

---

## Audio playback and recording using the STM32F4DISCOVERY

---

### 1 Introduction

This application note describes the audio (wave) playback and recording application based on the STM32F4xx microcontroller and the STM32F4-DISCOVERY board.

The audio data (wave) can be read from the internal Flash memory of the STM32F4xx microcontroller or on an external USB key (through USB FS core in Host mode). The recorded wave can be stored only in the external USB key.

The recording process is based on ST MP45DT02 MEMS microphone hardware with a PDM audio software decoding Library (converting PDM data produced by the microphone to PCM data stored in the USB key).

The document is structured as follows:

- A description of the principles of the audio playback and recording firmware and how to run the firmware demonstration are provided in [Section 2: Application overview](#).
- [Section 3](#) describes how to get started with the software and hardware.

The source code of this application is provided in the "STM32F4-DISCOVERY board firmware package" (V1.1.0 and later) under the following path 'Project\Audio\_playback\_and\_record.

#### 1.1 Reference documents

- STM32F4DISCOVERY high-performance Discovery board data brief
- *Getting started with software and firmware environments for the STM32F4DISCOVERY Kit* (UM1467)
- *PDM audio software decoding on STM32 microcontrollers* (AN3998)
- STM32F40x reference manual (RM0090)
- STM32F405xx STM32F407xx datasheet

The above documents are available at [www.st.com/stm32f4-discovery](http://www.st.com/stm32f4-discovery).

# Contents

- 1 Introduction ..... 1**
  - 1.1 Reference documents ..... 1
  
- 2 Application overview ..... 4**
  - 2.1 Application description ..... 4
  - 2.2 Firmware driver description ..... 6
  - 2.3 Audio playback application ..... 7
    - 2.3.1 Playback from USB key ..... 8
    - 2.3.2 Playback from internal Flash ..... 9
  - 2.4 Audio record application ..... 9
  
- 3 How to use the audio playback and recording application ..... 12**
  - 3.1 System requirements ..... 12
  - 3.2 Running the application ..... 12
  
- 4 Revision history ..... 14**

## List of figures

Figure 1. Audio playback and record architecture ..... 4  
Figure 2. Audio playback and record modules ..... 5  
Figure 3. Audio playback/record firmware driver flowchart ..... 7  
Figure 4. Wave player procedures ..... 8  
Figure 5. Recording application procedure ..... 10  
Figure 6. Microphone connection ..... 11  
Figure 7. Hardware environment ..... 12

## 2 Application overview

### 2.1 Application description

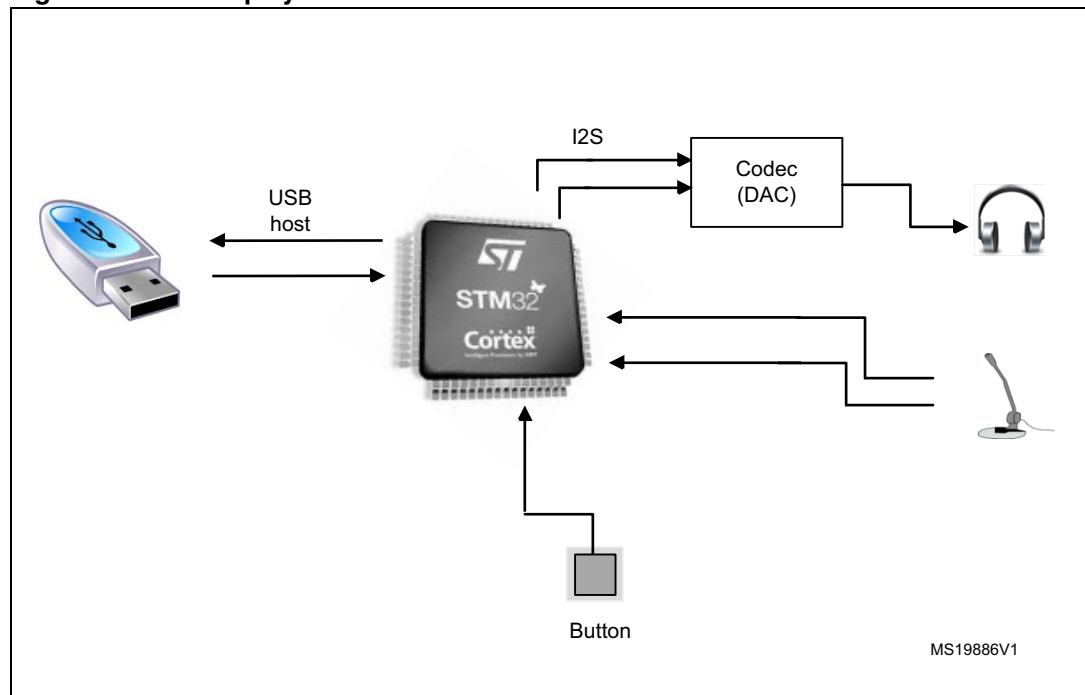
The audio playback and record applications support two types of mass storage media. They can play audio data in the internal Flash of the microcontroller or on an external USB key, and record data only to an external USB key. This is selected by defines in the main.h file. In the workspace toolbar, select the project configuration:

