Running the USB AUDIO example from the LPC1700 Code Bundle from NXP (for Keil MDK-ARM)

Product Line Microcontrollers
Business Line Standard ICs

Contents

Prerequisites for running this example

▸ There are several hardware and software tools necessary in order to run this example:

  – MCB1760 or MCB1750 Evaluation Board from Keil

  – LPC1700 Code Bundle software package

  – PC with 2 USB ports and 125MB of available hard disk space

  – Keil MDK-ARM IDE (free eval version is ok)

  – FlashMagic In-System Programming software and PC serial port – OR – ULINK2 / ULINK–ME JTAG debugger

▸ Please refer to the LPC1700 Sample Code Bundle Introduction / Setup file (code.bundle.lpc17xx.keil.intro.setup.pdf) for more details on setting up the hardware and software environment

  – This file can be found here:
    http://www.standardics.nxp.com/support/software/code.bundle.lpc17xx.keil/

Example Description

▸ The Audio project is a demo program for the Keil MCB1700 Board using the NXP LPC17xx Microcontroller

▸ It demonstrates an USB Audio Device - Speaker

▸ The USB Audio Device is recognized by the host PC running Windows which will load a generic Audio driver and add a speaker which can be used for sound playback on the PC

▸ Potentiometer on the board is used for setting the Volume

▸ Since DMA is used for ISO communication, the USB RAM is allocated for DMA purposes. A scatter file, dma.sct, is used in the Keil project file

Running the USB AUDIO example from the LPC1700 Sample Code Bundle using the ULINK2 or ULINK-ME JTAG debugger

USB AUDIO example using JTAG debugger Step 1: Hardware Setup

▸ Make sure your computer and your board are connected as shown

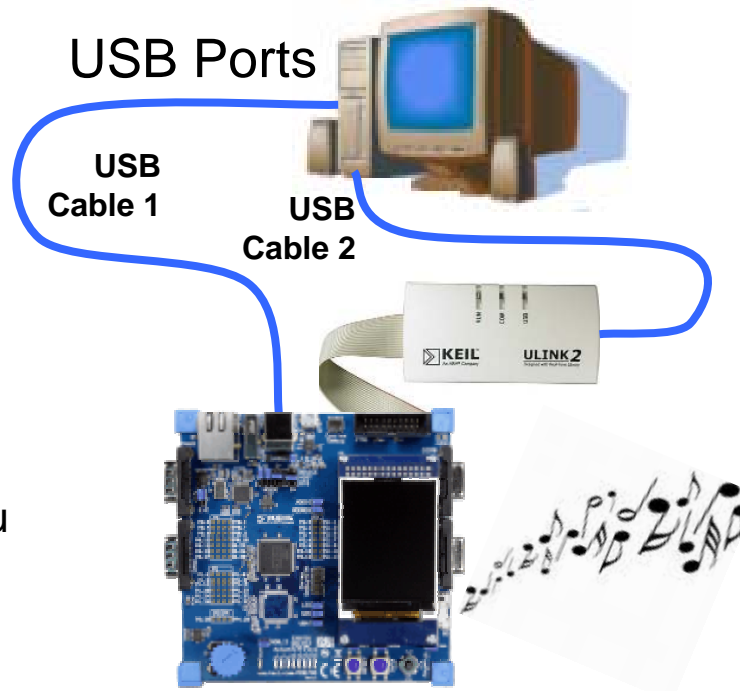▸ This powers the board, provides USB device connection to the PC, and provides JTAG debugging / flash programming

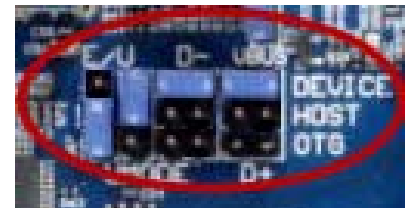**Note:** Remove the RST and ISP jumpers on the MCB1700 board:

USB Ports

USB Cable 1

USB Cable 2

Note: Position USB jumpers as shown below (VBUS jumper is closed):

Note: instead of ULINK2 you can also use ULINK-ME:

USB AUDIO example using JTAG debugger Step 2: Opening up the Keil uVision3 project

▸ Open the project by double-clicking on the following file here:
  `..\LPC17xxSampleSoftware.xxx\Keil\USBAudio\usbaudio.uv2`

  – Note: Ignore all the 'invalid path' warnings if you see some (just click 'OK')

▸ Familiarize yourself with the following files:

  – audio.h   USB Audio related header

  – adcuser.h   USB Audio Device Class user specific header

  – adcuser.c   USB Audio Device Class user APIs

  – usb.h   USB header

  – usbaudio.h   USB Audio misc. definition header

  – usbcfg.h   USB configuration header

  – usbcore.h   USB Core header

  – usbcore.c   USB Core APIs

  – usbdesc.h   USB descriptor header

  – usbdesc.c   USB descriptor APIs

USB AUDIO example using JTAG debugger Step 2: Opening up the Keil project (cont.)

