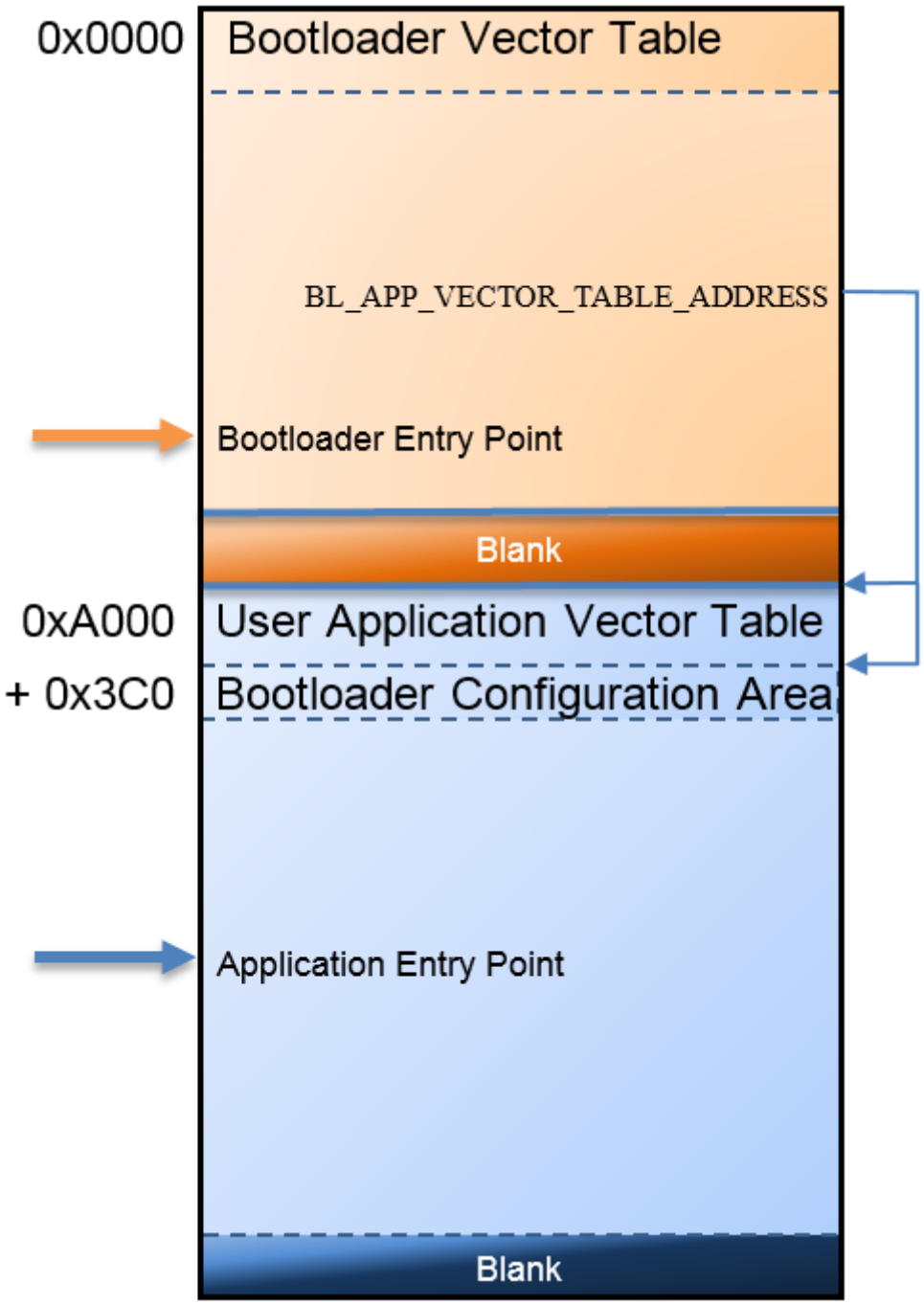


Figure 13. Device memory map

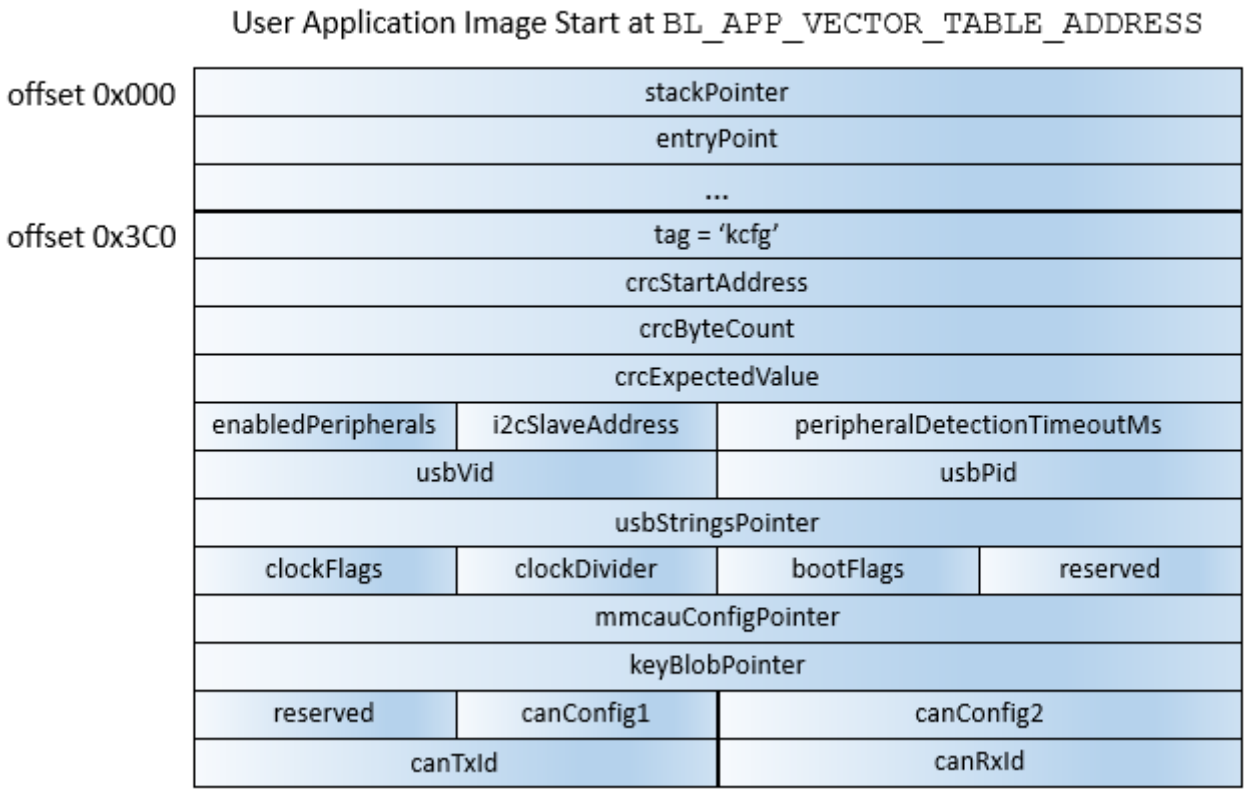


Figure 14. User application vector table and Bootloader Configuration Area (BCA)

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.