- MEDIA\_IntFLASH
- MEDIA\_USB\_KEY

The firmware driver can:

- Play a stored wave from an external USB key or internal Flash
- Record a wave in an external USB key
- Switch from play to record

**Figure 1. Audio playback and record architecture**



This application is based on a STM32F4xx device and an STM32F4-DISCOVERY board.

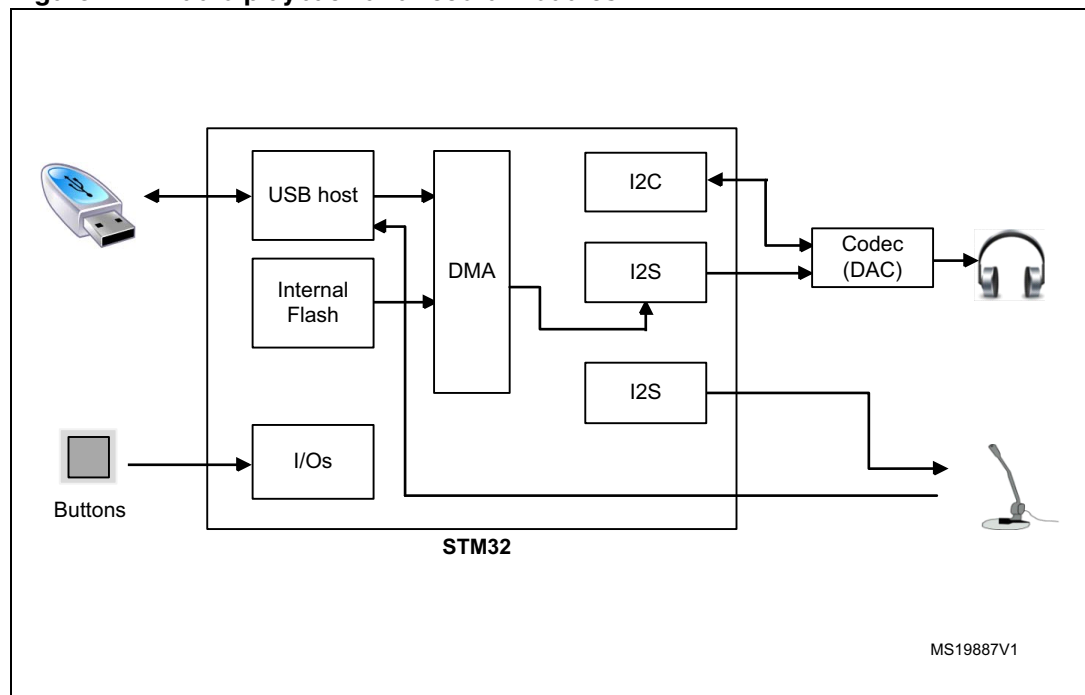
The main features of the application are:

- MEMS microphone
- Audio Codec DAC
- Headphone
- USB key (if this the storage media used)
- MEMS accelerometer

The main STM32 modules used by this application are:

- **USB peripheral:** configured in Host mode. Mass Storage Class (MSC) is implemented to transmit and receive audio data from/to USB key.
- **I2S peripheral:** configured in Master Transmitter mode and used to transmit audio data to the external audio codec (DAC). It is also used as a Master Receiver as an input clock for the MEMS microphone.
- **DMA:** is used to transmit data from the buffers to the I2S peripheral. This significantly reduces the CPU load.
- **I2C peripheral:** is used to control several external devices like the audio Codec and to obtain data from this device.
- **SPI peripheral:** used to control the MEMS accelerometer.
- **User button:** is used to monitor the applications (playback or record).

