## **Freescale Semiconductor**

Document Number: USBTWRK24F120MUG Rev. 1.0, 08/2014

User's Guide

# Freescale MQX USB Stack for TWR-K24F120M GA User's Guide

## 1 Read Me First

This document describes how to compile the USB stack and examples, download a binary image, and run the examples. This document also provides the board-specific information related to TWR-K24F120M.

#### Contents

1	Read Me First	1
2	Requirements for Building USB Examples	2
3	USB Code Structure	4
4	Compiling or Running the USB Stack and Example	s 5
5	USB Stack Configuration	19





© Freescale Semiconductor, Inc., 2014. All rights reserved.

## 2 Requirements for Building USB Examples

#### 2.1 Hardware

- TWR-K24F120M board
- J-Link debugger
- USB cables

#### 2.2 Software

- Freescale MQX<sup>TM</sup> RTOS for the TWR-K24F120M release package
- IAR Embedded Workbench for ARM Version 6.70.1, available for Kinetis devices
- Keil μVision5 Integrated Development Environment Version 5.0.5.15, available for Kinetis ARM<sup>®</sup> CortexM4 devices
- Keil.Kinetis\_K20\_DFP.1.0.2.pack for TWR-K24F120M which is available in <install\_dir>/tools/keil\_extensions/uVision4 in this release package
- GNU Tools for ARM Embedded Processors 4.7 2013Q3
- MinGW v3.82.90 with mingw32-base and msys-base packages installed
- Segger J-Link driver V4.88 or later

### 2.3 Board jumper settings

This document focuses on the USB-related jumper settings on the board. For the other jumper settings, refer to the board-related user guide.



Figure-1 Board jumper settings

• J26 1-2: enables 5V VBUS

## 3 USB Code Structure

The USB stack is located in the usb\_v2 subfolder of the MQX RTOS root folder. There are five subfolders in it:

🚞 adapter	File Folder	3/27/2014 1:33 PM
🛅 build	File Folder	3/27/2014 1:33 PM
🚞 example	File Folder	3/27/2014 1:33 PM
🚞 output	File Folder	3/27/2014 1:33 PM
🚞 usb_core	File Folder	3/27/2014 1:32 PM

#### Figure-2 usb\_v2 folder structure

• adapter

Includes the adapter files that allow the USB stack to run on different RTOSs with the same USB core code.

• build

Includes the GCC make files.

• example

Includes all the source code and project files of the USB examples.

• output

The USB library binary file is generated into this folder and all the USB-related public header files are copied to this folder. The examples need to include one folder as the including path in the example project settings.

• usb\_core

Includes the USB source files, such as HAL, controller driver, and class drivers. It also includes the USB library projects.

# 4 Compiling or Running the USB Stack and Examples

## 4.1 Step by step guide for IAR

This section takes IAR as an example to show how to build examples. The other tool chains have similar steps.

1. Open IAR as follows.



Figure-3 IAR

- 2. Add MQX RTOS bsp and psp projects by clicking **Project** → **Add Existing Project**. You can find the corresponding IAR project files in the following paths:
  - bsp <install\_dir>/mqx/build/iar/bsp\_twrk24f120m
  - o psp <install\_dir>/mqx/build/iar/psp\_twrk24f120m



Figure-4 bsp and psp projects

- 3. Add a USB stack library project in the following paths:
  - USB Device Stack

<install\_dir>/usb\_v2/usb\_core/device/build/iar/usbd\_mqx\_twrk24f120m

• USB Host Stack

 $<\!\!install\_dir\!>\!\!/usb\_v2/usb\_core/host/build/iar/usbh\_mqx\_twrk24f120m$ 

• USB OTG Stack

<install\_dir>/usb\_v2/usb\_core/otg/build/iar/usbotg\_mqx\_twrk24f120m



Figure-5 Folder name of USB device stack

4. Add a USB example project.

All the USB examples are located in the example folder. The folder structure is as follows.



This guide adds the USB HID mouse device example.