- usbhw.h   USB hardware header

- usbhw.c   USB hardware APIs

- usbreg.h   USB misc. register header

- usbuser.h   USB user header

- useuser.c   USB user APIs

- usbmain.c   USB Audio test module

- usbaudio.uv2   uVision project file

- dma.sct   uVision scatter file for use of DMA

- system_LPC17xx.c: Cortex-M3 Device Peripheral Access Layer Source File (CMSIS–compliant)

- startup_LPC17xx.s: CMSIS Cortex-M3 Core Device Startup File (CMSIS–compliant)

USB AUDIO example using JTAG debugger Step 3: Compiling the example project

▸ Make sure the uVision3 Target setting is 'FLASH'   FLASH

▸ Build the project clicking the "Rebuild all target files"-button

▸ Example should build without errors or warnings

```
Build target 'Flash'
compiling usbuser.c...
compiling usbcore.c...
compiling usbdesc.c...
compiling usbdmain.c...
compiling usbhw.c...
compiling adcuser.c...
compiling system_LPC17xx.c...
compiling core_cm3.c...
assembling startup_LPC17xx.s...
linking...
Program Size: Code=6404 RO-data=552 RW-data=88 ZI-data=1344
FromELF: creating hex file...
".\Obj\usbaudio.axf" - 0 Error(s), 0 Warning(s).
```

USB AUDIO example using JTAG debugger
Step 4: Programming the code to on-chip Flash

▸ Select Flash as the target (if not already done) FLASH

▸ Make sure JTAG debugger is connected to PC and board

▸ Click on the LOAD button to start download LOAD

▸ Verify that the download is ok (in the Build window)

```
Erase Done.
Programming Done.
Verify OK.
```

▸ Please note that the code is not running yet…

USB AUDIO example using JTAG debugger Step 5: USB Audio device enumeration

▶ Remove ULINK / ULINK-ME from the board

▶ Press RESET button to start code execution

▶ PC should indicate device enumeration

▶ The USB connection between the MCB1700 board and the PC now not only provides power to the board but also connects the MCB1700 to the PC as an Audio Device via USB

USB AUDIO example using JTAG debugger Step 6: Verifying USB device enumeration

- Make sure the potentiometer on the MCB1700 (= volume) is not all the way down to 0

- Start an mp3 file (e.g. in Windows Media Player)

- You should hear the song playing from the MCB1700 (via isochronous USB transfers)
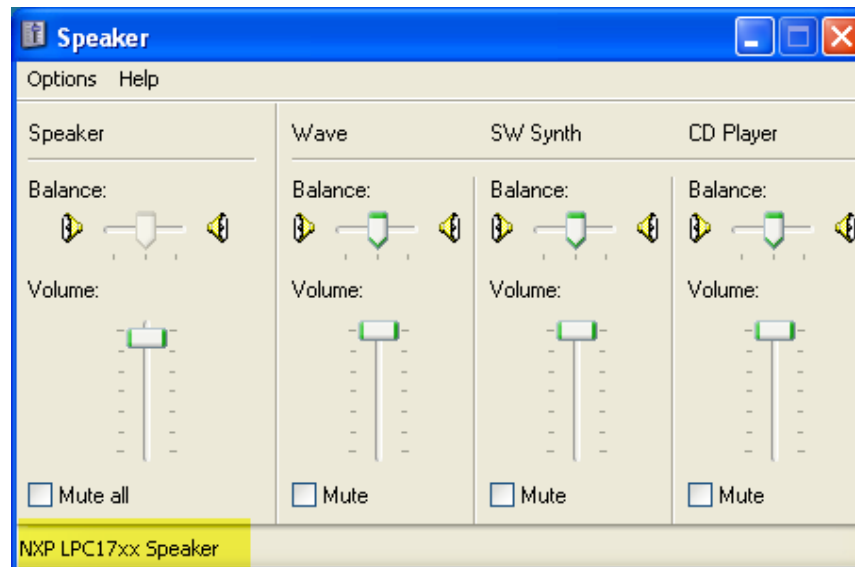
USB AUDIO example using JTAG debugger Step 7: Verifying the Device Manager