8 Appendix B - Kinetis Bootloader Development platforms

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

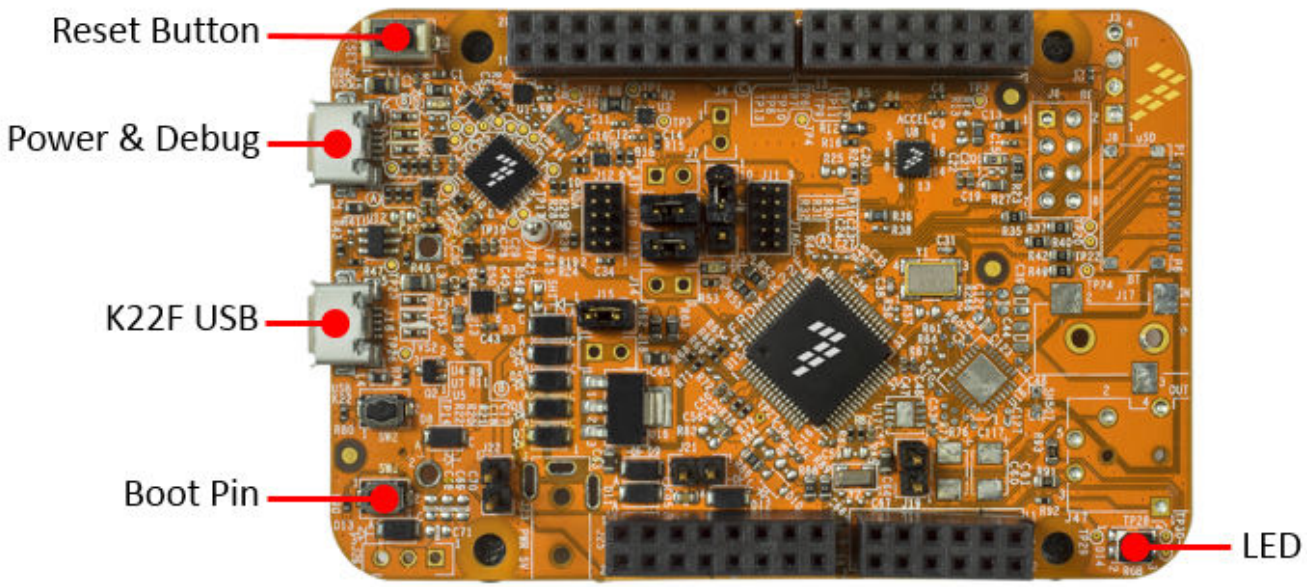


Figure 15. FRDM-K22F platform

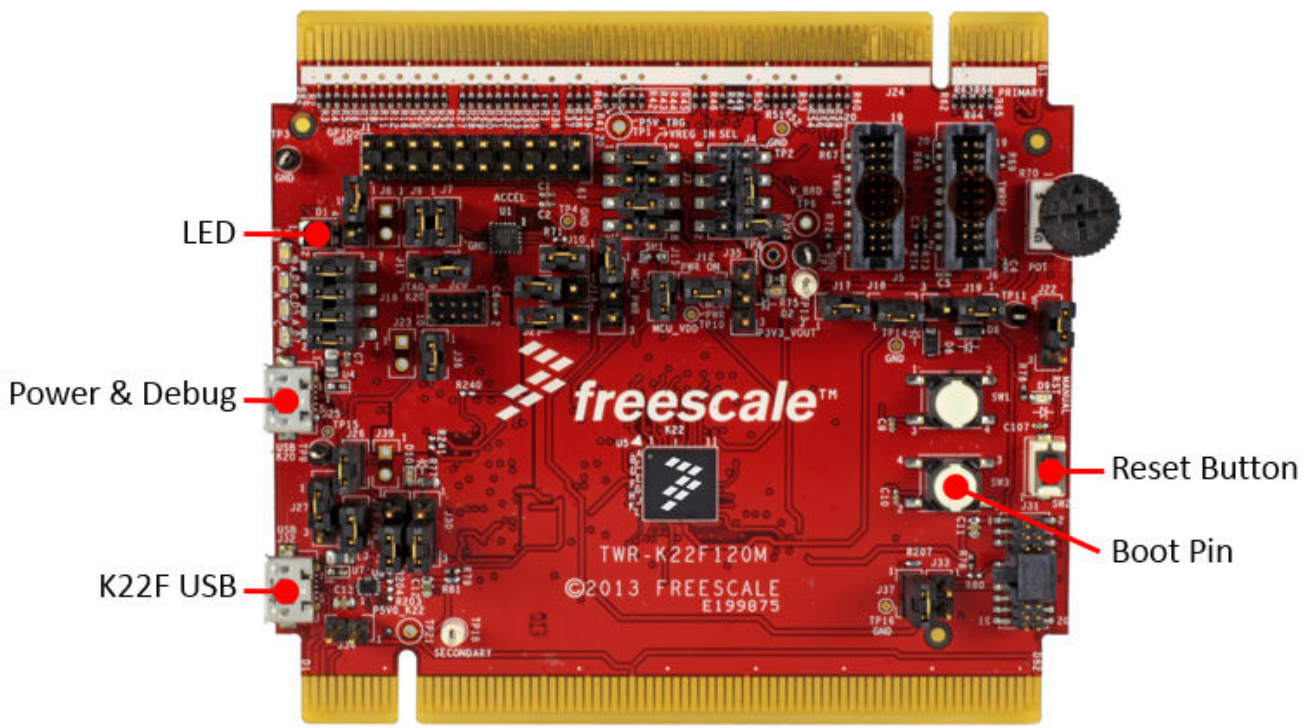


Figure 16. TWR-K22F120M platform

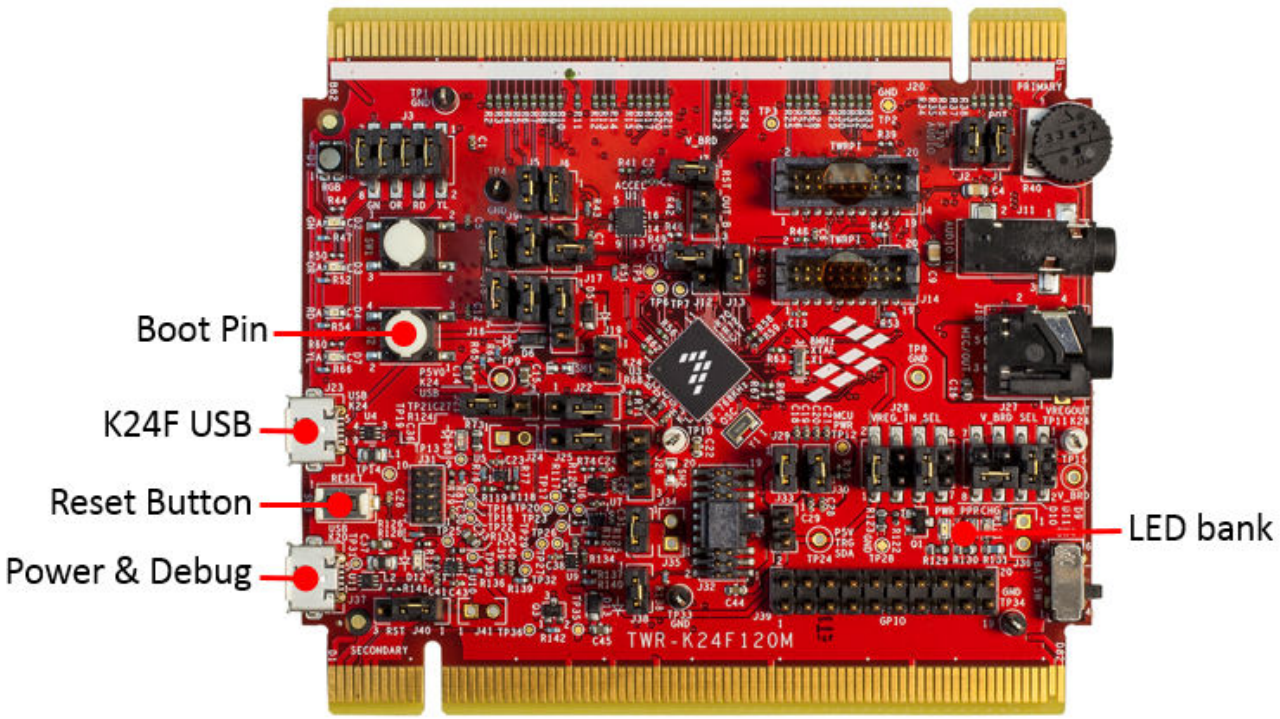


Figure 17. TWR-K24F120M platform

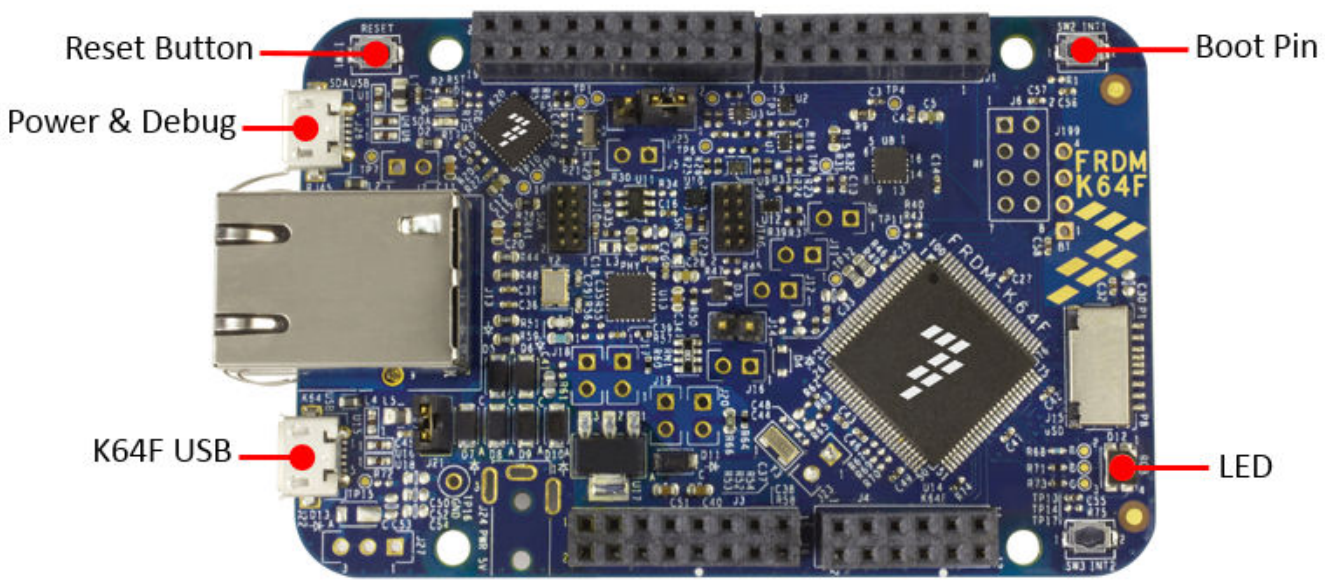


Figure 18. FRDM-K64F platform

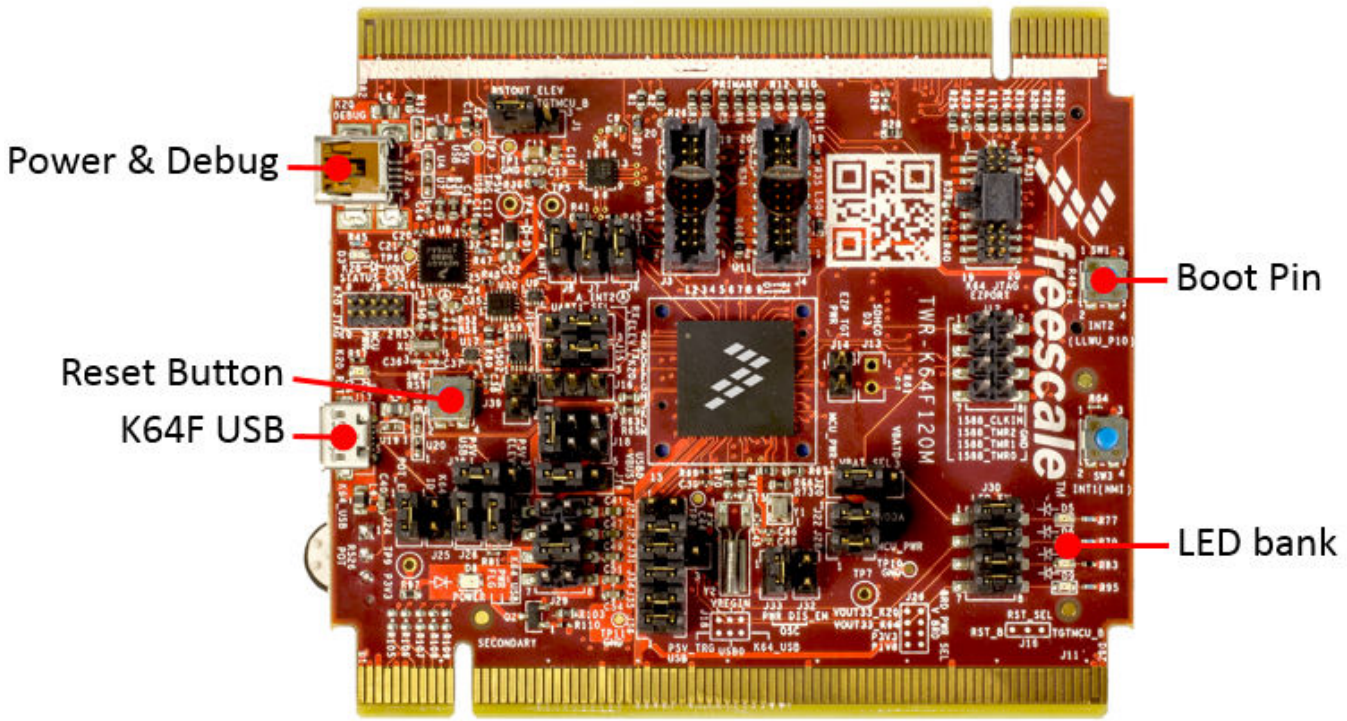


Figure 19. TWR-K64F120M platform

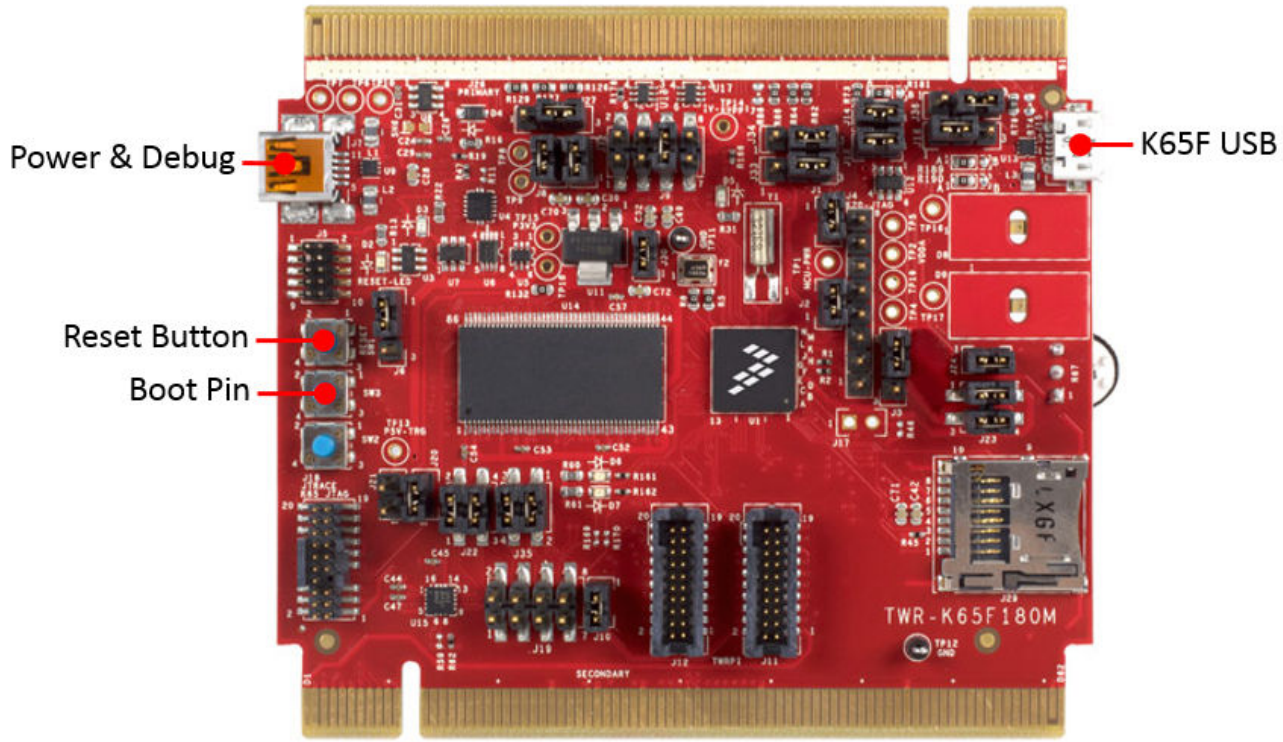


Figure 20. TWR-K65F180M platform

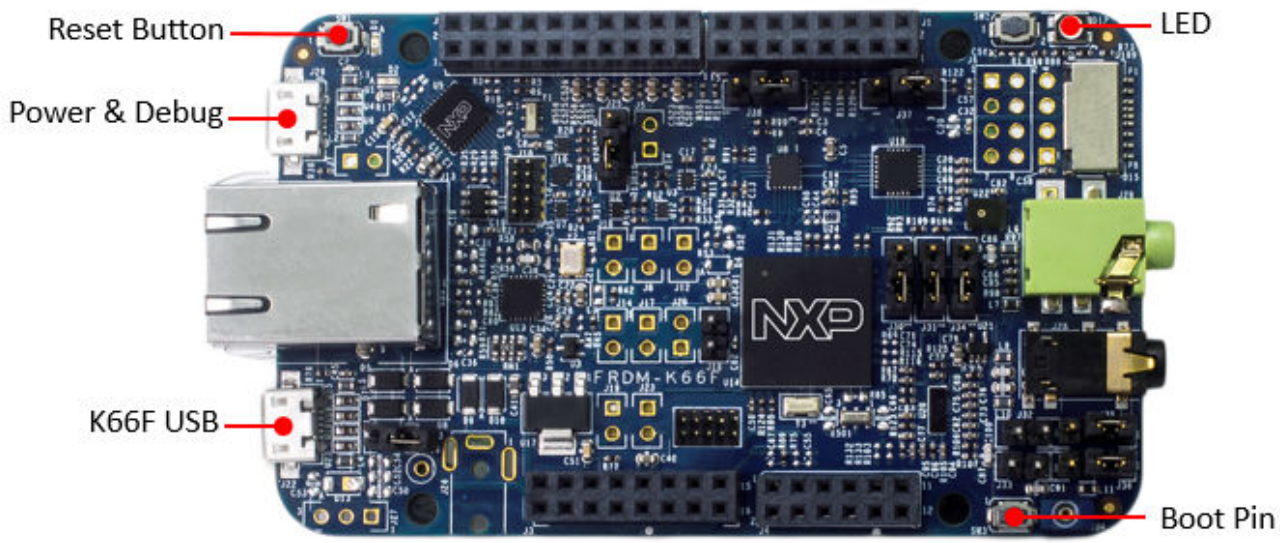


Figure 21. FRDM-K66F platform

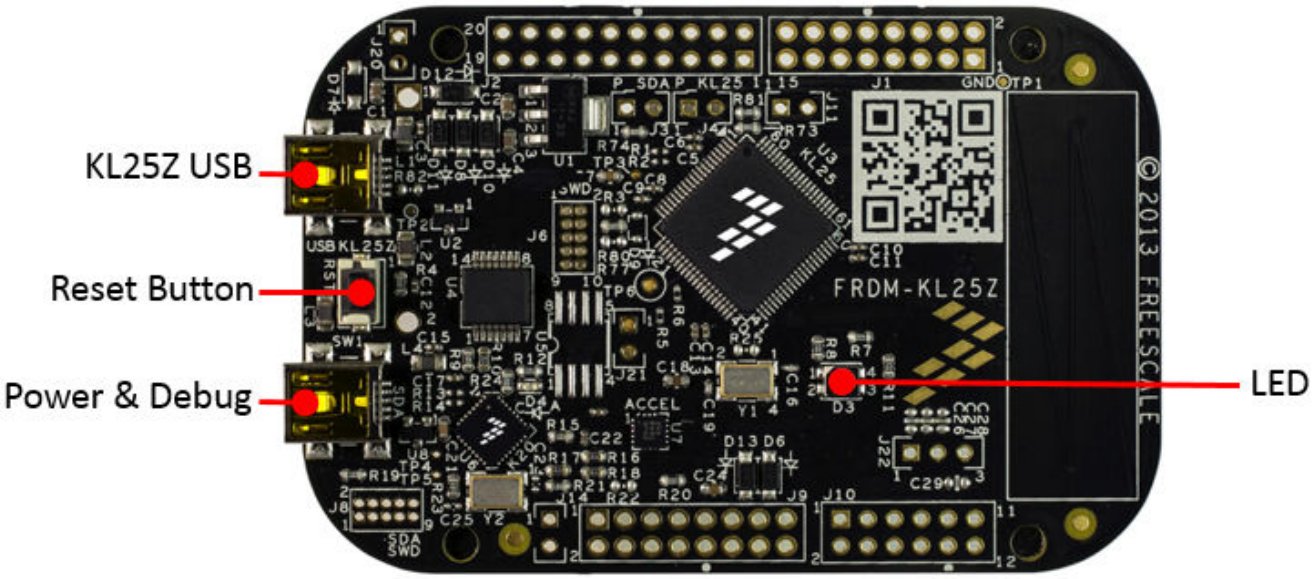


Figure 22. FRDM-KL25Z platform

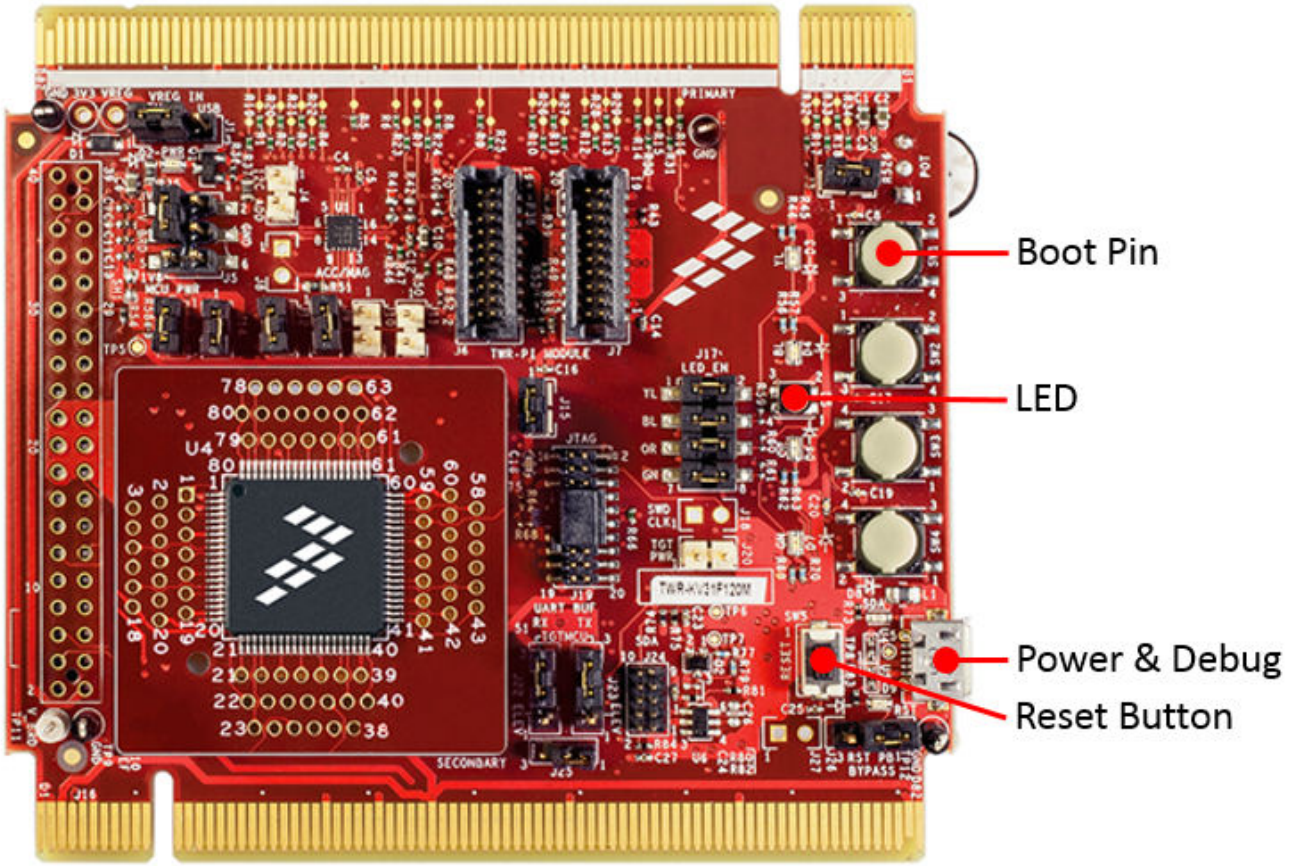


Figure 23. TWR-KV31F120M platform

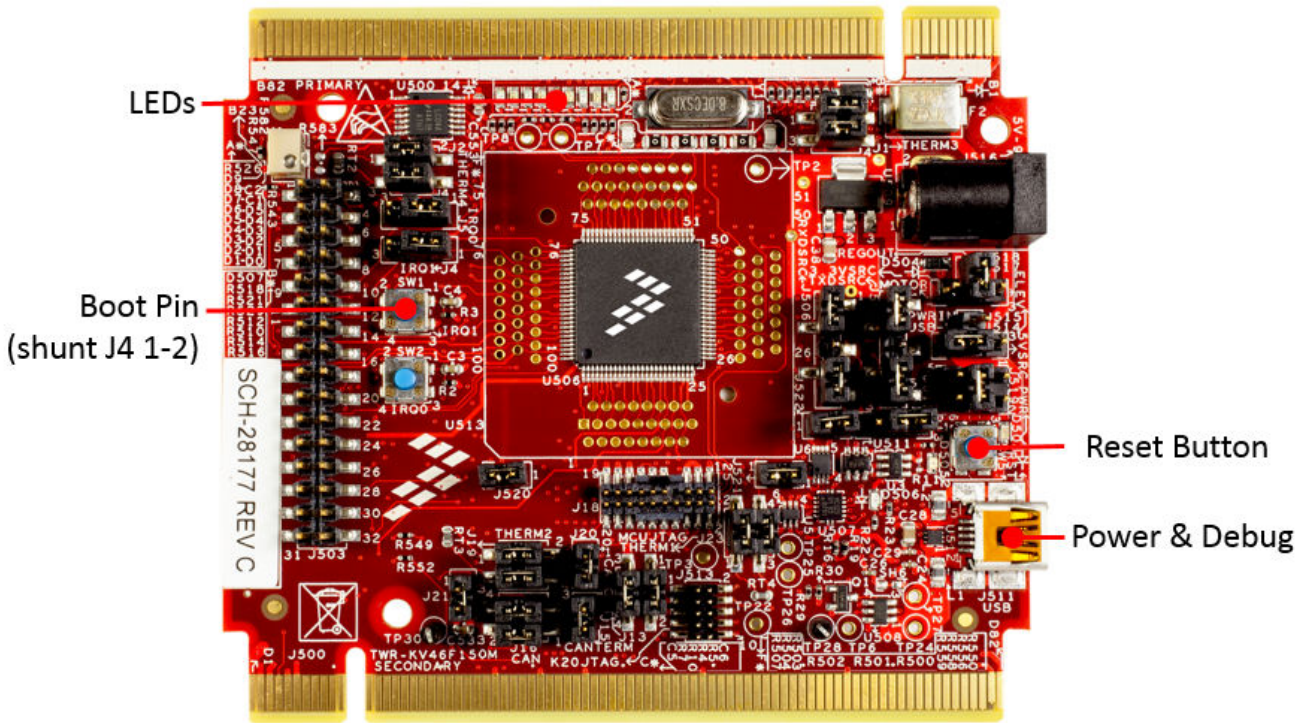


