

Processor SDK RTOS USB

RTOS Software Developer Guide → USB

Overview • Driver Overview • Additional References	User Interface • Driver Configuration • API Call Flow • Hardware Setup
Application • Examples	Debug

Driver Overview

PDK USB driver (USB LLD) provides the following USB class functions while isolating application from the complexity of low level USB and USB class protocols:

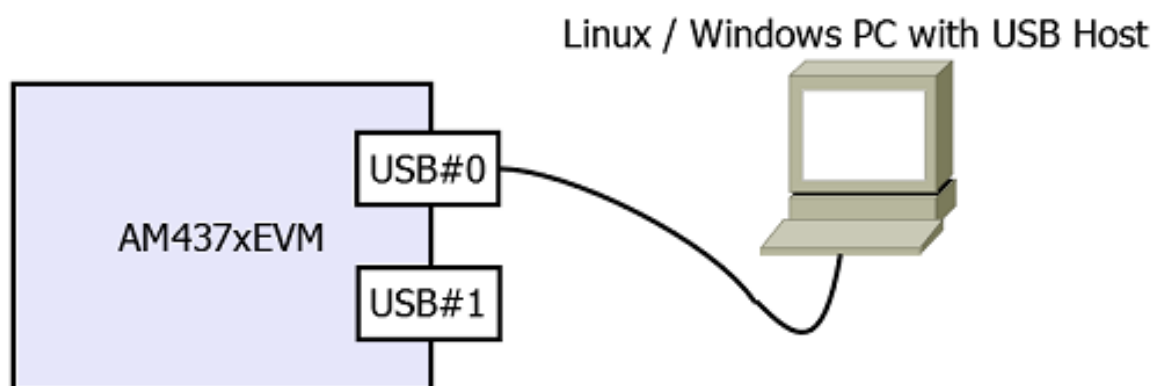
- USB device Mass Storage Class
- USB host Mass Storage Class
- USB device Audio Class

In rest of the page AM437x EVM is being referred as an example. Please check Release Notes for list of supported EVMs for driver.

Modes of Operation

- **USB device Mass Storage Class**
 - **USB device Mass Storage Class with RAM DISK**

In this mode, a user-selected USB instance of the EVM will be working in device mode and will behave like a USB thumb drive. It uses part of the EVM DDR memory and exposes it as a physical storage for another USB host application. After the host PC enumerates this EVM-thumb drive, the PC will see a USB storage device. This EVM-thumb drive is not yet formatted with any file system and requires user to format it before use.

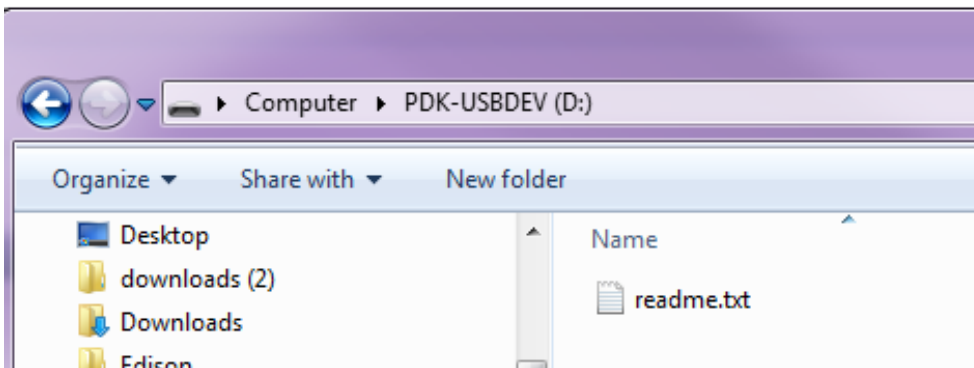
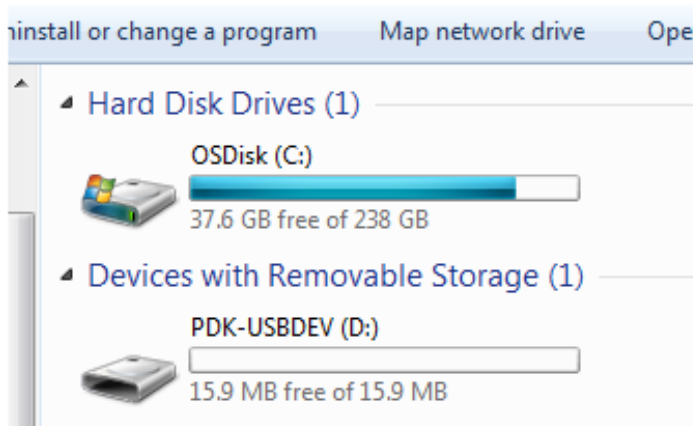


