

Using a USB Device with ColdFire® OTG Module MCF522xx and a PC

by: Paolo Alcántara
RTAC Americas

1 Introduction

This document is an introduction to a universal serial bus (USB) on the go (OTG) module with Freescale's MCF522xx ColdFire microcontroller unit (MCU).

The USB module is a versatile module with 16 endpoints that allow the programmer to build robust applications. The application note demonstrates how to use the OTG USB module with the ColdFire MCF5221x MCU and the CMX USB stack. The PC uses WinUSB as the usb driver and used as the device that sends and receives information to a PC using the M52210 demo board. Codes included for Visual Basic and Codewarrior can be used as a starting point for any project that needs communication between a PC and ColdFire.

Demonstrated in this application note is how to use the USB OTG features with the included software. This software consists of an MCU with a PC running Windows® XP or Vista OS. The MCU functions as a USB device in three different modes, control, interrupt, and bulk transfer.

Contents

1	Introduction	1
2	Getting Started with Software	2
2.1	DEMO Software Features	2
2.2	Basic Steps for Testing the Application Note Software	3
2.3	Running the ColdFireUSB Application.	6
3	ColdFireUSB Protocol	7
3.1	USB Customized Data Packets between the PC and Demoboard	7
4	Using the CMX USB Stack with ColdFire MCF522xx	7
4.1	MCF522xx Software Features.	7
4.2	First Approach to the Demo Board	8
4.3	Changing the USB Device Descriptor in the Project	9
4.4	UART, DMA, RTC and ADC Drivers	10
4.5	USB Device Jumpers	10
5	PC USB Host Software	11
5.1	Project Features	11
5.2	Generic Driver for a PC Application.	11
5.3	WinUSB API	12
5.4	Starting a Project with Visual Basic and FSLwinusb DLL File.	13
6	Conclusion.	13
7	Considerations and References	13

2 Getting Started with Software

2.1 DEMO Software Features

The graphical user interface (GUI) included with this application note is shown in [Figure 1](#). The GUI and M52210DEMO (DEMO board) do the following:

- Send and receive data/information through a USB connection from the DEMO board (USB device) to the PC (USB host).
- Send and receive the following data/information. See [Figure 1](#):
 - Interrupt transfer — receive accelerometer data on x, y and z axis periodically from Freescale part number A7260.
 - Control transfer — send and receive time using a real time clock (RTC) taking advantage that the RTC must be enabled once.
 - Bulk transfers — send typed information from the PC GUI text box to the DEMO board. The Freescale board receives information using universal asynchronous receiver transmitter (UART) interface.

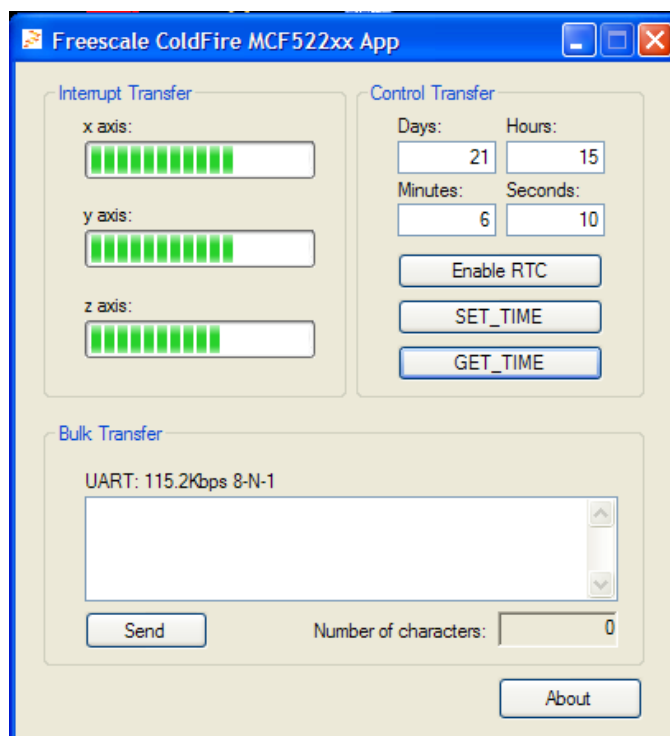


Figure 1. PC GUI

- The PC USB driver installation is required once.
- The PC USB driver used is the WinUSB from Microsoft Windows Development Kit (WDK). The executable file was built using Microsoft Visual Basic 2008 Express Edition. The Dynamic link library (DLL) includes an API to work with Freescale MCF522xx USB. All the software used on the PC is free to distribute. The DLL name is FSLwinusb.dll.

2.2 Basic Steps for Testing the Application Note Software

To test this application note, the following steps must be followed:

1. Download the CMX stack from the Freescale webpage and accept the license.
2. Download the CMXUSB_LITE folder and unzip.
3. Download the AN3750SW.zip into the CMXUSB_LITE folder. This step is important because it avoids access path problems during compilation.
4. The following files are included in the software:
 - ColdFireUSB.exe — This is an application running on the PC. The NET Framework 2.0 must be installed before using the executable file.
 - FSLwinusb.zip — This is the installation file requested by the PC when the ColdFire board is connected.
 - usb_app.elf.s19 — Flash object image for the MCF52210 MCU.
 - Bulk_app.zip — Code for the M52210DEMO board.
 - VisualBasicUSB — Code for Visual Basic on the PC.
5. Download the usb_app.elf.s19 file onto the M52210DEMO board.
6. Connect the M52210DEMO board to a PC using a USB cable. The PC must have Windows XP or Vista installed.
7. Reset the board. For more information on how to use the board go to the *M52210DEMO Hardware User Guide* document written for Freescale, located on the web.
8. For details on how to download the code to ColdFire boards go to the following documents inside the CodeWarrior path installation:
 C:\Program Files\Freescale\CodeWarrior for ColdFire V6.4\Help\PDF\ Targeting_ColdFire.pdf
9. After steps from 1 to 8 a default window appears. See [Figure 2](#):



Figure 2. Found New Hardware Wizard - Part 1

10. Select the option, Install from a list or a specific location [Advanced] and click Next. A new default window appears. Click Browse and select the FSLwinusb carpet as shown in Figure 3. Click next.

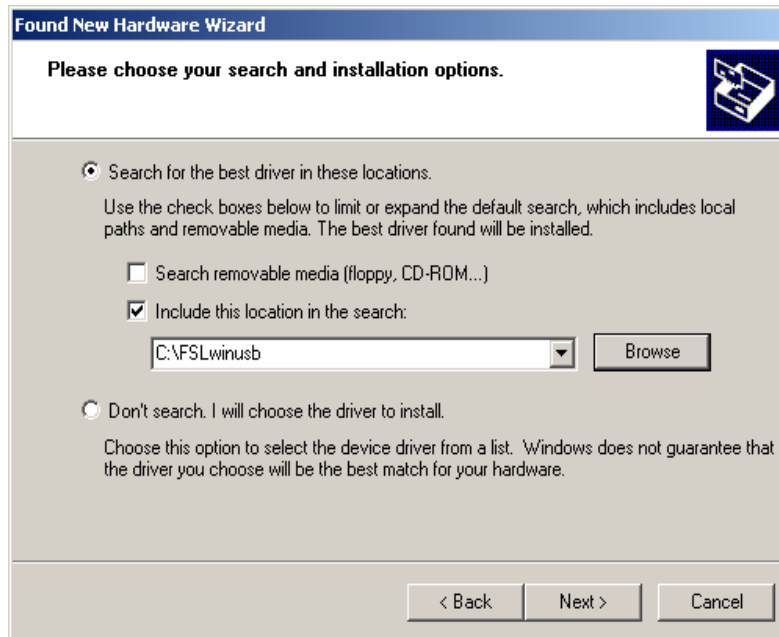


Figure 3. Found New Hardware Wizard - Part 2

11. The INF and SYS files are then installed. See Figure 4 and Figure 5:

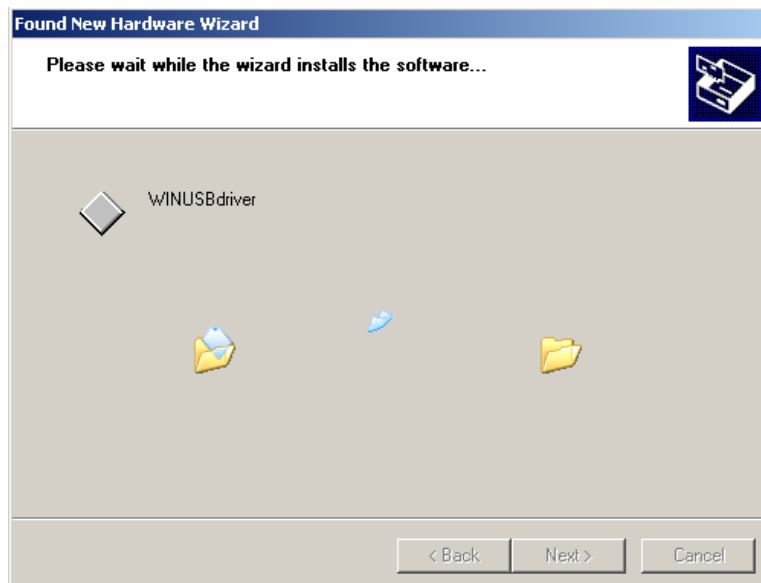


Figure 4. Found New Hardware Wizard - Part 3

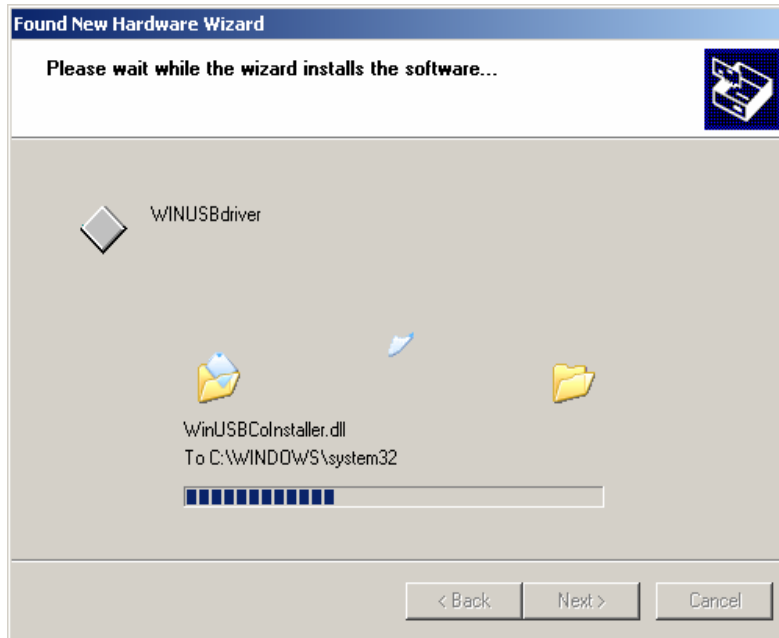


Figure 5. Found New Hardware Wizard - Part 3

12. A message then appears informing installation has been finished. See [Figure 6](#). Click Finish.



Figure 6. Completing the Found New Hardware Wizard

After following the prior steps, launch the ColdFireUSB file.

NOTE

The application note software does not work unless the ColdFire DEMO board has enumerated with the PC.