Figure 24. TWR-KV46F150M platform

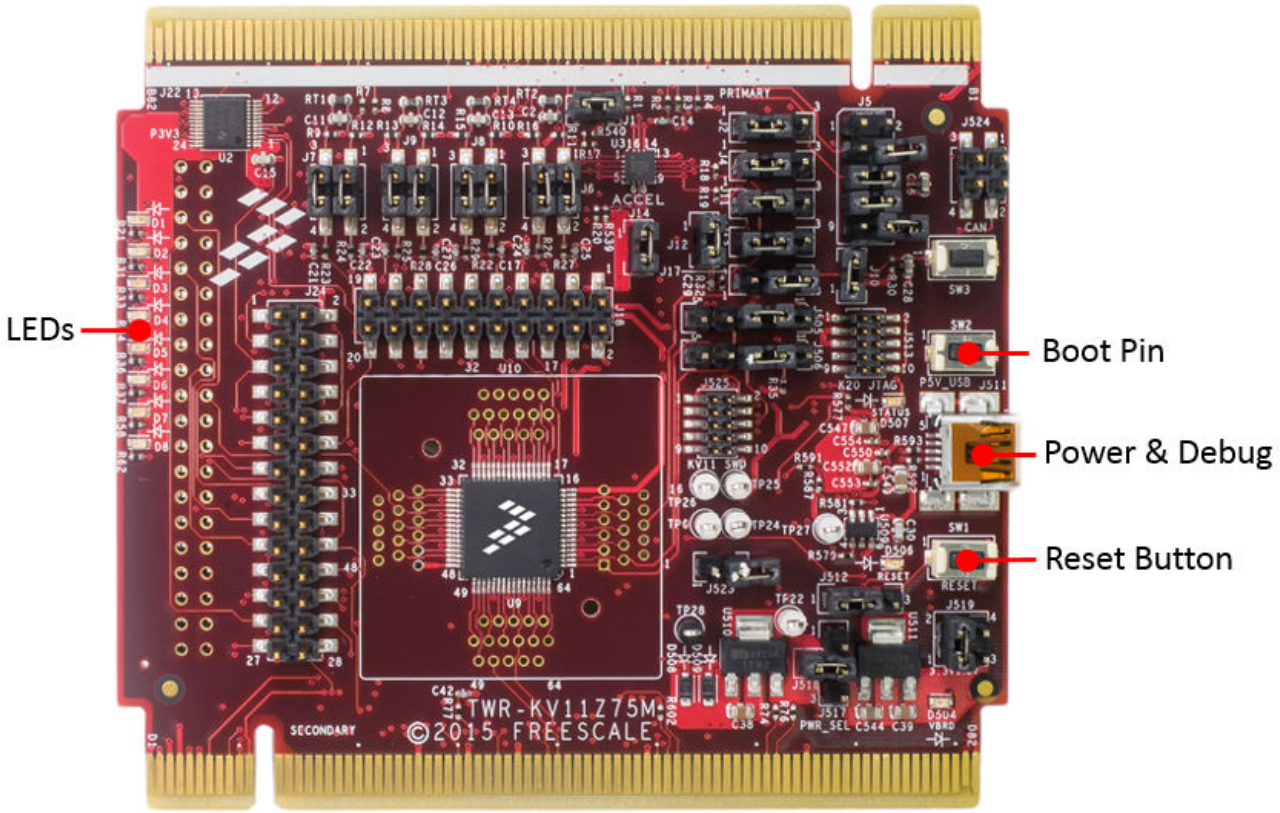


Figure 25. TWR-KV11Z75M platform

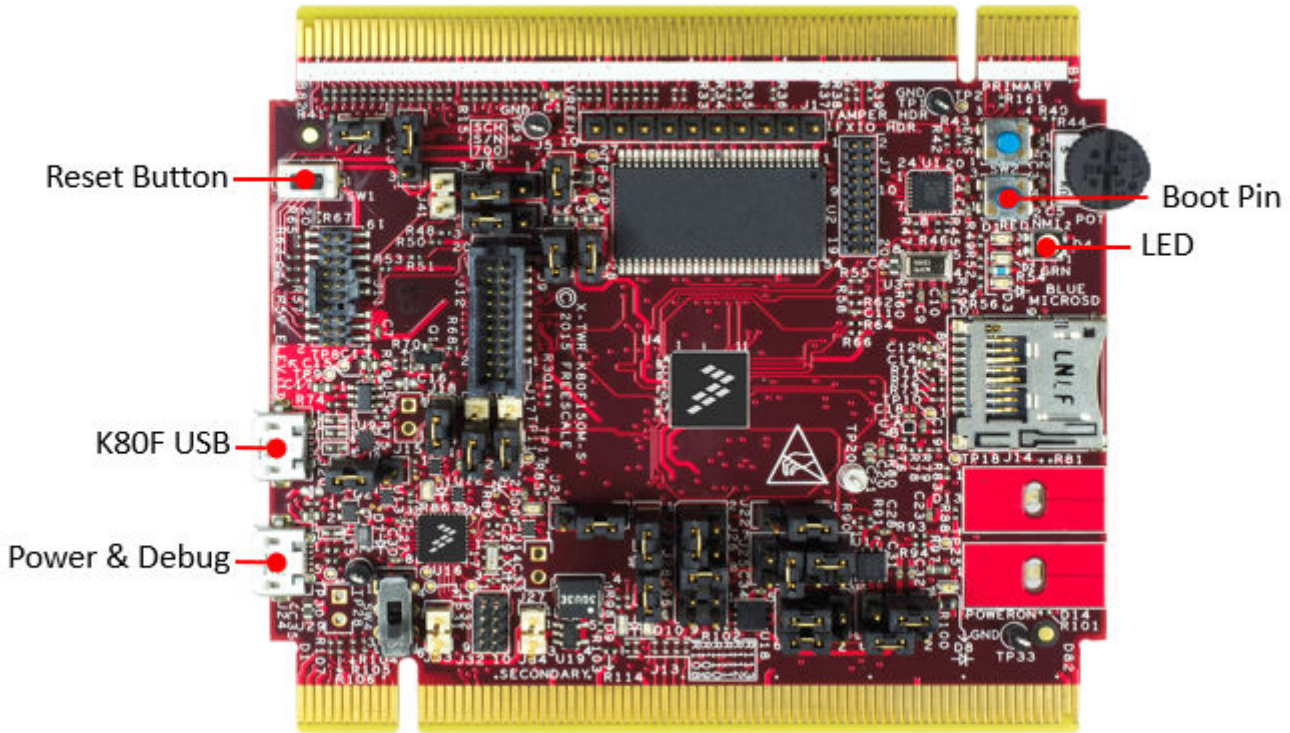


Figure 26. TWR-K80F150M platform

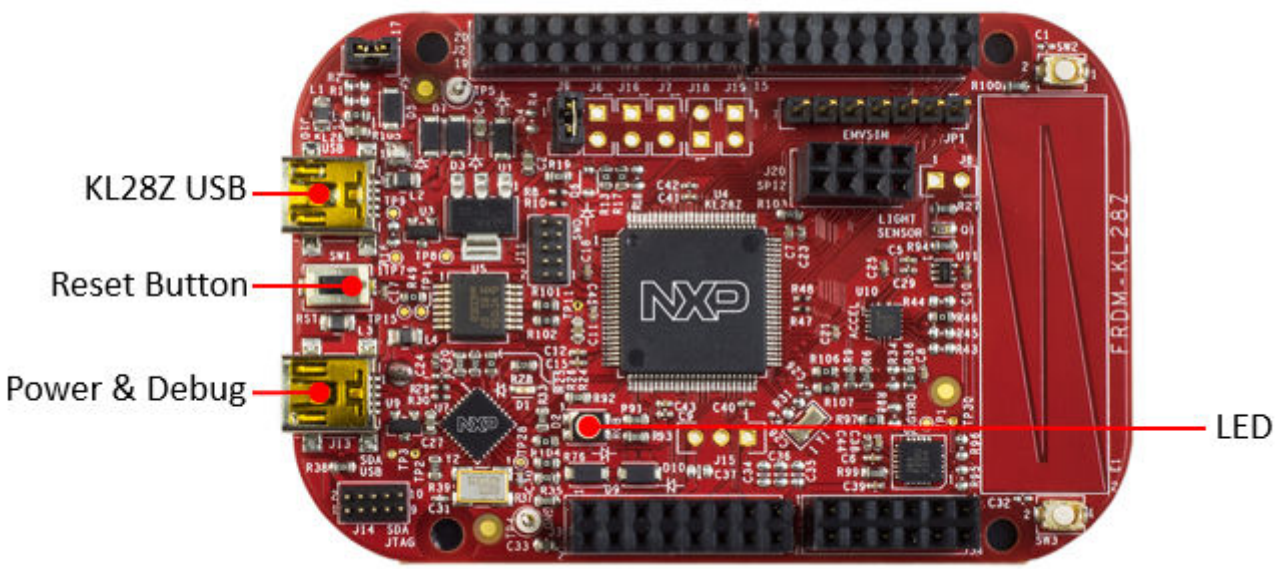


Figure 27. FRDM-KL28Z platform

FRDM-K82F

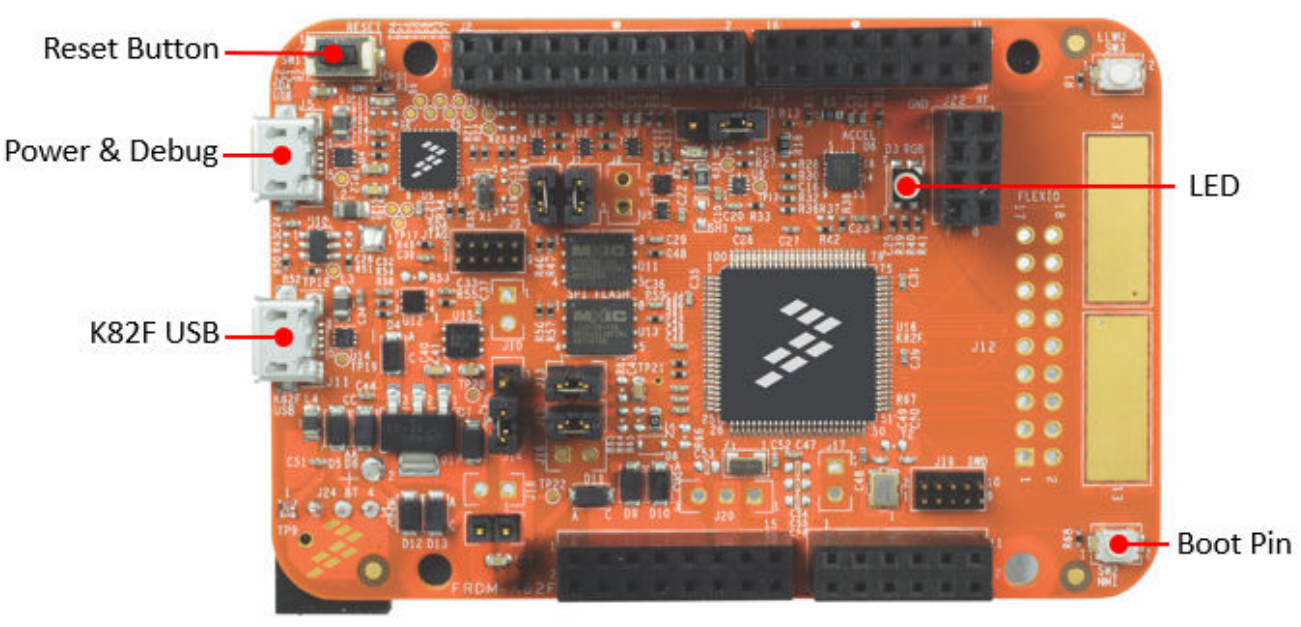


Figure 28. FRDM-K82F platform

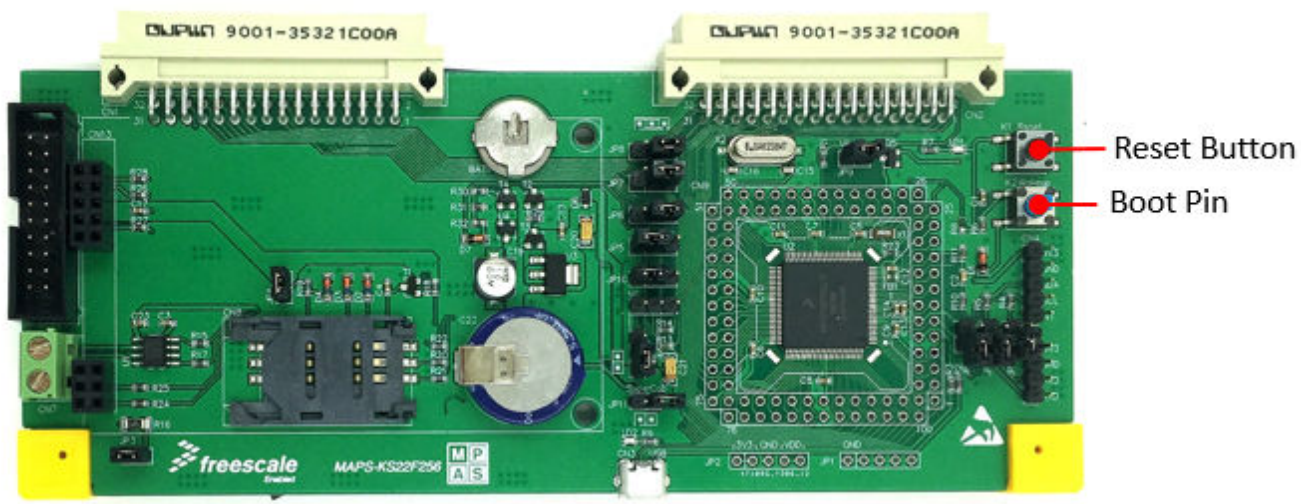


Figure 29. KS22F256 Bootloader MAPS-KS22 platform

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

Table continues on the next page...

Table 3. MKL25Z4 bootloader – FRDM-KL25Z (continued)

Peripheral	Instance	Port	Signal	AltMode	FRDM-KL25Z Test points
		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
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

Table continues on the next page...

Table 5. MK64F12 flashloader – TWR-K64F120M (continued)

Peripheral	Instance	Port	Signal	AltMode	TWR-K64F120M Test points
		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	PTD8	I2C0_SCL	2	J13 pin 2
		PTD9	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	PTD8	I2C0_SCL	2	J13 pin 2
		PTD9	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. MK65F18 bootloader – FRDM-K66F

Peripheral	Instance	Port	Signal	AltMode	FRDM-K66F Test points
UART	0	PTB17	UART0_TX	3	port J26
		PTB16	UART0_RX		
I2C	1	PTC10	I2C1_SCL	2	J4 pin 20
		PTC11	I2C1_SDA		J4 pin 18
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
HS USB	-	-	USB1_DP	Default	USB J22
			USB1_DM		

Table 10. MKV30F12810 bootloader/flashloader – TWR-KV30F100M

Peripheral	Instance	Port	Signal	AltMode	TWR-KV30F100M Test points
UART	1	PTC3	UART1_RX	3	Elev B47
		PTC4	UART1_TX		Elev A37
I2C	0	PTB0	I2C0_SCL	2	Elev A30
		PTB1	I2C0_SDA		Elev B28
SPI	0	PTE16	SPI0_PCS0	2	Elev B46
		PTE17	SPI0_SCK		Elev B48

Table continues on the next page...

Table 10. MKV30F12810 bootloader/flashloader – TWR-KV30F100M (continued)

Peripheral	Instance	Port	Signal	AltMode	TWR-KV30F100M Test points
		PTE18	SPI0_SOUT		Elev B45
		PTE19	SPI0_SIN		Elev B44

Table 11. MK02F12810 bootloader/flashloader – TWR-KV30F100M

Peripheral	Instance	Port	Signal	AltMode	TWR-KV30F100M Test points
UART	1	PTC3	UART1_RX	3	Elev B47
		PTC4	UART1_TX		Elev A37
I2C	0	PTB0	I2C0_SCL	2	Elev A30
		PTB1	I2C0_SDA		Elev B28
SPI	0	PTE16	SPI0_PCS0	2	Elev B46
		PTE17	SPI0_SCK		Elev B48
		PTE18	SPI0_SOUT		Elev B45
		PTE19	SPI0_SIN		Elev B44

Table 12. 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
SPI	0	PTE16	SPI0_PCS0	2	Elev B46
		PTE17	SPI0_SCK		Elev B48
		PTE18	SPI0_SOUT		Elev B45
		PTE19	SPI0_SIN		Elev B44

Table 13. MKV31F512 bootloader – FRDM-KV31F

Peripheral	Instance	Port	Signal	AltMode	TWR-KV31F120M Test points
UART	0	PTB17	UART0_TX	3	OpenSDA port
		PTB16	UART0_RX		
I2C	0	PTB2	I2C0_SCL	2	J2 pin 20
		PTB1	I2C0_SDA		J2 pin 18
SPI	0	PTE16	SPI0_PCS0	2	J1 pin 15
		PTE17	SPI0_SCK		J2 pin 12

Table continues on the next page...

Table 13. MKV31F512 bootloader – FRDM-KV31F (continued)

Peripheral	Instance	Port	Signal	AltMode	TWR-KV31F120M Test points
		PTE18	SPI0_SOUT		J2 pin 8
		PTE19	SPI0_SIN		J2 pin 10

Table 14. 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 15. 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 16. 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

Table 17. KS22F256 Bootloader - MAPS-KS22

Peripheral	Instance	Port	Signal	AltMode	MAPS-KS22F256 Test points	
UART	1	PTE0	UART1_TX	3	M1-5	OpenSDA port on Dock CN14
		PTE1	UART1_RX		M1-6	
I2C	0	PTB0	LPI2C0_SCL	2	CN4-4	
		PTB1	LPI2C0_SDA		CN4-3	
SPI	1	PTD4	SPI1_PCS0	7	CN13-1	
-		PTD5	SPI1_SCK		CN13-3	
-		PTD6	SPI1_SOUT		CN13-30	
-		PTD7	SPI1_SIN		CN13-29	
USB	0	-	USB0_DP	Default	CN3	
-		-	USB0_DM			
FlexCAN	0	PTB18	CAN0_TX	2	CN7-1(CANH)	
-		PTB19	CAN0_RX		CN7-2(CANH)	

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

Table 18. MKV58F22 bootloader - TWR-KV58F220M

Peripheral	Instance	Port	Signal	AltMode	TWR-KV11Z128M Test points
UART	0	PTB1	UART0_TX	7	mbed port J22
		PTB0	UART0_RX		
I2C	0	PTB2	I2C0_SCL	2	J14 pin 7
		PTB3	I2C0_SDA		J14 pin 5
SPI	0	PTE16	SPI0_PCS0	2	Elev B46
		PTE17	SPI0_SCK		Elev B48
		PTE18	SPI0_SOUT		Elev B45
		PTE19	SPI0_SIN		Elev B44
FlexCAN	0	PTB16	CAN0_TX	2	Elev B68
		PTB17	CAN0_RX		Elev B67