The following screen shots show what one would expect when running the device mode demo application and plugging in a USB cable from the EVM USB port #0 to a PC running Windows

Printout from demo application:

```
RTOS USB Dev MSC example!!  
Done configuring USB and its interrupt
```

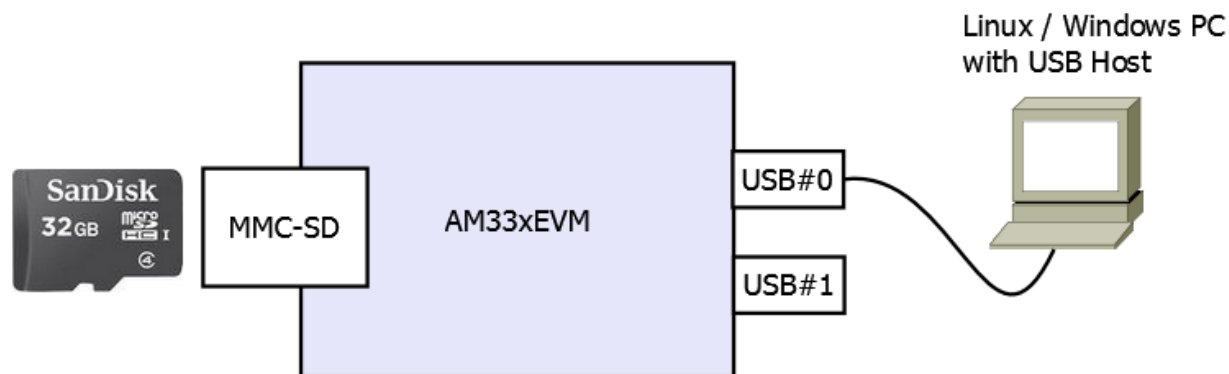
The MSC device is detected in Windows, and a FAT formatted USB drive named "PDK-USBDEV" should be seen in the "Window Explorer". The content of the drive is just a readme.txt file. This USB drive can be manipulated like any other removable USB drive.



Windows might show a message saying it should be scanned and fixed. We can just ignore it and just continue without scanning.

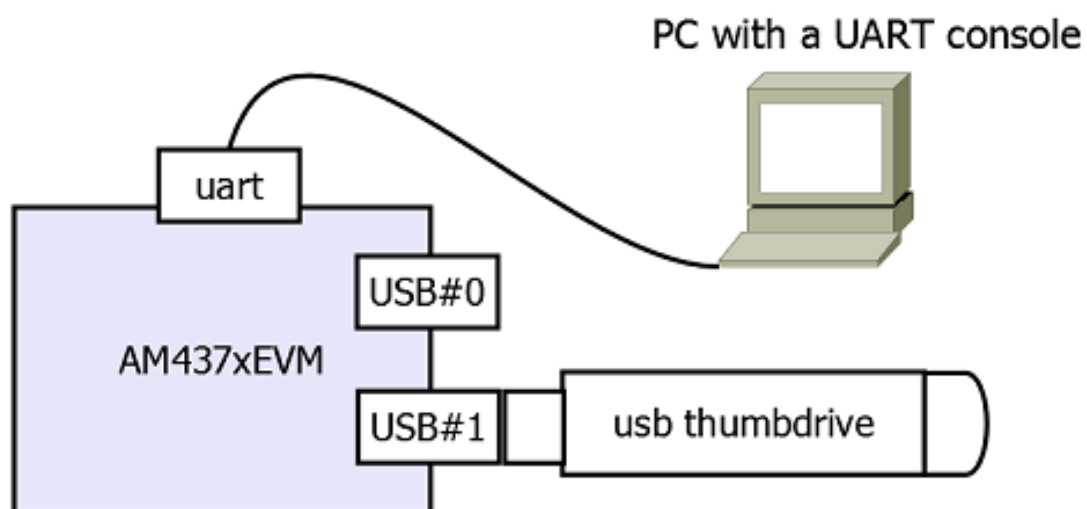
- **USB device Mass Storage Class with MMCSD card**

This example acts like a USB - MMCSD card reader. The example exposes the EVM's MMCSD card to the Host PC via USB MSC. The Host PC can manipulate files on the attached MMCSD card on the EVM. This example is currently supported on AM335GP EVM. This is how it looks. Its code is similar to that of the USB device MSC example but with the call back functions calling MMCSD API's instead of RamDisk APIs



- **USB host Mass Storage Class**

In this mode, the USB instance will act as a USB host communicating with a USB device that supports Mass Storage Class function (USB thumb drive or a small USB hard drive). The demo example code utilizes a shell interface via the EVM via UART for interaction with the example. The shell provides some basic commands to manipulate the content of the attached USB disk drive.