**Figure 2. Audio playback and record modules**



*Note: This application note is based on the STM32 USB On-The-Go (OTG) host and device library. For more details about the USB Host stack and a mass storage demonstration, please refer to the “STM32F105/7, STM32F2xx and STM32F4xx USB On-The-Go host and device library” user manual (UM1021).*

## 2.2 Firmware driver description

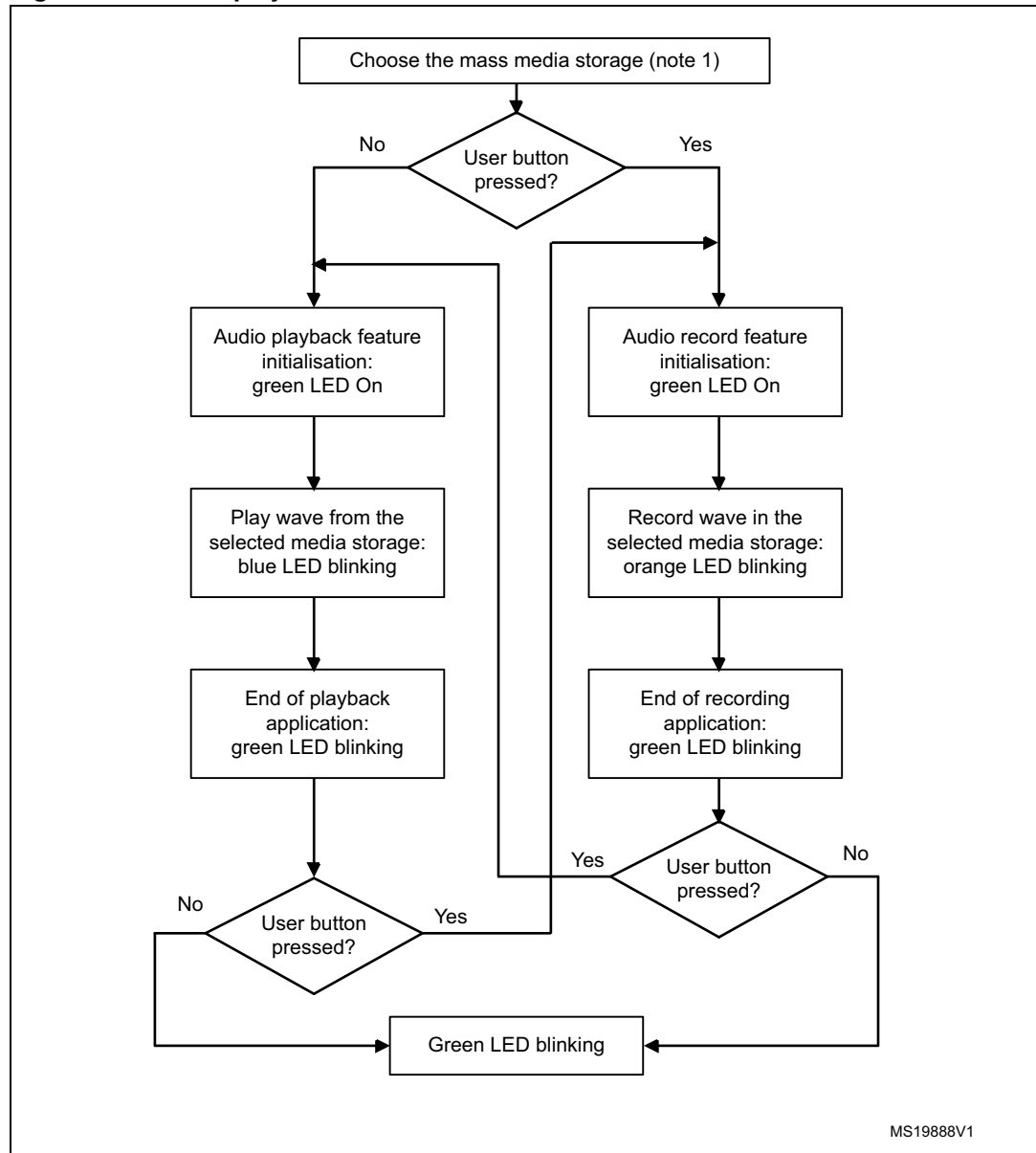
This application contains the following set of source files:

- **main.c:** contains the initialization code and starts the application depending on the selected 'MEDIA\_IntFLASH' or 'MEDIA\_USB\_KEY' configuration.
- **stm32f4xx\_it.c:** contains the interrupt handlers for the application.
- **waveplayer.c:** implements the functions used for playback.
- **waverecorder.c:** implements the functions used for record.
- **usb\_bsp.c:** implements the board support package for the USB host library.
- **usbh\_usr.c:** includes the USB host library user callbacks.

After each board reset, the wave player application runs from the selected mass storage media.

- If the selected media is USB key, if the user button is pressed, the playback application is stopped and the application switches to recording. Each time the user button is pressed, it stops the running application and switches to executing the other one.
- If the selected media is internal Flash, pressing the user button has no effect.

Figure 3. Audio playback/record firmware driver flowchart



1. If internal Flash is selected as mass storage media, pressing the user button has no effect.

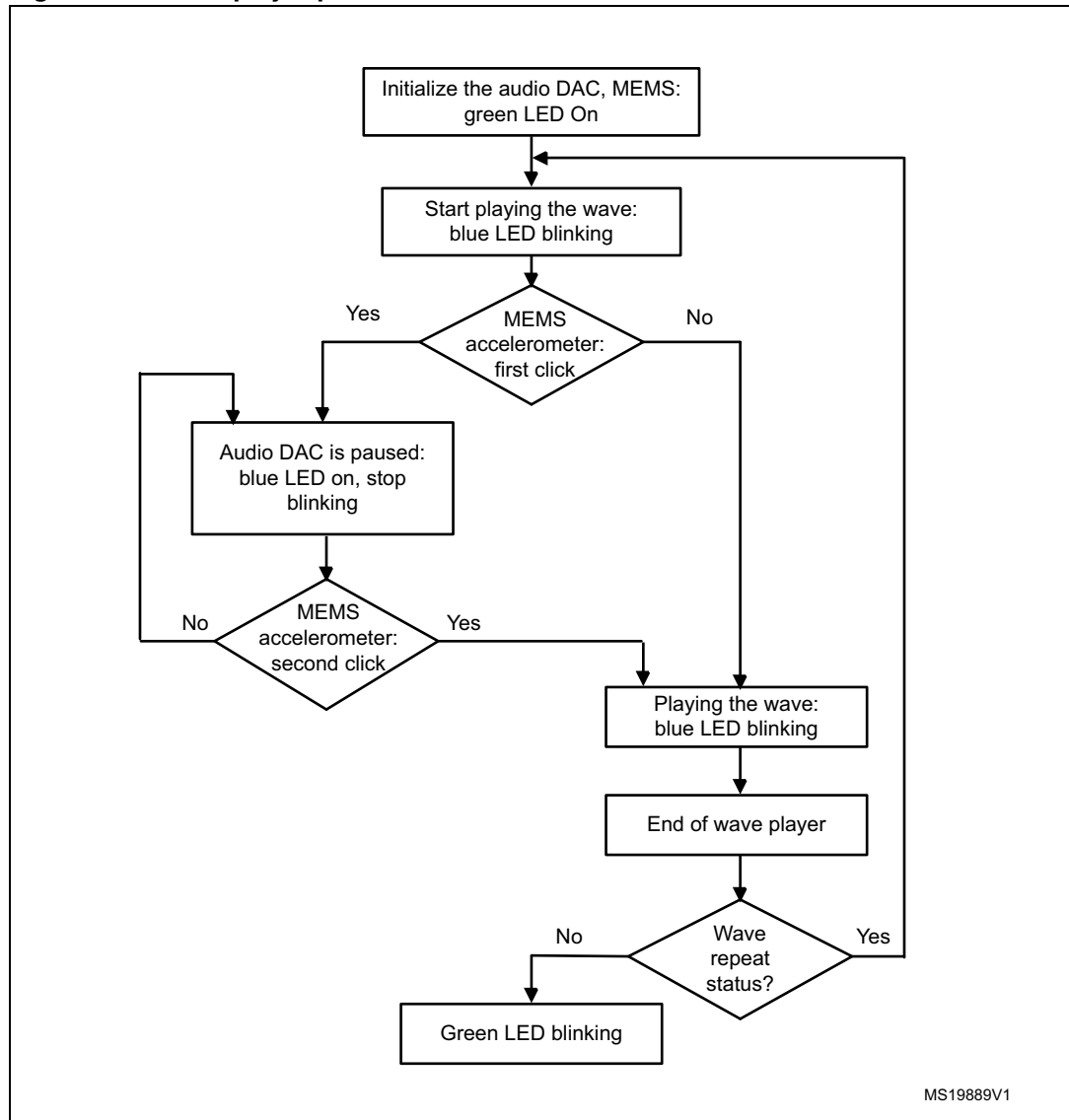
## 2.3 Audio playback application

The flowchart in [Figure 4](#) describes the playback application. It implements several control features like Pause/Resume, Repeat On/Repeat Off.

The MEMS accelerometer is used to support Pause/Resume. When the wave player is running, the first click on the board stops playing and the second click on it resume playing.

The Repeat On/Repeat Off feature is managed with DEFINES in the main.h file.

Figure 4. Wave player procedures



At any time (if USB key is selected as mass storage media), if the user button is pressed, playback is stopped and the record application is executed.

### 2.3.1 Playback from USB key

In this demonstration, any wave file stored on the USB Key can be opened using the FatFs file system and transferred to the internal SRAM block (1024 bytes) using the DMA and the I2S interface.

The voice sampling period is read from the Wave File Header. An audio DAC is connected to the I2S interface to play the stored wave files.

The name of the wave file loaded in the USB Key can be changed by modifying the "WAVE\_NAME" definition in the main.h file. The wave file name must have eleven characters.



This application reads all wave files from the USB Key and displays only the .WAV files that have the following format:

- **Audio format:** PCM (an uncompressed wave data format in which each value represents the amplitude of the signal at the time of sampling)
- **Sample rate:** such 8000, 11025, 16000, 22050, 44100 Hz or 48000 Hz.
- **Bits per sample:** 16 bits (audio sample data values are in the range [0-1024])
- **Number of channels:** 2 (stereo)

The wave from the USB Key is parsed to detect the sample rate in order to configure the I2S accordingly. When the play back begins the blue LED starts toggling.

The playback is managed with double buffering. A first buffer is used to store the wave data retrieved from the USB Key, using the FatFs file system.

Once this buffer is filled:

- The DMA sends its content to the I2S peripheral which transfers it to the external audio codec DAC
- The data from the USB key is stored in a second buffer

Then these two buffers are swapped indefinitely, till end of the playback process.

At any time, if the USB Key is disconnected from the DISCOVERY board, the blue LED is off, the audio DAC is stopped and red LED goes on.

When the USB Key is reconnected again to the DISCOVERY board, the red LED goes off and the last running application starts again.

### 2.3.2 Playback from internal Flash

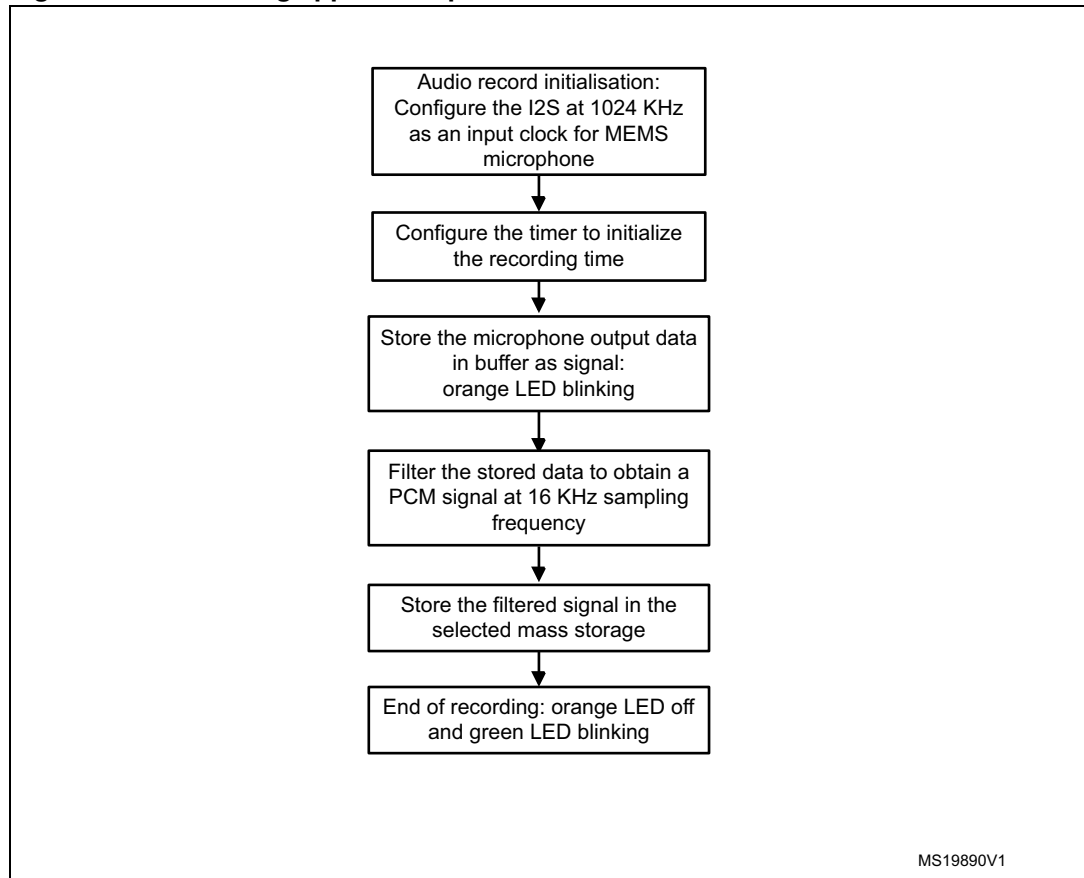
In this demonstration, the wave file is stored in the internal Flash as a const array declared in the audio\_sample.c file.

After a reset, the playback application starts playing the wave stored in the internal Flash after initializing the Audio DAC.

## 2.4 Audio record application

The flowchart in [Figure 5](#) describes the recording application. It is based on a MEMS microphone. Audio record is available only when USB key is selected as mass storage media.

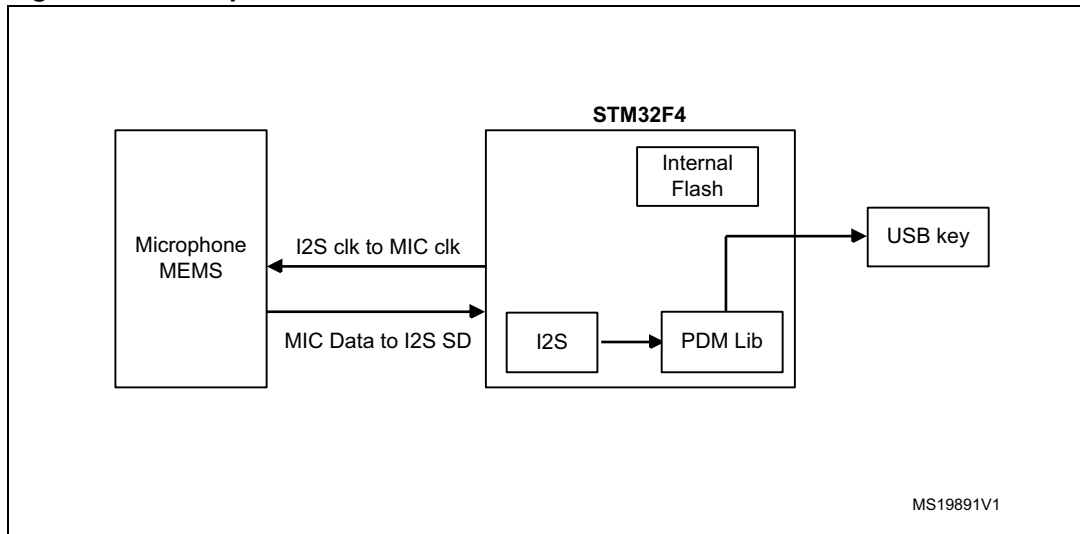
Figure 5. Recording application procedure



The I2S peripheral is configured as master in order to generate the correct clock (1,024 MHz) for the digital microphone. The 1,024 MHz clock is calculated from the output audio streaming (16 KHz) and the decimation factor (64) chosen for the demo (16000 Hz x 64 = 1.024 MHz). (Refer to AN3998 for PDM audio software decoding).