Table 19. MKV58F22 flashloader - TWR-KV58F220M

Peripheral	Instance	Port	Signal	AltMode	TWR-KV11Z128M Test points
UART	0	PTD7	UART0_TX	3	Elev A80
		PTD6	UART0_RX		J31 pin 14
I2C	0	PTB0	I2C0_SCL	2	J24 pin 2
		PTB1	I2C0_SDA		J25 pin 2
SPI	0	PTE16	SPI0_PCS0	2	Elev B46
		PTE17	SPI0_SCK		Elev B48
		PTE18	SPI0_SOUT		Elev B45
		PTE19	SPI0_SIN		Elev B44
FlexCAN	0	PTB16	CAN0_TX	2	Elev B68
		PTB17	CAN0_RX		Elev B67

NOTE

CAN connection - Option 1, use the CAN transceiver on the board

1. CAN0_TX, Elev B68 -> TWR-SER J5 pin 8 , TWR-SER J7 pin 1 -> BusPal (KV46) J13 pin 2
2. CAN0_RX, Elev B67 -> TWR-SER J5 pin 6 , TWR-SER J7 pin 3 -> BusPal (KV46) J13 pin 1

Bootloader for ROMs:

- Define macros to use ROM bootloader pins for all devices having ROM bootloader:

  #define BL_FEATURE_ROM_UART_PORT (0)

  #define BL_FEATURE_ROM_I2C_PORT (0)

  #define BL_FEATURE_ROM_SPI_PORT (0)

- User ROM boot pin - PA4

Table 20. MK80F256 bootloader/flashloader - TWR-K80F150M

Peripheral	Instance	Port	Signal	AltMode	TWR-K80F150M Test points
LPUART	1	PTC3	UART1_RX	3	Mbed port J24
		PTC4	UART1_TX		
I2C	1	PTC10	I2C1_SCL	2	Elev B50
		PTC11	I2C1_SDA		Elev B51
SPI	1	PTD4	SPI1_PCS0	7	Elev A78
		PTD5	SPI1_SCK		Elev A79
		PTD6	SPI1_SOUT		Elev A80
		PTD7	SPI1_SIN		Elev A56
USB	0	-	USB0_DP	Default	J19
		-	USB0_DM		

Table 21. MK80F256 bootloader/flashloader - FRDM-K82F

Peripheral	Instance	Port	Signal	AltMode	FRDM-K82F Test points
LPUART	1	PTC14	UART4_RX	2	Mbed port J10
		PTC15	UART4_TX		
I2C	1	PTC10	I2C1_SCL	2	J1 pin 12
		PTC11	I2C1_SDA		J1 pin 10
SPI	1	PTD4	SPI1_PCS0	7	J2 pin 6
		PTD5	SPI1_SCK		J22 pin 5
		PTD6	SPI1_SOUT		J22 pin 6
		PTD7	SPI1_SIN		J22 pin 7
USB	0	-	USB0_DP	Default	J11
		-	USB0_DM		

Table 22. MKL28Z7 bootloader - TWR-KL28Z72M

Peripheral	Instance	Port	Signal	AltMode	TWR-KL28Z72M Test points

Table continues on the next page...

Table 22. MKL28Z7 bootloader - TWR-KL28Z72M (continued)

LPUART	0	PTA1	UART0_RX	2	Mbedport J10
		PTA2	UART0_TX		
LPI2C	1	PTE1	I2C1_SCL	6	Elev B45
		PTE0	I2C1_SDA		Elev A35
LPSPI	2	PTB20	SPI2_PCS0	7	Elev B9
		PTB21	SPI2_SCK		Elev B7
		PTB22	SPI2_SOUT		Elev B10
		PTB23	SPI2_SIN		Elev B11
USB	0	-	USB0_DP	Default	J29
		-	USB0_DM		

Table 23. MKL28Z7 bootloader - FRDM-KL28Z

Peripheral	Instance	Port	Signal	AltMode	TWR-KL28Z72M Test points
LPUART	0	PTB16	UART0_RX	3	Mbed port J13
		PTB17	UART0_TX		
LPI2C	1	PTC1	I2C1_SCL	2	J4 pin 12
		PTC2	I2C1_SDA		J4 pin 10
LPSPI	2	PTB20	SPI2_PCS0	7	J20 pin 4
		PTB21	SPI2_SCK		J20 pin 5
		PTB22	SPI2_SOUT		J20 pin 6
		PTB23	SPI2_SIN		J20 pin 7
USB	0	-	USB0_DP	Default	J10
		-	USB0_DM		

Table 24. MKL81Z7 bootloader - TWR-KL82Z72M

Peripheral	Instance	Port	Signal	AltMode	TWR-KL82Z72M Test points
LPUART	1	PTC3	UART1_RX	3	Mbed port J24
		PTC4	UART1_TX		
I2C	1	PTC10	I2C1_SCL	2	Elev B50
		PTC11	I2C1_SDA		Elev B51
SPI	1	PTD4	SPI1_PCS0	7	Elev A78
		PTD5	SPI1_SCK		Elev A79
		PTD6	SPI1_SOUT		Elev A80
		PTD7	SPI1_SIN		Elev A56
USB	0	-	USB0_DP	Default	J11
		-	USB0_DM		

Table 25. MKL82Z7 bootloader - FRDM-KL82Z

Peripheral	Instance	Port	Signal	AltMode	FRDM-K82F Test points
LPUART	0	PTB16	UART0_RX	3	Mbed port J5
		PTB17	UART0_TX		
I2C	1	PTC10	I2C1_SCL	2	J1 pin 20
		PTC11	I2C1_SDA		J1 pin 18
SPI	1	PTD4	SPI1_PCS0	7	J22 pin 4
		PTD5	SPI1_SCK		J22 pin 5
		PTD6	SPI1_SOUT		J22 pin 6
		PTD7	SPI1_SIN		J22 pin 7
USB	0	-	USB0_DP	Default	J11
		-	USB0_DM		

Table 26. MKL82Z7 flashloader – FRDM-KL82Z/TWR-KL82Z72M

Peripheral	Instance	Port	Signal	AltMode	FRDM-KL82Z Test points	TWR-KL82Z72M Test points
LPUART	0	PTB16	LPUART0_RX	3	Mbed port J5	N/A
		PTB17	LPUART0_TX			
LPUART	1	PTC3	LPUART1_RX	3	J1 pin 14	Mbed port J24
		PTC4	LPUART1_TX		J2 pin 6	
LPUART	2	PTD2	LPUART2_RX	3	J12 pin 3	Elev A76
		PTD3	LPUART2_TX		J12 pin 4	Elev A77
I2C	1	PTC10	I2C1_SCL	2	J1 pin 20	Elev B50
		PTC11	I2C1_SDA		J1 pin 18	Elev B51
SPI	1	PTD4	SPI1_PCS0	7	J22 pin 4	Elev A78
		PTD5	SPI1_SCK		J22 pin 5	Elev A79
		PTD6	SPI1_SOUT		J22 pin 6	Elev A80
		PTD7	SPI1_SIN		J22 pin 7	Elev A56
USB	0	-	USB0_DP	Default	J11	J11
		-	USB0_DM			

10 Revision history

This table summarizes revisions to this document.

Table 27. Revision history

Revision number	Date	Substantive changes
0	07/2015	Kinetis Bootloader 1.2.0 initial release
1	12/2015	Updates for standalone Kinetis KS22F256 bootloader v1.0.0 based on Kinetis bootloader v1.2.0 initial release.
2	04/2016	Kinetis Bootloader v.2.0.0 release

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 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: nxp.com/SalesTermsandConditions.

Document Number: KBTLDRDEMOUG
Rev. 2
04/2016