Screenshot of a MSC host mode example running in RTOS after plugging in a USB thumb drive into USB port #1

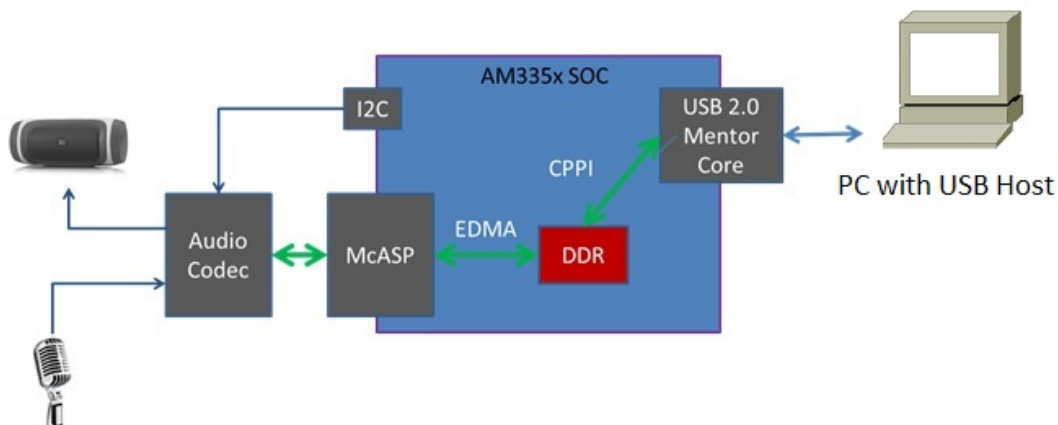
```

0:>help
Available commands
-----
help : Display list of commands
ls   : Display list of files
cd: Change directory
mkdir: Create directory
rm   : Delete a file or an empty directory
pwd  : Show current working directory
cat  : Show contents of a text file : cat <FILENAME>
      Write to a file : cat <INPUTFILE> > <OUTPUTFILE>
      Read from UART : cat dev.UART
      Write from UART : cat dev.UART > <OUTPUTFILE>
0:>lsD---- 2013/03/18 14:06      0  DRIVER~1
----A 2013/11/22 10:14      233984  SOCKET~1.DOC
----A 2013/11/22 10:16      75520  UDP~1.PCA
----A 2013/11/20 17:12      50456  CAP~1.PCA
----A 2013/11/20 17:12       1100  mylog.txt
----A 2013/10/15 13:45      1734  README.TXT

```

- **USB device Audio Class**

In this mode, USB instance of the EVM will be working in device mode and will behave like a USB headset with MIC. It uses audio codec on the EVM for running playback and record initiated by the USB host. McASP module will be used to transfer the data between USB device and audio codec on the board. EVM will appear as a new USB audio device on the host PC. Changing the default audio device on the host PC to EVM USB device will allow the playback and record operations between EVM and USB host. This mode of operation is currently supported on AM335X GP EVM, OMAP-L137 EVM and OMAP-L138 LCDK.



Driver Configuration

- **Data Structures:**

- **tUSBDMSCDevice:** Defined in usbdmsc.h. It is used in USB device mode application. This structure must be filled with the intended vendor and product ID as well as other product information and also the function pointers to functions that handle the disk functions (open/read/write/close, etc.). These product information will show up in the device and interface USB descriptors that are used during device enumeration. This device MSC class data is then assigned to the field usbClassData of the USB_Params bellow.

- **USB params:** USB_Params structure is declared in usb_drv.h. This structure must be provided to the USB driver. It must be initialized before the USB_open() function is called.
- **USB APIs:** main USB LLD and USB MSC API's are declared in usb_drv.h and usbdmhc.h and usbmhc.h provided in the root USB LLD directory.
- **General USB LLD expectations:**

The USB LLD will setup appropriate USB clock and power domains for the particular SOC being in used as part of its "device specific peripheral" functions.

After the USB_open() is called, the driver expects the application code to sets up USB interrupts with the interrupt handler being the USB LLD provided interrupt handler. Then the application have to call the USB LLD provided API USB_irqConfig() which enables USB module's interrupts. In device mode, both USB core and USB misc interrupts are used in the USB device MSC application. In host mode, the USB host MSC only uses USB core interrupts.

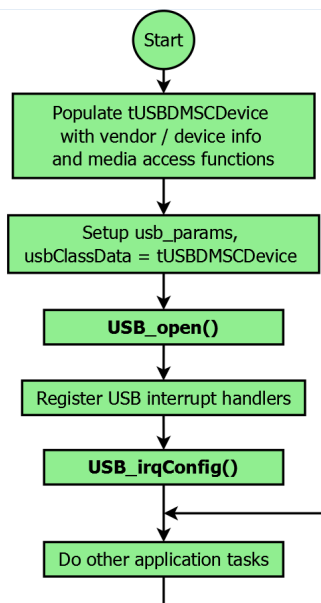
After these steps, application code then can expect to have USB enumeration done and start USB transfer through the provided APIs.

API Call Flow

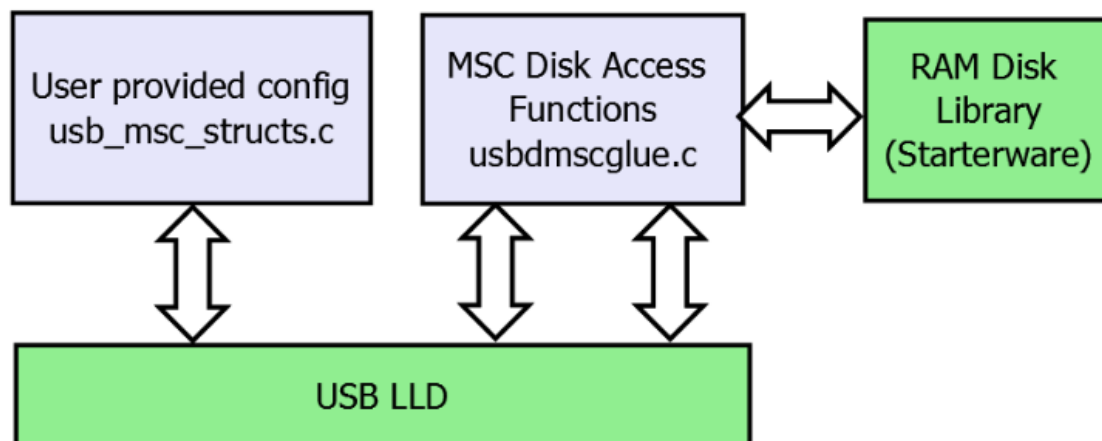
• USB Device MSC

The example application code uses the USB library, configures it as a USB device with MSC function. The example also provides functional codes that access a RAM disk (included from the Utils library in the included Starterware). The LLD calls these MSC back-ends functions to access the RAM disk. User can replace these functions with other functions that access other types of media or devices (MMCSD for example). The RAM disk image provided in the example demo application is not currently formatted. Thus the once enumerated, the PC will require the USB disc to be formatted before use.

Below diagram is the sequence of API calls that starts the USB device MSC application. All USB events are handled internally in the LLD and in the interrupt context.



User provided disk functions will be called from the LLD to handle the actual physical disk access. The overview of USB Device MSC example application:

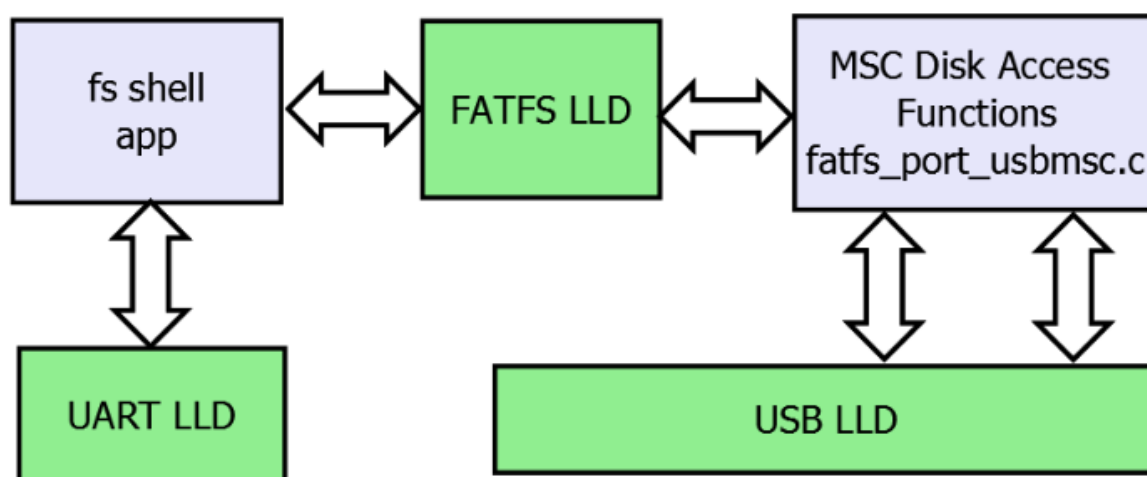


The content of the file: `usb_msc_structs.c` can be replaced with customer USB device information (PID/VID, device names, etc.)

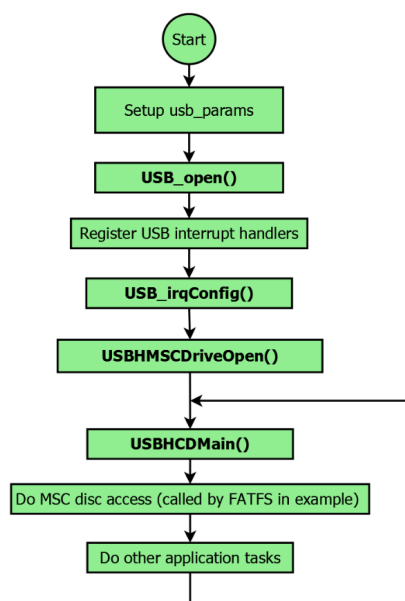
- **USB Host MSC**

The LLD also provides a USB host MSC example. The USB LLD is acting as a USB host, waiting for a USB thumb drive/memory stick to be plugged in. A console with a simple shell command is also provided so that the demo example can display and manipulated content of the USB device.

The following is how the USB host MSC example demo is organized:



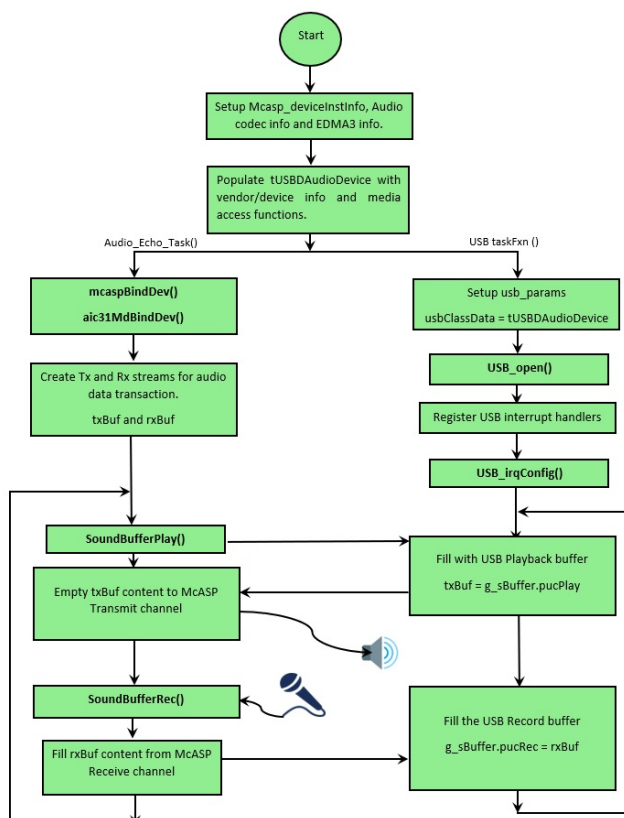
The following is the sequence of the APIs that were used:



• USB Device Audio

The example application code uses the USB library, configures it as a USB device with Audio class function. USB LLD along with the application enumerates as the USB audio class device and allows accessing the audio ports on the EVM from USB host. It supports audio playback and record operations.

Below diagram is the sequence of API calls that starts the USB device audio application. All USB events are handled internally in the LLD and in the interrupt context.



Examples

Bare-metal and RTOS/BIOS examples of the USB device and USB host MSC are provided with the USB LLD. RTOS/BIOS examples are provided for USB device audio class. Please see the readme.txt in the docs directory for more info on how to build the example/demo application. Refer SDK release notes for supported EVMs.

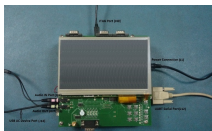
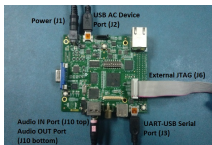
Example Name	EVM's supported	Notes
usb_dev_msc_mmcsd	AM335GP EVM	
usb_dev_msc	AM335GP, AM437xGP, AM57xIDK, AM572GP EVM, K2G EVM, OMAP137EVM, OMAPL138LCDK	OMAPL13 LCDK host and device examples share the same USB port.
usb_host_msc	AM335GP, AM437xGP, AM57xIDK, K2G EVM, OMAP137EVM, OMAPL138LCDK	OMAPL13 LCDK host and device examples share the same USB port. Need OTG cable for EVM with OTG port to work in host mode. USB3.0 host supported on AM572IDK.
usb_device_audio	AM335x EVM,OMAPL137 EVM,OMAPL138 LCDK	Refer to Hardware Setup and How to Run the Demo sections below

Hardware Setup

This section provides the specific HW setup required to run the USB examples.

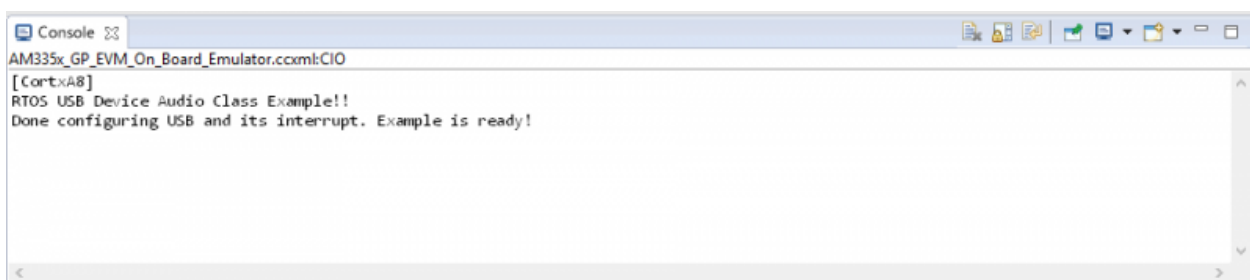
• USB Device Audio

USB audio class demo requires additional setup for running playback and record operations. Below sections provide the setup details for each platform supported.

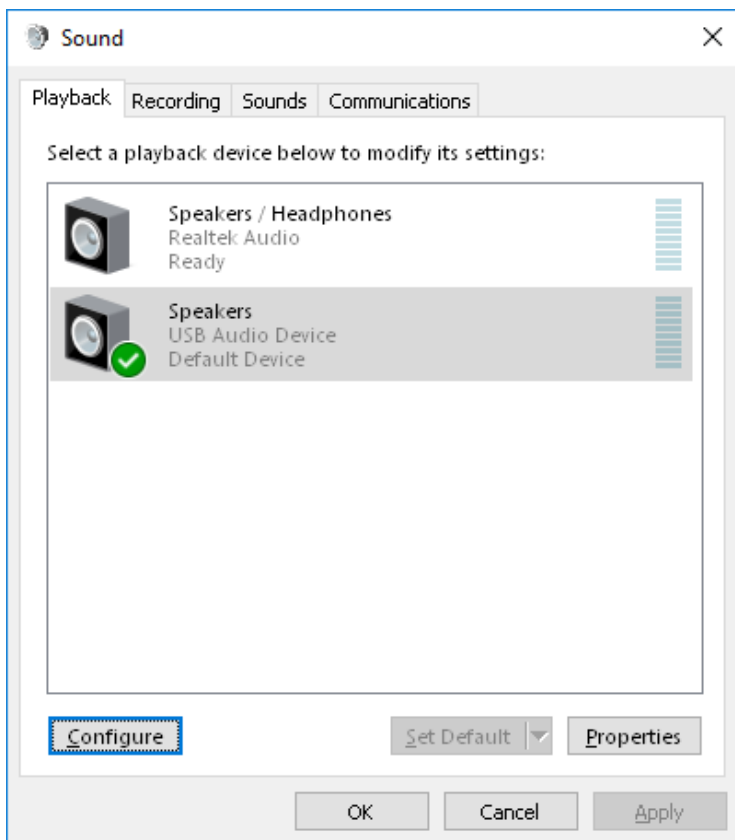
AM335x GP EVM	OMAPL137 EVM	OMAPL138 LCDK
		

How to Run the Demo

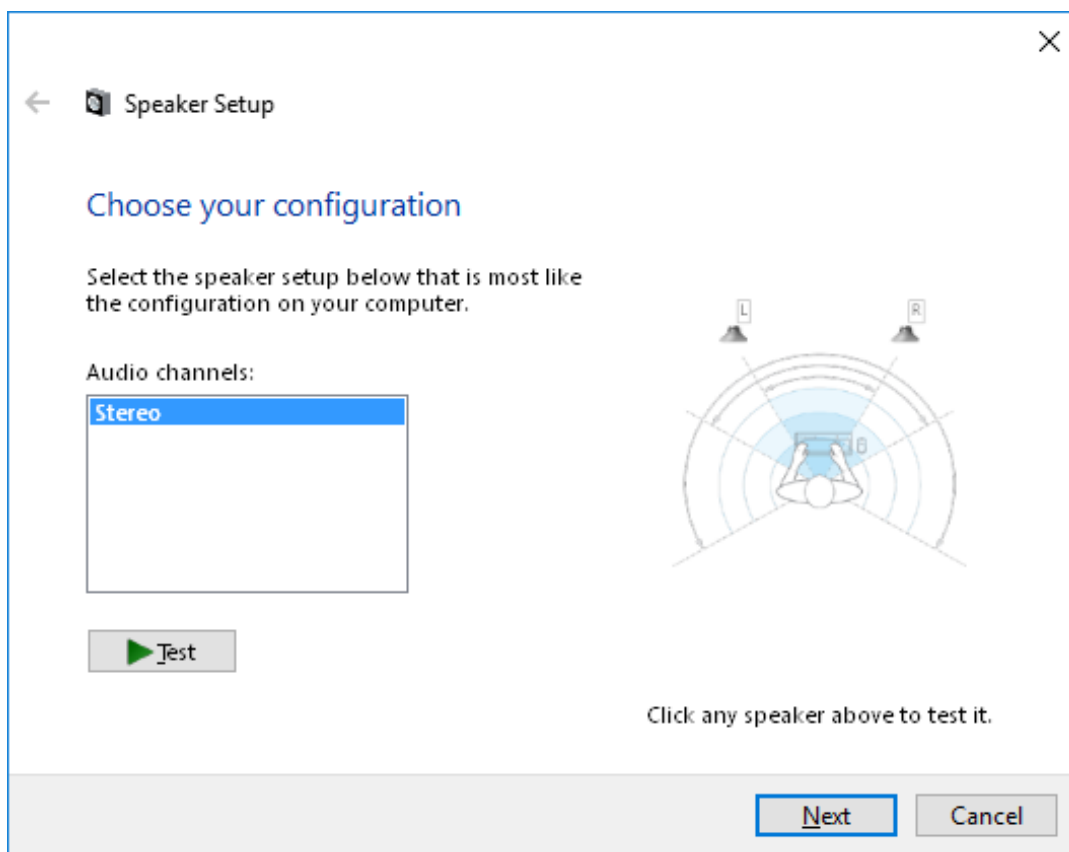
- Follow this link (http://processors.wiki.ti.com/index.php/Processor_SDK_RTOS_Setup_CCS) to get target configuration setup correctly.
- Use CCS to import the USB_DevAudio_<board>_<core>ExampleProject under pdk_<platform>_<version>/packages/MyExampleProjects.
- Build the imported project. the OUT file will be at pdk_<platform>_<version>/packages/MyExampleProjects/USB_DevAudio_<board>_<core>ExampleProject/Debug.
- Make the HW connections as shown in Hardware Setup section
- Launch the target configuration for the EVM from CCS 7.x.
- Connect to ARM or DSP core as applicable.
- Load the pdk_<platform>_<version>/packages/MyExampleProjects/USB_DevAudio_<board>_<core>ExampleProject/Debug/USB_DevA
- Run the program (loaded previously) by pressing F8
- The CCS ConsoleIO will display the following:



- Right click on the "Speaker Icon" on the USB Host (right side of the toolbar), then select "Playback devices"
- Wait until the "Speakers USB Audio Device" shows up in the "Sound" dialog



- Select the "Speakers USB Audio Device" in the "Sound" dialog, then click the "Configure"



- Click the "Test" in "Speaker Setup", you should hear the testing tone in the headphone connected to the EVM

Note:

'board' can be evmAM335x, evmOMAPL137 or lcdkOMAPL138

'core' can be arm or c674x

[end]

Article Sources and Contributors

Processor SDK RTOS USB *Source:* <http://ap-fpdsp-swapps.dal.design.ti.com/index.php?oldid=234327> *Contributors:* A0271499, A0272049, A0850439, RajSivarajan, ThanhTran, X0257344

Image Sources, Licenses and Contributors

Image:USB MSC device.PNG *Source:* http://ap-fpdsp-swapps.dal.design.ti.com/index.php?title=File:USB_MSC_device.PNG *License:* unknown *Contributors:* A0271499

Image:Device mode printout.png *Source:* http://ap-fpdsp-swapps.dal.design.ti.com/index.php?title=File:Device_mode_printout.png *License:* unknown *Contributors:* A0271499

Image:Windows_pdk_usb.png *Source:* http://ap-fpdsp-swapps.dal.design.ti.com/index.php?title=File:Windows_pdk_usb.png *License:* unknown *Contributors:* ThanhTran

Image:usb_device_mmcsc.PNG *Source:* http://ap-fpdsp-swapps.dal.design.ti.com/index.php?title=File:Usb_device_mmcsc.PNG *License:* unknown *Contributors:* ThanhTran

Image:USB MSC host.PNG *Source:* http://ap-fpdsp-swapps.dal.design.ti.com/index.php?title=File:USB_MSC_host.PNG *License:* unknown *Contributors:* A0271499

Image:Host shell screen shot.png *Source:* http://ap-fpdsp-swapps.dal.design.ti.com/index.php?title=File:Host_shell_screen_shot.png *License:* unknown *Contributors:* A0271499

Image:am335x_usb_ac_bd.jpg *Source:* http://ap-fpdsp-swapps.dal.design.ti.com/index.php?title=File:Am335x_usb_ac_bd.jpg *License:* unknown *Contributors:* ThanhTran, X0257344

Image:USB MSC device API flow.PNG *Source:* http://ap-fpdsp-swapps.dal.design.ti.com/index.php?title=File:USB_MSC_device_API_flow.PNG *License:* unknown *Contributors:* A0271499

Image:USB MSC device example blocks.PNG *Source:* http://ap-fpdsp-swapps.dal.design.ti.com/index.php?title=File:USB_MSC_device_example_blocks.PNG *License:* unknown *Contributors:* A0271499

Image:USB MSC host example blocks.PNG *Source:* http://ap-fpdsp-swapps.dal.design.ti.com/index.php?title=File:USB_MSC_host_example_blocks.PNG *License:* unknown *Contributors:* A0271499

Image:USB MSC host API flow.PNG *Source:* http://ap-fpdsp-swapps.dal.design.ti.com/index.php?title=File:USB_MSC_host_API_flow.PNG *License:* unknown *Contributors:* A0271499

Image:USB Audio class flowchart.jpg *Source:* http://ap-fpdsp-swapps.dal.design.ti.com/index.php?title=File:USB_Audio_class_flowchart.jpg *License:* unknown *Contributors:* X0257344

Image:am335x_usb_ac_setup_mod.jpg *Source:* http://ap-fpdsp-swapps.dal.design.ti.com/index.php?title=File:Am335x_usb_ac_setup_mod.jpg *License:* unknown *Contributors:* X0257344

Image:omap1137_usb_ac_setup.jpg *Source:* http://ap-fpdsp-swapps.dal.design.ti.com/index.php?title=File:Omap1137_usb_ac_setup.jpg *License:* unknown *Contributors:* X0257344

Image:omap1138_usb_ac_setup.jpg *Source:* http://ap-fpdsp-swapps.dal.design.ti.com/index.php?title=File:Omap1138_usb_ac_setup.jpg *License:* unknown *Contributors:* X0257344

File:CCS_console_output.png *Source:* http://ap-fpdsp-swapps.dal.design.ti.com/index.php?title=File:CCS_console_output.png *License:* unknown *Contributors:* Mwei

File:Sound.png *Source:* <http://ap-fpdsp-swapps.dal.design.ti.com/index.php?title=File:Sound.png> *License:* unknown *Contributors:* Mwei

File:Speaker_setup.png *Source:* http://ap-fpdsp-swapps.dal.design.ti.com/index.php?title=File:Speaker_setup.png *License:* unknown *Contributors:* Mwei