The ColdFireUSB file can communicate with the DEMO board. The WinUSB is installed in the Add or Remove Programs list from the control panel shown in [Figure 7](#):

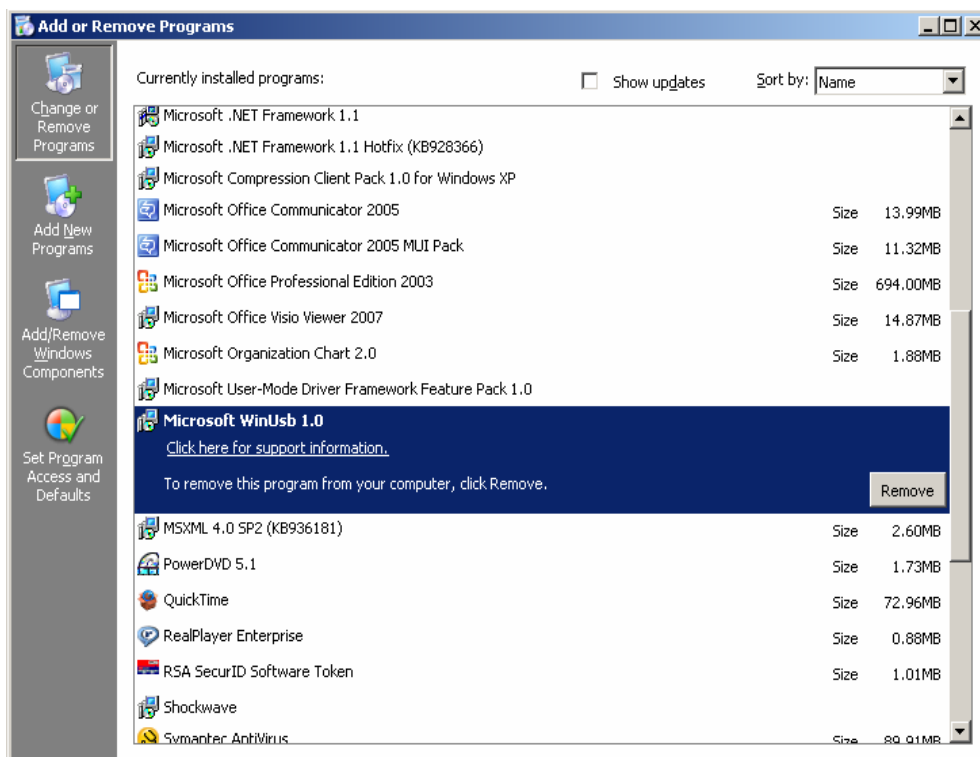


Figure 7. Control Panel → Add/Remove Programs View

No additional hardware is needed to test this application note.

2.3 Running the ColdFireUSB Application

The software included with this application note has the following features:

- Interrupt transfer — Move the DEMO board slowly and watch how the accelerometer (X/Y/Z) readouts change
- Control transfer — To start the real-time clock on the MCU, press the Enable RTC button:
 - a) Write actual time and date in the days, hours, minutes, and seconds boxes
 - b) To initialize the RTC values press the SET_TIME button
 - c) To retrieve the updated RTC reading press the GET_TIME button
- Bulk Transfer — Configure the PC HyperTerminal to setting 115200,8–N–1
- Connect the serial cable from the PC to the COM port on the DEMO board

- Type a text in the window. After completed, press the Send button to transmit the text box contents to the DEMO board

NOTE

The text only transfers from the PC to the DEMO board.

3 ColdFireUSB Protocol

3.1 USB Customized Data Packets between the PC and Demo board

Data is sent and received between the PC and demo board using customized control packets for this application note. RequestType numbers are arbitrarily selected.

Table 1. Customized Control Transfer Descriptions

Request Type	Direction	RequestType	Description
UART_BULK_TRANSFER	OUT	0x76	Sends the number of bytes transferred on a bulk transfer
RTC_ENABLE	OUT	0x35	Enables the RTC controller. One byte is sent
RTC_GET_CLOCK	IN	0x56	Receives RTC time. Five bytes are received
RTC_SET_CLOCK	OUT	0x49	Sends RTC time. Five bytes are sent.

4 Using the CMX USB Stack with ColdFire MCF522xx

4.1 MCF522xx Software Features

The software released with this application note has the following characteristics:

- Project tested in CodeWarrior 6.4 v and 7.0 v. An individual project file for each version is provided.
- One control endpoint (default endpoint 0)
- One bulk endpoint (OUT) and one interrupt endpoint (IN)
- Customized control transfers manage the RTC
- Interrupt transfers manage the on board accelerometer
- Bulk transfers manage the UART
- A customized USB class to communicate between the PC and demo board
- Customized control packets are sent from the PC to the demo board that requests an action for controlling the RTC and bulk transfer length.
- The demo board periodically sends USB interrupt transfers. These transfers are read every 16 milliseconds.

- By changing the power jumpers on the demo board, the demo board receives power from the HOST controller .
- After starting the RTC controller it uses energy from the capacitor (1.5 F) acting as a battery.
- The following MCF52210 modules are used:
 - USB OTG controller
 - RTC
 - ADC
 - DMA
 - UART

4.2 First Approach to the Demo Board

Figure 8 shows the organized codes for the MCF5221x with CodeWarrior. Table 2 shows the function of each file:

CodeWarrior	3K	8	•
CMXUSB_LITE	21K	806	•
USB application	7K	818	•
Drivers	7K	817	•
uart-dma	1K	0	•
uart-dma.c	1504	0	•
uart-dma.h	0	0	
usb	1K	183	•
usb_descriptor.c	520	170	•
usb_descriptor.h	0	0	
usb_driver.c	1260	13	•
usb_driver.h	0	0	
qspi	1K	80	
qspi.c	1344	80	
qspi.h	0	0	
iic	2K	554	
i2c.c	2228	554	
i2c.h	0	0	
adc	228	0	•
adc.c	228	0	•
adc.h	0	0	
rtc	516	0	•
rtc.c	516	0	•
rtc.h	0	0	
app_main.c	416	1	•
27 files	32K	1K	

Figure 8. Files in Project

Each file has an action. These actions are illustrated in [Table 2](#):

Table 2. File Description

File names	Functions
app_main.c	Joins the entire project making possible the USB communication
usb_driver.c	File containing the USB control/interrupt/bulk request for IN/OUT transactions
usb_descriptor.c	USB descriptors for the device
qspi.c	Driver that controls a SPI memory
iic.c	Driver that controls an IIC memory
uart_dma.c	Driver that controls DMA with UART
adc.c	Driver that controls the accelerometer using ADC
rtc.c	Driver that control the RTC
usb_config.h	Link between CMX stack and application note software
mcf5222x_vectors.s	Vector table connects the interrupts to the proper exception routine

4.3 Changing the USB Device Descriptor in the Project

The `Usb_descriptor.c` contains the device information. To learn more about device descriptors consult Freescale application note *AN3492 — USB and Using the CMX USB Stack*.

In this example code and for future applications, the following information is highlighted and must be customized for each descriptor:

- Customized USB class — This can define customized control packets, number of endpoints, and other behaviors depending upon the programmer's needs.
- USB vendor and product ID — These two variables must be equal to the INF file in Windows. The operating system (OS) associates the device descriptor to a specific PC USB driver. The VID and PID are selected arbitrarily in this application and are not intended to be used for production.
- Size of the device descriptor — If variables are added or removed from device descriptor, the size variable must be changed.
- Number, type, size and direction of endpoints — Depending on your application; number defines the amount of pipes created, type selects from four kinds of endpoints (control, bulk, isochronous, and interrupt), size is the maximum packet length driven by the device, and direction defines if the packets come from or go to a USB host. The project endpoint map is shown in [Figure 9](#).

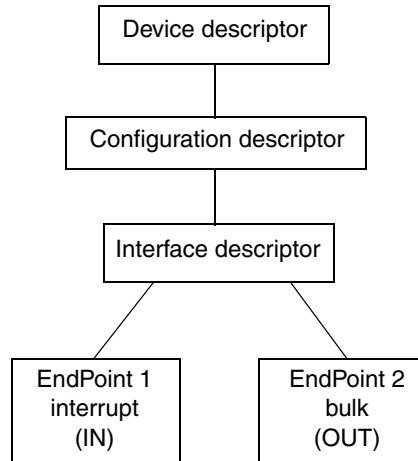


Figure 9. Device Descriptor Map

In this application, endpoints have the following functions:

- Endpoint 0 (default IN/OUT) — Sends and receives standard control transfers. Customized packets are added for control:
 - Bulk transfer length
 - RTC enable command
 - RTC receive-clock
 - RTC send-clock
- Endpoint 1 (IN) — Periodically sends the state of the accelerometer.
- Endpoint 2 (OUT) — Receives bulk transfers from the host.

NOTE

A control transfer sends the length of a transfer. The data length is limited to 1024 bytes.

4.4 UART, DMA, RTC and ADC Drivers

For a fast response from the PC the UART, DMA, RTC and ADC modules are managed by interrupts. These drivers are adapted to manage peripherals. For example, the accelerometer and the 32.768 KHz oscillator. The data read from the accelerometer is sent in a raw format to a PC. No filter is applied.

4.5 USB Device Jumpers

The board was designed to give power when working as a USB host and receive power when working as a USB device. [Table 2](#) are the required positions for the jumpers.

Table 3. Jumper Positions

Jumper	Position
+5V_SEL	VUSB
+3V_SEL	VUSB
FLT	SET
MODE	P
DP	REMOVE
DM	REMOVE
JP1	RTC
J4	1-2

If the demo board is not self-powered it must receive power from the USB receptacle and notifies self-power change to the device descriptor.

JP1 must be set to the RTC option for oscillator to work, otherwise it does not work. If JP1 is not set the GUI time buttons (Enable RTC, Get RTC and Set RTC) freeze the firmware and the GUI.

5 PC USB Host Software

5.1 Project Features

The GUI has the following characteristics:

- Programmed with the Microsoft Visual Basic 2008 Express edition.
- FLSwinusb.dll file is used to communicate the USB low-level driver and the GUI.
- WdfCoInstaller01005.dll and WinUSBCoInstaller.dll are files needed for the USB low level drivers. These files come from Windows Development Kit libraries, available on the Microsoft website.
- The GUI only works on computers with .NET Framework 2.0 or greater.
- The USB driver can be installed on Windows XP with SP2 and Windows Vista 32-bit edition.

5.2 Generic Driver for a PC Application

The following details must be used to change the Visual Basic project included with this application note:

- USB_API.vb — Connects the DLL file containing all the functions available from the usb driver. A brief explanation of each function is included.
- GUI.vb — Contains all the DLL functions and how it works when opening and closing the USB port on the PC.

The INF file (FSLwinusb.inf) vendor and product ID must be the same as included in the device descriptor file of the demo board. In this example, the driver uses a VID = 0x1983 and a PID = 0x0101 shown in [Figure 10](#).

The 32 hexadecimal DeviceInterfaceGUIDs number must be the same for the executable file and the INF file.

```
[Manufacturer]
%ProviderName% = MyDevice_winUSB,NTx86,NTamd64
;;; use your own PID and VID
[MyDevice_WinUSB.NTx86]
%USB\MyDevice.DeviceDesc%=USB_Install, USB\VID_1983&PID_0101

[MyDevice_WinUSB.NTamd64]
%USB\MyDevice.DeviceDesc%=USB_Install, USB\VID_1983&PID_0101
.
.
.

[Dev_AddReg]
; use your own GUID!!!
HKR,,DeviceInterfaceGUIDs,0x10000,"{C3AFBF6C-7C33-11DC-9733-2CD755D89593}"
```

Figure 10. PID and VID on the FSLwinusb.INF File

Figure 11 explains how these variables must match.

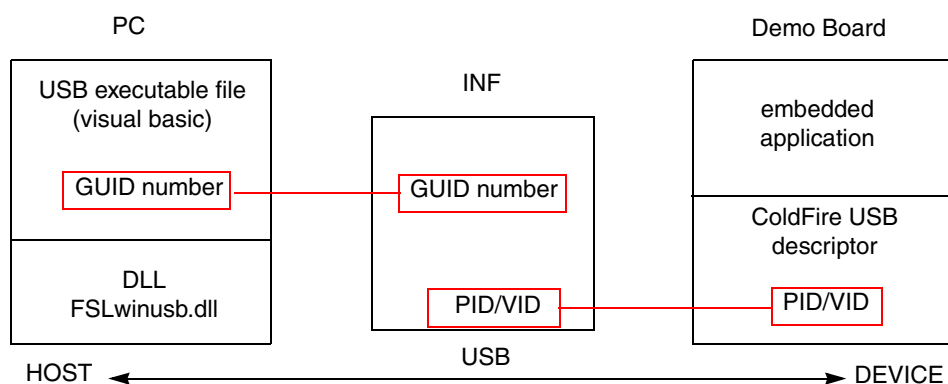


Figure 11. Variables in the PC, INF, and Demo Board

5.3 WinUSB API

The use of the Visual Basic 2008 Express Edition does not require buying a license and can be downloaded from the Microsoft web page. The DLL provided with this application note is a generic file that does not need to be changed and can be used on more complex USB applications. The only files that need to be changed are the Visual Basic code and the INF file.

The PC USB application source code and description is provided with this application note.

For more information about WinUSB functions, consult the WinUSB API from the Microsoft website.

5.4 Starting a Project with Visual Basic and FSLwinusb DLL File

The best way to start a new USB application is to use the Visual Basic USB source code as a starting point. It contains data types, and functions needed by the USB DLL. The DLL file names can not be changed or the application will not work.

6 Conclusion

This application note proposes a simple and low cost solution for the MCF522xx OTG MCU family that requires a method to send and receive information to a PC running Windows XP or Vista OS.

Included is the application note software that can be used as a starting point to build more complex USB applications with a few changes.

7 Considerations and References

- Find the newest software updates and configuration files for the MCF522xx on the Freescale Semiconductor home page, www.freescale.com.
- This application note uses MCF52210 and MCF52211 MCUs. However it can be adapted to all V2 with internal ROM memory and a USB OTG controller.
- The M52210DEMO Development Board was employed for the USB software demo.
- For more information of the USB for the MCF52211 go to the *MCF52211 ColdFire Integrated Microcontroller Reference Manual* at www.freescale.com.
- The USB application software was developed in CodeWarrior for ColdFire 6.4 v and 7.0 v.
- Download the application note software from www.freescale.com. Inside the AN3750SW.zip are CodeWarrior source codes and a Visual Basic project to control the USB in a PC with Windows XP or Windows Vista 32-bit edition.

How to Reach Us:

Home Page:

www.freescale.com

Web Support:

<http://www.freescale.com/support>

USA/Europe or Locations Not Listed:

Freescale Semiconductor, Inc.
Technical Information Center, EL516
2100 East Elliot Road
Tempe, Arizona 85284
1-800-521-6274 or +1-480-768-2130
www.freescale.com/support

Europe, Middle East, and Africa:

Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
www.freescale.com/support

Japan:

Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064
Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:

Freescale Semiconductor China Ltd.
Exchange Building 23F
No. 118 Jianguo Road
Chaoyang District
Beijing 100022
China
+86 10 5879 8000
support.asia@freescale.com

For Literature Requests Only:

Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or +1-303-675-2140
Fax: +1-303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

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

Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor 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 Semiconductor 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 Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Freescale Semiconductor was negligent regarding the design or manufacture of the part.



Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners.

© Freescale Semiconductor, Inc. 2008. All rights reserved.

AN3750

Rev. 0
09/2008