Freescale MQX USB Stack for TWR-K24F120M GA User's Guide, Rev. 1.0, 08/2014

💥 IAR Embedded Workbench IDE	
<u>File Edit View Project CMSIS-DAP Tools Window Help</u>	
	' 🍾 🏋 🔯 🗩 🦛 🍓 🖦 🛤 🐯 🕺 🕭 🕭
Workspace	IAR Information Center for ARM
dev_hid_mouse_twrk24f120m - Int Flash Debug	
Files 🌮 📴	
Ines     C     Ines     C     Ines     C     Ines	
La Chates workspace	
📙 🕂 🗇 dev_hid_mouse_twrk24f120m - Int Flash Debug 🗸 🗸	
P → P psp_twrk24f120m - Debug	IAR Information Center for ARM
Head → Harring → Har	
	Here you will find all the information you need to get started: tutorials, examp
	and reference guides, support information, and release notes.
	GETTING STARTED USER GUIDES EXAMPLE PROJECTS INTEGRA -
Overview bsp_twrk24f120m dev_hid_mouse_twrk24f120m psp_twrk24f120m	GETTING STARTED USER GOIDES EXAMPLE PROJECTS INTEGRA
Log	
Thu Jul 10, 2014 17:53:11: Loading the I-jet/JTAGjet driver	
8	
Debug Log	
Det	
Ready	NUM 🔜 🔬

Figure-7 Adding a USB example project

5. Build the bsp and psp libraries.

Save the workspace in the resulting dialog box and proceed with the builds.

Save Workspace As	and a start of the	and a start of	×
🕒 🖉 🖉 🖌 🖉 build	d ▶ iar ▶ bsp_twrk24f120m ▶	arch bsp_twrk24f120m	Q
Organize 🔻 New	folder	:== ▼	?
🚺 Downloads 🗐 Recent Places	Name Name	Date modified 7/9/2014 10:42 AM	Type File folder
<ul> <li>Libraries</li> <li>Documents</li> <li>Git</li> <li>Music</li> <li>Pictures</li> <li>Videos</li> </ul>	Release	7/9/2014 2:45 PM 7/9/2014 10:43 AM	File folder File folder
Computer	<b>▼</b> ∢		4
File <u>n</u> ame: Save as <u>t</u> ype: V	Vorkspace Files (*.eww)		•
Aide Folders		Save	cel

Figure-8	Save	Workspace	As	dialog	box
		pass		ananeg	~~~

6. Build a USB stack library.

IAR Embedded Workbench IDE     File Edit View Project CMSIS-DAP Tools Window H	lein	
	• 4	/ 🍾 🗽 🖾 🗩 📣 🎥 📴 😳 🤌 🕭
Workspace	>	
dev_hid_mouse_twrk24f120m - Int Flash Debug	•	
Files	8: B	
🛛 🗗 Untitled Workspace *		
→ ☐ bsp_twrk24f120m - Debug	ebua 🗸	
□	ebug V	
🔲 🗖 usbd_mqx_twrk24f120m - Debua	✓	IAR Information Center for ARM
└─⊞ 🗇 usbh_mqx_twrk24f1 Options	~	Here you will find all the information you need to get started: tutorials, examp
Make		and reference guides, support information, and release notes.
Compile		
Rebuild All		
Clean		
Stop Build		
Add	•	
Remove		
Rename		GETTING STARTED USER GUIDES EXAMPLE PROJECTS INTEGRA
Overview bsp_twrk24f120m Version Control System	▶ 20m ∢ ►	GETTING STARTED USER GUIDES EXAMPLE PROJECTS INTEGRA +
Log     Open Containing Folder		
Thu Jul 10, 2014 17:53 File Properties		
Set as Active		
Log		
Debug Log		
Clean and make the selected project		NUM

- 7. Check the USB library build result.
  - After the USB library is built, you can find the generated library binary file (usbd.a) under <install\_dir>/usb\_v2/output/twrk24f120m.iar/debug/usbd/mqx/
  - In addition, all the USB-related public header files are copied to this folder.

8. Build the USB device HID mouse example. The USB library must compile successfully. Otherwise, the build for the example project may fail.