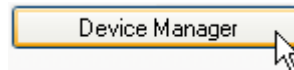▸ On your Desktop, click on "My Computer"
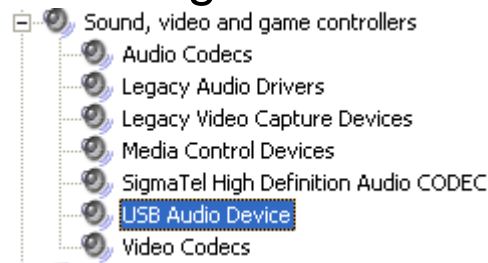
▸ Click on "Control Panel"
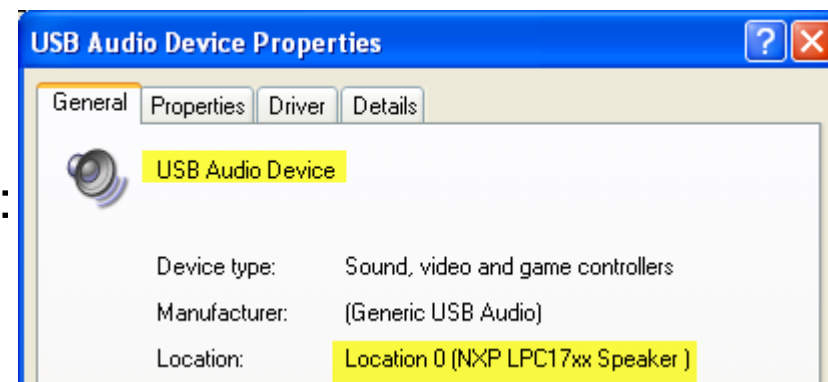
▸ Double-Click on "System"

▸ Click on the "Hardware" tab

▸ Click on "Device Manager"

▸ Under "Sound, video and game controllers", a new USB Audio Device should be shown
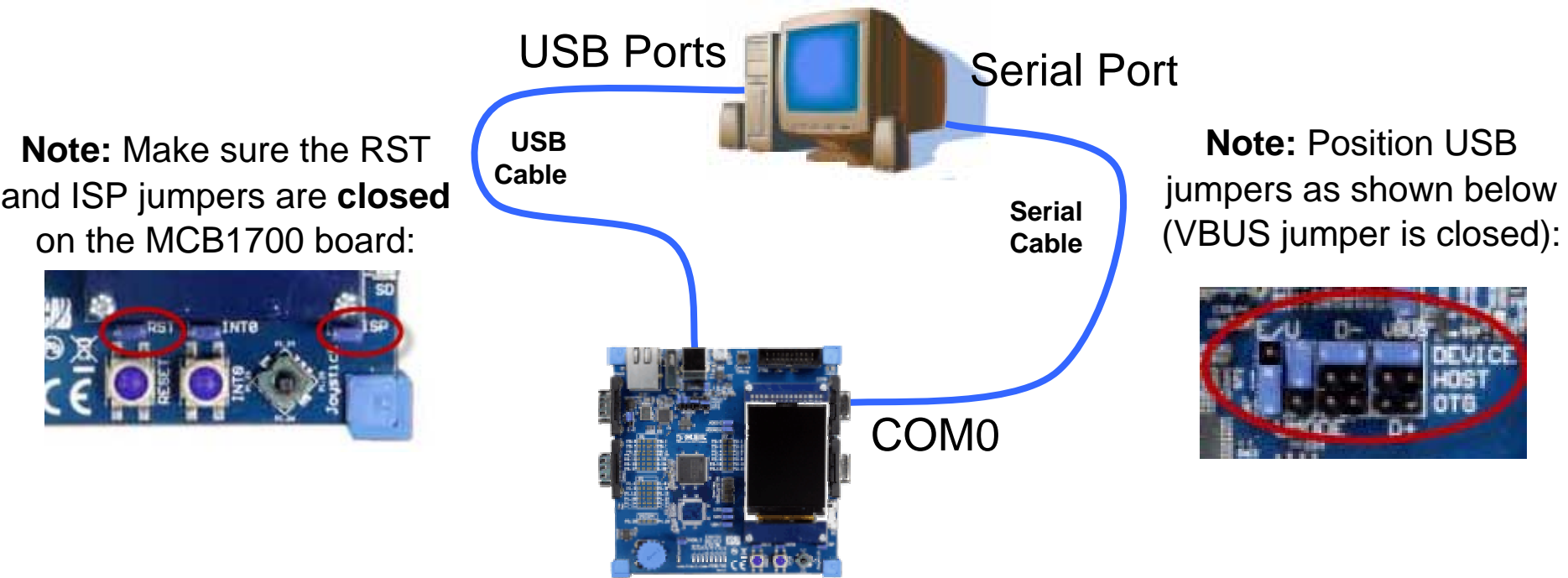
▸ Double-click it to see the full information:

Running the USB AUDIO example from the LPC1700 Sample Code Bundle using the FlashMagic ISP software

USB AUDIO example using FlashMagic
Step 1: Hardware Setup

▸ Make sure your computer and your board are connected as shown

▸ This powers the board, provides output from the UARTs to the PC, and provides for ISP programming

USB Ports

Serial Port

Note: Make sure the RST and ISP jumpers are **closed** on the MCB1700 board:

USB Cable

Serial Cable

Note: Position USB jumpers as shown below (VBUS jumper is closed):

COM0

USB AUDIO example using FlashMagic
Step 2: Opening up the Keil uVision3 project

▶ Open the project by double-clicking on the following file here:
   `..\LPC17xxSampleSoftware.xxx\Keil\USBAudio\usbaudio.uv2`

   – Note: Ignore all the 'invalid path' warnings if you see some (just click 'OK')

▶ Familiarize yourself with the following files:

   – audio.h   USB Audio related header

   – adcuser.h   USB Audio Device Class user specific header

   – adcuser.c   USB Audio Device Class user APIs

   – usb.h   USB header

   – usbaudio.h   USB Audio misc. definition header

   – usbcfg.h   USB configuration header

   – usbcore.h   USB Core header

   – usbcore.c   USB Core APIs

   – usbdesc.h   USB descriptor header

   – usbdesc.c   USB descriptor APIs

USB AUDIO example using FlashMagic
Step 2: Opening up the Keil project (cont.)

- usbhw.h   USB hardware header
- usbhw.c   USB hardware APIs
- usbreg.h   USB misc. register header
- usbuser.h   USB user header
- useuser.c   USB user APIs
- usbmain.c   USB Audio test module
- usbaudio.uv2   uVision project file
- dma.sct   uVision scatter file for use of DMA
- system_LPC17xx.c: Cortex-M3 Device Peripheral Access Layer Source File (CMSIS–compliant)
- startup_LPC17xx.s: CMSIS Cortex-M3 Core Device Startup File (CMSIS–compliant)

USB AUDIO example using FlashMagic
Step 3: Compiling the example project

▸ Make sure the uVision3 Target setting is 'FLASH'

▸ Build the project clicking the "Rebuild all target files"-button

▸ Example should build without errors or warnings

```
Build target 'Flash'
compiling usbuser.c...
compiling usbcore.c...
compiling usbdesc.c...
compiling usbdmain.c...
compiling usbhw.c...
compiling adcuser.c...
compiling system_LPC17xx.c...
compiling core_cm3.c...
assembling startup_LPC17xx.s...
linking...
Program Size: Code=6404 RO-data=552 RW-data=88 ZI-data=1344
FromELF: creating hex file...
".\Obj\usbaudio.axf" - 0 Error(s), 0 Warning(s).
```

▸ As you can see, a .hex file is created in the progress…

  – If not, check the 'Create HEX File' box under
    'Project > Options for Target 'FLASH' > Output':

USB AUDIO example using FlashMagic
Step 4: Setting up FlashMagic

▸ Make sure no other program is using the serial port
(e.g. TeraTerm, Hyperterminal, etc.)

▸ Make sure the serial cable is connected to the PC and to COM0
(=UART0) on the MCB1700 board

▸ Launch FlashMagic application

▸ To make sure you're connected to the LPC1768, click
"ISP > Read Device Signature" and look for the following response:

Device Signature

Device ID: 0x   26013F37

Bootloader Ver: 4.1
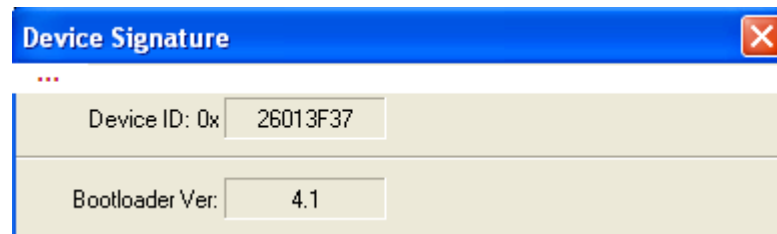
– If you get an error message instead, check your connections…

USB AUDIO example using FlashMagic
Step 5: Programming the code to on-chip Flash
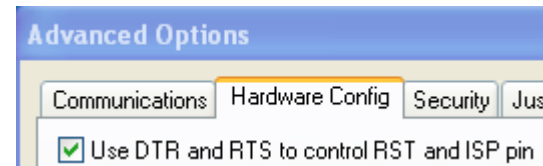
▸ Make the following selections:

Step 1 - Communications		Baud Rate:	19200
Device:	LPC1768	Interface:	None (ISP)
COM Port:	COM 1	Oscillator (MHz):	12.000000

▸ Go to "Options > Advanced Options > Hardware Config"

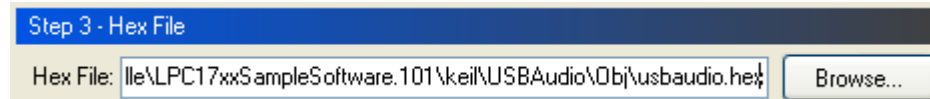- Make sure the following box is checked:

Advanced Options

Communications | Hardware Config | Security | Jus

☑ Use DTR and RTS to control RST and ISP pin

▸ Use the "Browse" button to select the USB HID project hex file:
   `…\LPC17xxSampleSoftware.xxx\keil\USBAudio\Obj\usbaudio.hex`

Step 3 - Hex File

Hex File: lle\LPC17xxSampleSoftware.101\keil\USBAudio\Obj\usbaudio.hex   Browse...

▸ Click Start!

- FlashMagic programs the selected
  hex file into the LPC1768

Step 5 - Start!

Start

USB AUDIO example using FlashMagic
Step 6: USB AUDIO device enumeration

▸ The code should already be running…

▸ If not, press the Reset button to start code execution

▸ PC should indicate device enumeration

ⓘ **Found New Hardware** ✕
Your new hardware is installed and ready to use.

▸ The USB connection between the MCB1700 board and the PC now not only provides power to the board but also connects the MCB1700 to the PC as an Audio Device via USB

USB AUDIO example using FlashMagic
Step 6: Verifying USB device enumeration

▸ Make sure the potentiometer on the MCB1700 (= volume) is not all the way down to 0

▸ Start an mp3 file (e.g. in Windows Media Player)

▸ You should hear the song playing from the MCB1700
(via isochronous USB transfers)
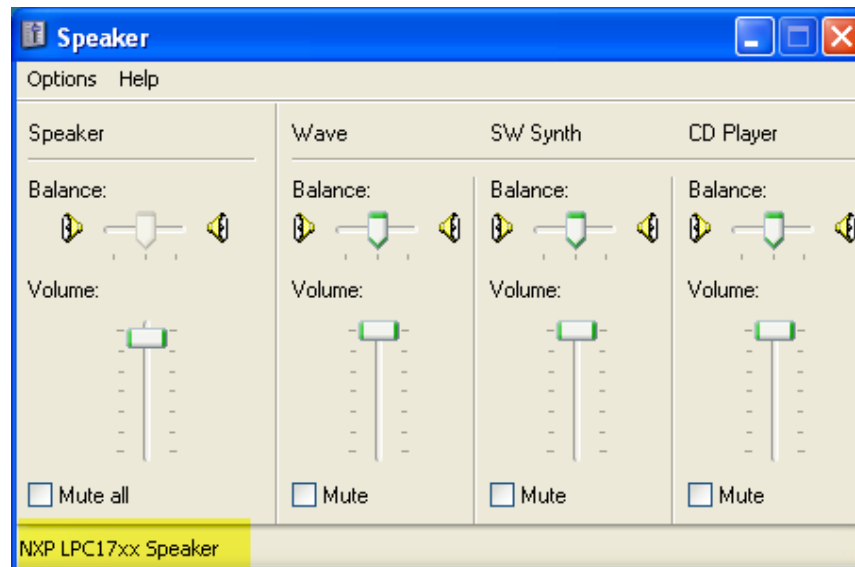
USB AUDIO example using FlashMagic
Step 7: Verifying the Device Manager

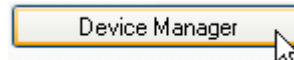▸ On your Desktop, click on "My Computer"
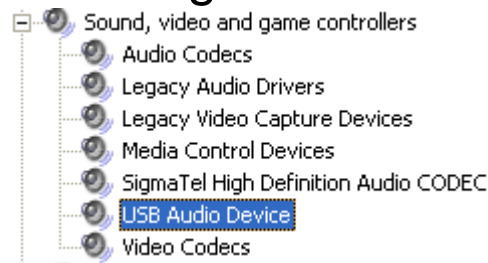
▸ Click on "Control Panel"

▸ Double-Click on "System"

▸ Click on the "Hardware" tab

▸ Click on "Device Manager"

▸ Under "Sound, video and game controllers", a new USB Audio Device should be shown

▸ Double-click it to see the full information: