

Kinetis KS22F256 Bootloader v1.0.0 based on Kinetis Bootloader v1.2.0 Release Notes

1 Overview

These are the release notes for the Standalone Kinetis KS22F256 bootloader v1.0.0 based on the Kinetis bootloader v1.2.0 release. This release of the bootloader supports the MAPS-KS22F256 development platform. For more information and getting started instructions, see the Getting Started section of this document.

The Kinetis bootloader is an application that you program into the internal flash memory of a Kinetis device. The bootloader is designed to detect communication traffic on one of the supported peripherals (USB-HID, UART, SPI, I2C, and CAN), download a user application, and write the application to internal flash. The bootloader stays resident on flash along with your application.

This release includes the PC-hosted Kinetis Updater application. This application allows you to choose a device application image and send it to the bootloader over USB-HID or UART.

2 Development tools

The Kinetis KS22F256 bootloader release was compiled and tested with these development tools.

Firmware projects:

© 2015 Freescale Semiconductor, Inc.

Contents

1	Overview.....	1
2	Development tools.....	1
3	System requirements.....	2
4	Target requirements.....	2
5	Release contents.....	2
6	Getting started.....	3
7	Features.....	3
8	Host tools.....	3
9	New features.....	4
10	Fixed issues.....	4
11	Known issues.....	4
12	Tool notes.....	4
13	Revision history.....	5

System requirements

- Kinetis Design Studio (KDS) IDE v.3.0.0
- IAR Embedded Workbench for ARM® v7.50.1

Host projects:

- Microsoft® Visual Studio Professional 2013 for Windows® OS Desktop
- Microsoft NET Framework v4.5 (included in Windows OS 8)
- Microsoft Visual Studio C++ Redistributable for Visual Studio 2013 (vcredist_x86.exe)
- Python v2.7 (www.python.org)

NOTE

The Python path must be added to the system environment path. For example, C:\Python27.

- Apple Xcode® v6.3.1 (for the blhost tool)
- GNU Compiler (GCC) v4.8.1 (for the blhost tool), libstdc++6, libudev-dev, libc6, and libgcc1 (for the blhost tool)

3 System requirements

System requirements are based on the requirements for the development tools and the Kinetis Updater application.

The recommended PC configuration is 2 GHz processor, 2 GB RAM, and 2 GB free disk space.

4 Target requirements

This release of the Kinetis bootloader supports the following platforms:

- MAPS-KS22F256 development platform

There are no special requirements for the hardware other than what the board requires to operate.

5 Release contents

This table describes the release contents.

Table 1. Release contents

Deliverable	Location
Host updater and demo applications	<install_dir>/apps/...
Host binaries and utilities	<install_dir>/bin/...
Documentation	<install_dir>/doc/...
Bootloader and host tools source code	<install_dir>/src/...
Tool chain build projects	<install_dir>/targets/...

6 Getting started

See the *Demo Applications User's Guide for the Freescale Platforms* (document KBTLDRDEMOUG) for a description of how to use the Kinetis bootloader to load a user application on the Kinetis MCUs.

See the *Getting Started with Kinetis ROM Bootloader* (document KBTLDRUG) for Kinetis ROM-specific information.

See the *Getting Started with Kinetis Flashloader* (document KFLLDRUG) for Kinetis Flashloader-specific information.

For porting information, see Chapter 10, "Kinetis bootloader porting" in the *Kinetis Bootloader v1.2.0 Reference Manual* (document KBTLDR120RM).

For customization, see Chapter 11, "Create a custom flash-resident bootloader" in the *Kinetis Bootloader v1.2.0 Reference Manual* (document KBTLDR120RM).

7 Features

The bootloader release contains source code and toolchain projects for building flash-resident bootloaders and flashloaders for the supported platforms (see Section 4, "Target Requirements"). A flash-resident bootloader stays resident in flash along with the user application. The flash-resident bootloader can be used to download and program an initial application image into a blank area on the flash, and to later update the application. In contrast, a flashloader gets replaced in flash by the user application and thus is a one-time programming aid.

The Kinetis bootloader supports the following communication interfaces for downloading an application. Not all interfaces are supported on all platforms. See the individual platform reference manual for supported interfaces.

- USB-HID
- UART
- I2C
- SPI
- CAN

Usually, USB-HID and UART connections are made directly to a PC, whereas I2C and SPI require additional hardware. The bootloader, running on the target platform, acts as a communication slave. The bootloader can automatically detect which peripheral is being used to download the application and, in the case of UART, automatically detect the baud rate.

The application image is downloaded to the target through a series of command and data packets sent from a host PC or embedded host platform.

8 Host tools

The bootloader release contains source code and build projects for the following PC-based host tools:

- blhost - command line debug tool to send individual commands to the bootloader.
- Kinetis Updater - GUI application to download and flash an application image.

For more information, see the *Kinetis blhost User's Guide* (document KBLHOSTUG) and *Kinetis Updater User's Guide* (document KUPDTRUG).

9 New features

The following new features were introduced in this release:

- Addition of new supported platforms as indicated in Section 4, “Target Requirements”.
- Addition of BusPal example to demonstrate I2C and SPI communication.
- Support for the latest version of IAR toolchain.
- Support for KDS IDE toolchain.
- Addition of Flash Driver API example.
- Support for the blhost application on Windows® OS XP, Linux® OS, and Mac® OS X.
- Support of CAN peripheral.

10 Fixed issues

The following issues were fixed since the previous release:

- The blhost.exe application now issues a warning message if terminated with Ctrl + C.
- The BusPal example has been added to demonstrate I2C and SPI communication.

11 Known issues

The following are known issues with this release:

- Compiler warning messages for the KDS tool chain builds have been disabled. The warnings will be fixed in a future release.
- The Mac OS build of the blhost tool has not been validated with the UART interface. Use the USB interface.

12 Tool notes

- When changing an IAR project to generate additional output (Options->Output Converter->Generate additional output), the output filename extension (Linker->Output->Output filename) must be changed to ‘.out’.
- The ReceiveSBFile bootloader command has been removed from the *Kinetis Bootloader v1.2.0 Reference Manual* (document KBTLDLDR120RM). It will be fully supported in an upcoming release.
- The blhost tool accepts any speed setting in the “--buspal” option for the SPI and I2C peripherals. However, the maximum effective speed settings when using the BusPal example are approximately 300 kHz for I2C and 8000 kHz for SPI.
- When using KDS, load projects using File->Import->Existing Projects into Workspace. Do not use File->Import->Projects of Projects.

Each bootloader target supports one or more of the following project types:

1. maps bootloader – bootloader designed to execute from target flash memory on the MAPS platform.
2. flashloader – bootloader designed to execute from target RAM memory on either the Freedom, Tower, or MAPS platform.

3. flashloader_loader – bootstrap loader designed to execute from flash memory on either the Freedom or Tower platform. This loader copies an image of the flashloader into RAM, then executes the flashloader from RAM.

The flashloader_loader project uses the output of the flashloader build to create the flashloader image to load into RAM. For this reason, the flashloader project must be built before building the flashloader_loader project.

13 Revision history

The following table contains a history of changes made to this document.

Table 2. Revision history

Revision number	Date	Substantive changes
0	12/2015	Standalone Kinetis KS22F256 bootloader v1.0.0 based on Kinetis bootloader v1.2.0 initial 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 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. Tower is a trademark of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. All rights reserved.

© 2015 Freescale Semiconductor, Inc.

Document Number KBTLD RKS22F256RN
Revision 0, 12/2015