X IAR Embedded Workbench IDE	
File Edit View Project CMSIS-DAP Tools Windo	W Help
Workspace	× IAR Information Center for ARM
dev_hid_mouse_twrk24f120m - Int Flash Debug	▼
Files	
🗉 🗈 Untitled Workspace *	
📕 🗖 usbd_mqx_twrk24f120m - Debug	IAR Information Center
	mpile Here you will find all the information you need to get s
	and reference guides, support information, and relea
Clea	
Stop	p Build
Add	
Rem	nove
Ren	iame
Overview bsp_twrk24f120m dev_hid_mouse_tv Vers	sion Control System
× Log Ope	en Containing Folder
	Properties
Set	as Active
Debug Log	
	4 m
Clean and make the selected project	

Figure-9 Building the USB device HID mouse example

- 9. Connect the J-Link to the JTAG port (J32) on TWR-K24F120M.
- 10. Connect the micro USB cable from a computer to J37 of TWR-K24F120M to power on the board.
- 11. Click **Download and Debug**. Wait for the downloading to finish.
- 12. Click **Go** to run the example.
- 13. Connect the micro USB cable from a computer to the J23 port of TRW-K24F120M to enable the USB mouse device to work on the computer.

After the mouse device is enumerated by the computer, the mouse will be active, and the mouse pointer draws a rectangle on the computer.

#### 4.2 Additional actions for Keil

The compilation process for Keil is similar to that for IAR. This section focuses on the parts of the Keil downloading process that are different from IAR.

Before we can download the binary to the target board with Keil, we need to set a programming algorithm as follows.

1. Access the options for the target project by right-clicking the target project.

Options for Target 'hwtimer_twrk24f120m Int Flash Rel	ease'
Device Target Output Listing User C/C++ Asm	Linker Debug Utilities
Freescale MK24FN256xx12	
<u>X</u> tal (MHz): 120.0	Code Generation
Operating system: None	Use Cross-Module Optimization
System-Viewer File (.Sfr):	Use MicroLIB 🔲 Big Endian
MK24F25612.svd	Floating Point Hardware: Use FPU
Use Custom SVD File	,
	Deed With Marray Area
Read/Only Memory Areas default off-chip Start Size Startup	Read/Write Memory Areas default off-chip Start Size Nolnit
	RAM1:
□ ROM2: 0	□ RAM2:   □
□ ROM3: ○	RAM3:
on-chip	on-chip
IROM1: 0x0 0x40000 (€	IRAM1: 0x20000000 0x30000 □
□ IROM2: □ 0	□ IRAM2: 0x1FFF0000 0x10000 □
OK Car	Incel Defaults Help

Figure-10 Options for the target project

2. Click the **Debug** tab.

Options for Target 'hwtimer_twrk24f120m Int Flash Rel	lease'
Device Target Output Listing User C/C++ Asm	Linker Debug Utilities
○ Use Simulator     Settings       □ Limit Speed to Real-Time	⊡se: I-LINK / J-TRACE Cortex      Settings     S
Load Application at Startup Run to main() Initialization File: Edit	Load Application at Startup Run to main() Initialization File:
Restore Debug Session Settings	Restore Debug Session Settings
CPU DLL: Parameter: SARMCM3.DLL -REMAP	Driver DLL: Parameter: SARMCM3.DLL -REMAP
Dialog DLL: Parameter: DCM.DLL -pCM4	Dialog DLL: Parameter: TCM.DLL -pCM4
OK Car	ncel Defaults Help

Figure-11 Debug tab

3. Click Setting next to the Use: J-Link/J-Trace Cortex option. The Cortex JLink/JTrace Target Driver Setup dialog box appears.

Cortex JLink/JTrace Target Driver Setup	
Debug Trace Flash Download	
J-Link / J-Trace Adapter	SW Device
SN: 1651344405 💌	IDCODE Device Name Move
Device: J-Link Lite-FSL	SWD
HW : V1.00 dll : V4.84c	Down
FW : J-Link Lite-FSL V1 compiled Ji Port: Max Clock: SW V 2MHz V Auto Clk	Automatic Detection ID CODE:     Add Delete Update IR len:
Connect & Reset Options Connect: Normal  Reset: Normal Reset: Normal Reset: Normal	mal
Interface TCP/IP USB © TCP/IP Scan State: ready 127	ttings Port (Auto: 0) 0 . 0 . 1 : 0 Ping Misc JLink Info JLink Cmd
	OK Cancel Apply

Figure-12 Cortex JLink/JTrace Target Driver Setup dialog box

4. Click the **Flash Download** tab.

Cortex JLink/JTrace Target Driver S	etup			×
Debug Trace Flash Download				
Download Function C Erase Full Chip C Erase Sectors C Do not Erase	<ul> <li>✓ Program</li> <li>✓ Verify</li> <li>✓ Reset and Run</li> </ul>	RAM for Algorithm	Size: 0x1000	
Programming Algorithm				- II
Description	Device Size Device	Type Add	dress Range	
		Start:	Size:	
	Add	Remove		
		ОК	Cancel	Apply

Figure-13 Cortex JLink/JTrace Target Driver Setup dialog box - Flash Download tab

Description	Flash Size	Device Type	Origin
IKXX 256kB PFlash SEC(4	256k	On-chip Flash	Device Family Package
DuCM360 128kB Flash	128k	On-chip Flash	MDK Core
PC18xx/43xx S25FL032 SP		Ext. Flash SPI	
C28F640J3x Dual Flash	16M		
29GL064N Dual Flash	16M	Ext. Flash 32-bit	MDK Core
\Keil\ARM\PACK\Keil\Kinetis	_K20_DFP\1.0	.3\Flash\MKP256_4	KB_SECTOR.FLM

5. Click Add and select MKXX 256kB PFlash SEC(4KB).

Figure-14 Selecting MKxxN 256kB programming flash

### 4.3 Downloading GNU tools ARM embedded 4.7

The compilation process of GNU Tools ARM Embedded is similar to that of IAR and Keil. You need to access the corresponding folder and run mingw32-make to compile the project from the command line or just run the corresponding batch file build\_gcc\_arm.bat for each example.

**Note:** The makefile provided by the USB stack supposes that the GCC tool chain is installed in C:/PROGRA~1/GNUTOO~1/43F2B~1.720 (the default installation path). If the GCC tool chain is installed in another folder in your system, you need to change GCC\_TOOLCHAIN\_DIR in build\common\make\global.mak to the correct path manually. Otherwise, the compilation process will fail. In addition, the path should be in the short file name format. You can get the short file name by using the following command:

```
for %A in ("C:\Program Files\GNU Tools ARM Embedded\4.7 2013q3") do @echo %~sA
```

The string C:\Program Files\GNU Tools ARM Embedded\4.7 2013q3 in the command above should be replaced by the correct target long file name.

Some strange issues may occur when the default installation path of the GCC Tool Chain is changed, hence it is recommended not to change it. In addition, make sure that the **mingw32-base** and **msys-base** packages are installed in your system and the corresponding path has been added into the system path (MINGW\bin), and MINGW\msys\1.0\bin is not added to the system path.

The downloading steps are as follows:

#### 1. Run J-Link GDB Server.

This application is installed along with the J-Link. Select MK24FN256xxx12 as the target device and click **OK**.

SEGGER J-Link GDB Server ¥4.88 - Config		
Connection to J-Link		
Target device		
MK24FN256xxx12		
Little endian		
Target interface		
SWD		
Speed		
C Auto selection		
C Adaptive glocking		
Command line option		
-select USB -device MK24FN256xxx12 -if SWD -speed 1000		
Cancel		

Figure-15 J-Link GDB Server Configuration dialog box

SEGGER J-Link GDB Server ¥4.88		_ <b>_</b> ×	
<u>File</u> <u>H</u> elp			
GDB Waiting for connection J-Link Connected CPU MK24FN256xxx12	Initial SWD speed 1000 kHz Current SWD speed 1000 kHz 3.30 V	Localhost only     Stay on top     Show log window     Generate logfile     Verify download     Init regs on start	
Log output: <u>C</u> lear log			
Target interface speed: 1000kHz Target endian: little Connecting to J-Link J-Link is connected. Firmware: J-Link OpenSDA compiled Apr 24 2014 14:43:37 Hardware: V1.00 S/N: 621000000 Checking target voltage Target voltage: 3.30 V Listening on TCP/IP port 2331 Connecting to targetConnected to target Waiting for GDB connection			
0 Bytes downloaded	L JTAG device		

#### Figure-16 J-Link GDB server configuration result

2. Run **arm-none-eabi-gdb** under the folder where the target binary is located.

In the example, this folder is under the following path:

<install\_dir>/usb\_v2/example/device/hid/hid\_mouse/mqx/make/dev\_hid\_mouse\_twrk24f120m/gcc\_arm/intflash\_release



Figure-17 arm-none-eabi-gdb folder

3. On the gdb client, run the following commands:

```
target remote localhost:2331
monitor reset
monitor flash device = MK24FN256xxx12
load dev_hid_mouse_twrk24f120m.elf
monitor reg pc = (0x00000004)
monitor reg sp = (0x00000000)
monitor go
```

The mouse becomes active on the computer.

## 5 USB Stack Configuration

#### 5.1 Device configuration

All the device configurations are listed in the following file:

 $<\!\!install\_dir\!>\!\!/usb\_v2/usb\_core/device/include/twrk24f120m/usb\_device\_config.h$ 

We can enable or disable the USB class driver through this file, and we can configure the object number to decrease the memory usage or increase the object number to meet some specific requirements.

If you change the configuration of the device stack, both the USB library project and the example project need to be rebuilt.

#### Notes

1

The composite device examples can work only with:

USBCFG\_DEV\_COMPOSITE

All other non-composite device examples can work only with:

USBCFG\_DEV\_COMPOSITE 0

If incorrect settings are configured, a build error will occur and will need to be modified.

#### 5.2 Host configuration

All the host configurations are listed in the following file:

<install\_dir>/usb\_v2/usb\_core/host/include/twrk24f120m/usb\_host\_config.h

We can enable or disable the USB class driver through this file, and we can configure the object number to decrease the memory usage or increase the object number to meet some specific requirements.

If you change the configuration of the host stack, both the USB library project and the example project need to be rebuilt.

### 5.3 OTG configuration

All the OTG configurations are listed in the following files:

- <install\_dir>/usb\_v2/usb\_core/host/include/twrk24f120m/usb\_device\_config.h
- <install\_dir>/usb\_v2/usb\_core/host/include/twrk24f120m/usb\_host\_config.h

You can enable or disable the USB class driver through these files, and configure the object number to decrease the memory usage or increase the object number to meet specific requirements.

If you change the configuration of the OTG stack, both the USB library project and the example project need to be re-built.

#### NOTE

The OTG example requests to use the mini receptacle on the TWR-SER board. The jumper settings should be:

- J4 1 2
- J27 1 2
- For the jumper settings on the TWR-SER board, see the TWR-SER user's guide.

Additional configuration is needed for the host mode:

USBCFG\_HOST\_PORT\_NATIVE 0

Additional configuration is needed for the device mode:

USBCFG\_DEV\_COMPOSITE 0

#### How to Reach Us:

Home Page: www.freescale.com

Web Support: 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 solls products pursuant to standard terms and conditions of sale, which can be found at the following address: 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. The ARM Powered Logo is a trademark of ARM Limited. ©2014 Freescale Semiconductor, Inc.





Мы молодая и активно развивающаяся компания в области поставок электронных компонентов. Мы поставляем электронные компоненты отечественного и импортного производства напрямую от производителей и с крупнейших складов мира.

Благодаря сотрудничеству с мировыми поставщиками мы осуществляем комплексные и плановые поставки широчайшего спектра электронных компонентов.

Собственная эффективная логистика и склад в обеспечивает надежную поставку продукции в точно указанные сроки по всей России.

Мы осуществляем техническую поддержку нашим клиентам и предпродажную проверку качества продукции. На все поставляемые продукты мы предоставляем гарантию.

Осуществляем поставки продукции под контролем ВП МО РФ на предприятия военно-промышленного комплекса России, а также работаем в рамках 275 ФЗ с открытием отдельных счетов в уполномоченном банке. Система менеджмента качества компании соответствует требованиям ГОСТ ISO 9001.

Минимальные сроки поставки, гибкие цены, неограниченный ассортимент и индивидуальный подход к клиентам являются основой для выстраивания долгосрочного и эффективного сотрудничества с предприятиями радиоэлектронной промышленности, предприятиями ВПК и научноисследовательскими институтами России.

С нами вы становитесь еще успешнее!

#### Наши контакты:

Телефон: +7 812 627 14 35

Электронная почта: sales@st-electron.ru

Адрес: 198099, Санкт-Петербург, Промышленная ул, дом № 19, литера Н, помещение 100-Н Офис 331