

# Freescale KSDK USB Stack Porting New Platform User's Guide

## 1 Read Me First

This document provides the detailed steps to port the USB Unified Stack to a new platform. There are two main steps for porting:

- Porting the USB Stack
- Porting examples

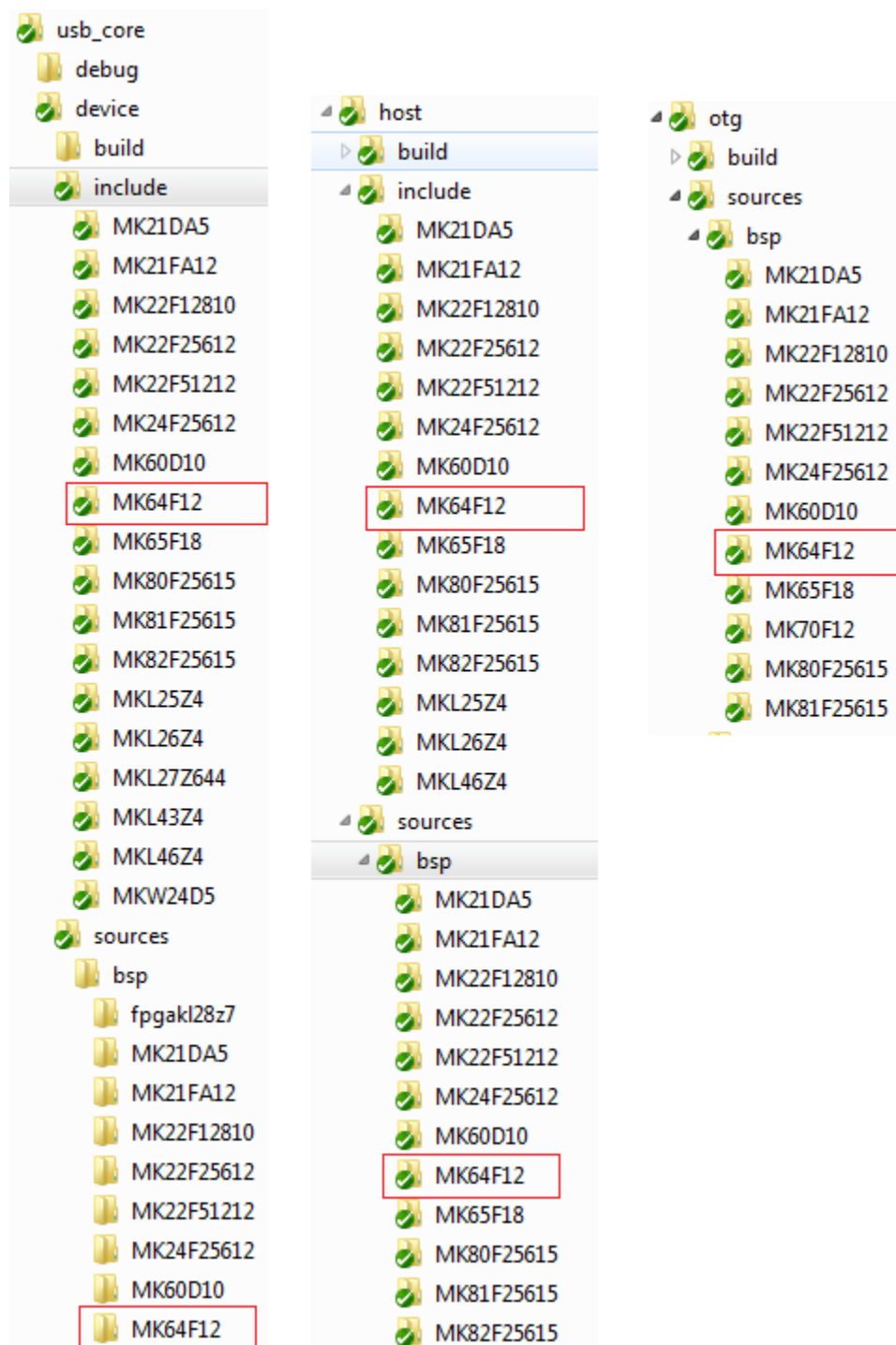
### Contents

1	Read Me First	1
2	Porting the USB Stack	2
3	Porting Examples	6
4	Revision history	8

## 2 Porting the USB Stack

### 2.1 Porting USB driver

The following figure shows the folder structure and all the files that need to be ported after this step. The folders marked with red borders contain the files that need to be ported. The directory of usb\_core is .../KSDK\_ROOT/usb/usb\_core.



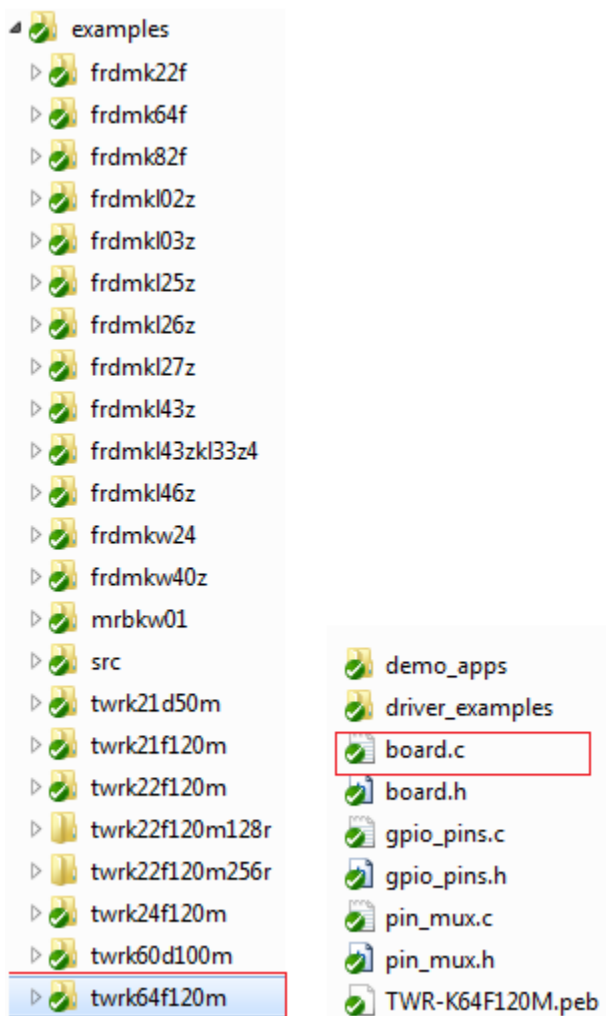


Figure 1 USB directory

### 2.1.1 Device

- BSP code
  - Port `/usb/usb_core/device/sources/bsp/{SOC_NAME}/usb_dev_bsp.c`. This file contains the functions:
    - `usb_dev_soc_init ()`: initializes the register and generates 48MHz clock for the USB module.
  - Port `<install_dir>/examples/BOARD_NAME/board.c`. This file contains the board-specific function for USB:
    - `usb_device_board_init()`: initialize the board-specific settings.
- Configuration file

- Defines the configuration for this platform in  
`/usb/usb_core/device/include/{SOC_NAME}/usb_device_config.h.`

## 2.1.2 Host

- BSP code
  - Port `/usb/usb_core/host/sources/bsp/{SOC_NAME}/usb_host_bsp.c`. This file contains the functions:
    - `usb_host_soc_init ()`: initializes the register and generate the 48 MHz clock for the USB module
  - Port `<install_dir>/examples/BOARD_NAME /board.c`. This file contains the board-specific function for USB:
    - `usb_host_board_init()`: initialize the board-specific settings.
- Configuration file
  - Define the configuration for this platform in  
`/usb/usb_core/host/include/{SOC_NAME}/usb_host_config.h`

## 2.1.3 OTG

- BSP code
  - Port `/usb/usb_core/otg/sources/bsp/{OTG_NAME}/usb_otg_bsp.c`. This file contains the functions:
    - `usb_otg_soc_init ()`: initializes the register and generate the 48 MHz clock for the USB module
  - Port `<install_dir>/examples/BOARD_NAME /board.c`. This file contains the board-specific function for USB:
    - `usb_otg_board_init()`: initialize the board-specific settings.
- Configuration file
  - Define the configuration for this platform in  
`/usb/usb_core/otg/include/{SOC_NAME}/usb_otg_config.h`

## 2.2 Creating a library project

Create a new library project for the new platform. For example, the library file for twrk64f120m is kept in:

- `<install_dir>/usb/usb_core/device/lib/bm/iar/MK64F12`  
`<install_dir>/usb/usb_core/host/lib/bm/iar/MK64F12<install_dir>/usb/usb_core/otg/lib/bm/iar/MK64F12`

## 3 Porting Examples

### 3.1 Creating an example project

List all the examples for each IDE that you want to port for new platform. The project file will be kept in the following folder structure:

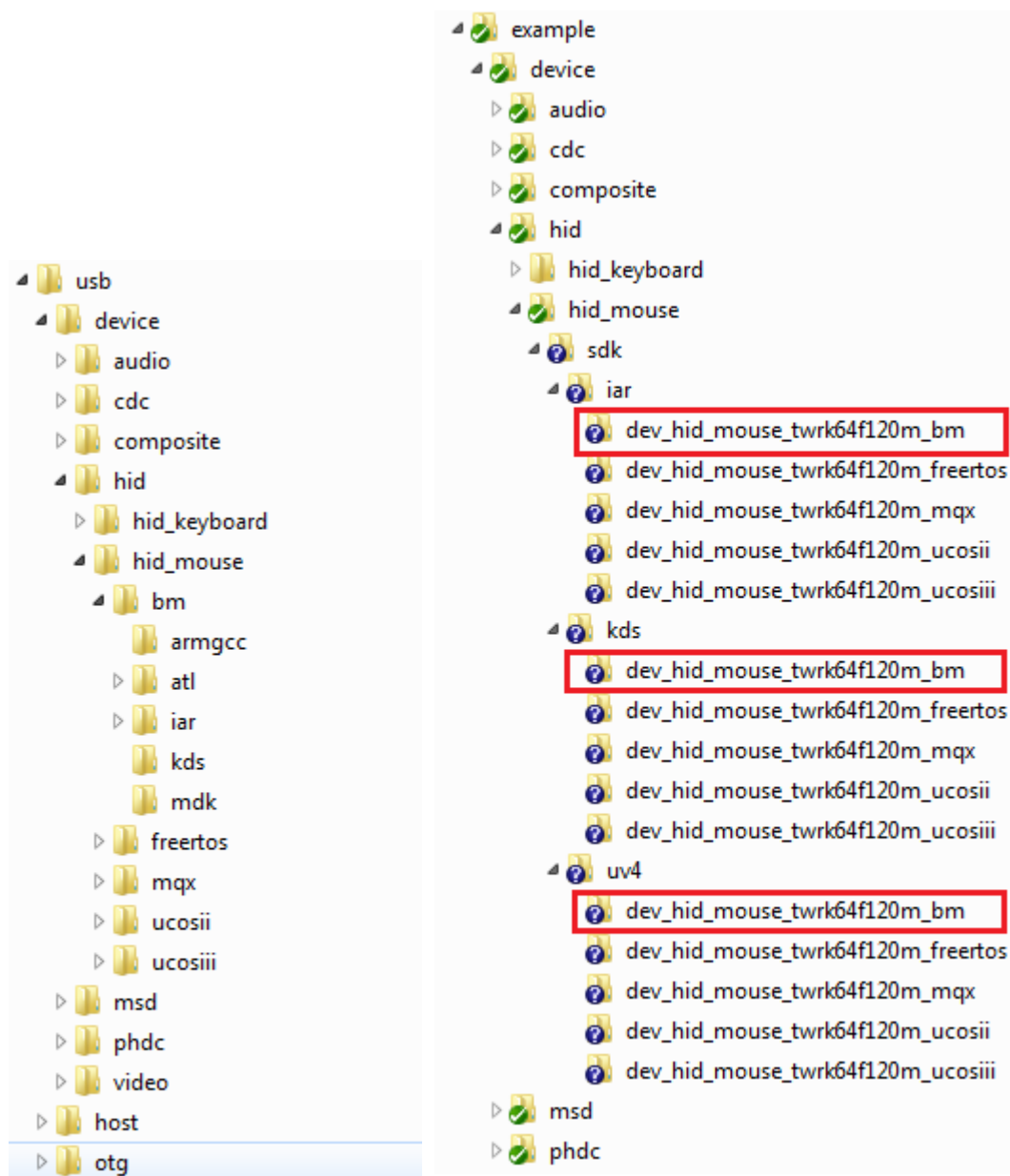



Figure 2 A structure for an example project

The steps for creating a new project are as follows:

1. Create a new empty project.

- 
2. Choose a device.
  3. Add all the source code into project based on the existing platform.
  4. Change the linker file.
  5. Change the include path.
  6. Change the debug configuration.

## 4 Revision history

This table summarizes revisions to this document since the release of the previous version

Revision History		
Revision number	Date	Substantive changes
1	04/2015	KSDK 1.2.0 Release
2	09/2015	Updated Section 2



**How to Reach Us:**

**Home Page:**

[www.freescale.com](http://www.freescale.com)

**Web Support:**

[www.freescale.com/support](http://www.freescale.com/support)

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

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

Freescale, Kinetis, and the Freescale logo are trademarks of Freescale Semiconductor, Inc., Reg. U.S. Pat. & Tm. Off. All other product or service names are the property of their respective owners.

©2015 Freescale Semiconductor, Inc.