The I2S peripheral is configured to generate an interrupt each time 16 bit samples have been acquired.

Figure 6. Microphone connection



The filtering process uses the PDM audio software decoding library. This library implements several filters for the 1-bit PDM high frequency signal output from a digital microphone and transforms it into a 16-bit PCM at a proper audio frequency.

The filtering process and the write into the USB key are managed with double buffering. When the PDM Library is filtering microphone data, the filtered data is stored in the USB key.

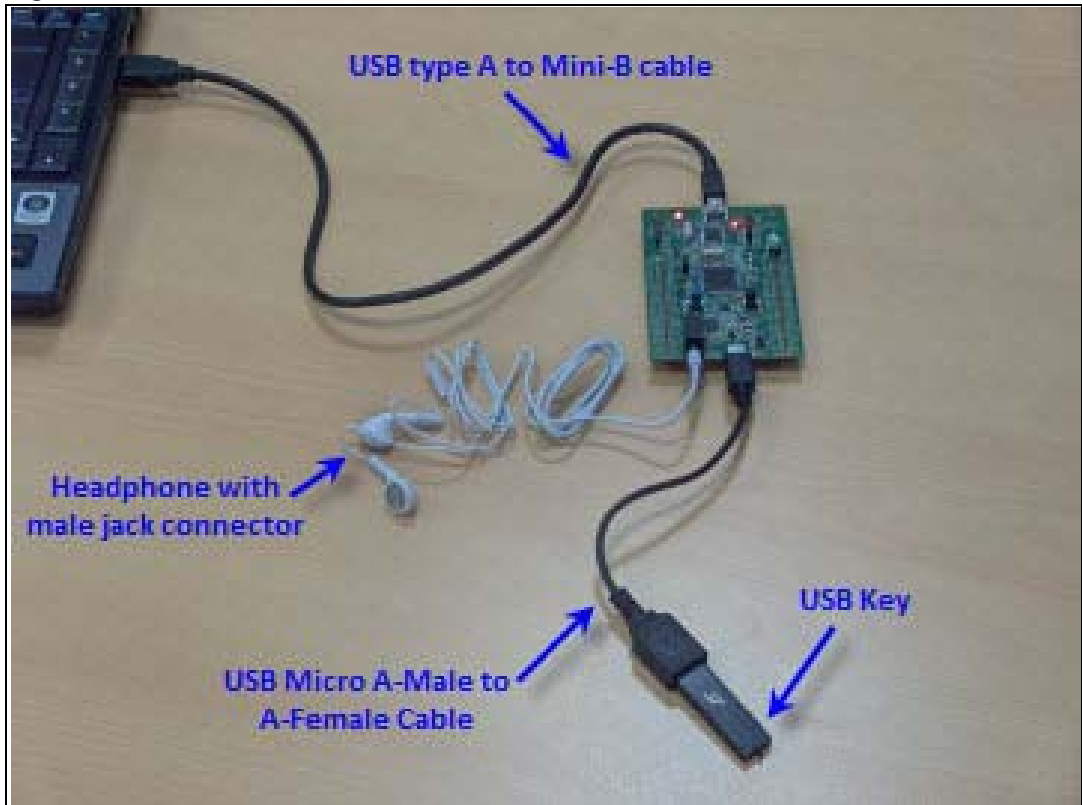
When a USB key is used for mass storage, if the USB key is disconnected, the recording is stopped and a red LED goes on. When it is connected again, the red LED goes off and the recording is started again.

## 3 How to use the audio playback and recording application

### 3.1 System requirements

Before running your application, you should establish the connection with the STM32F4DISCOVERY board as shown in [Figure 7](#).

**Figure 7. Hardware environment**



To run the audio playback and record applications on your STM32F4DISCOVERY board, the minimum requirements are as follows:

- Windows PC (2000, XP, Vista, 7)
- 'USB type A to Mini-B' cable, used to power the board (through USB connector CN1) from a host PC and to connect to the embedded ST-LINK/V2 for debugging and programming.
- USB 'micro A male to B female' cable, used to connect the USB key (through USB connector CN5) as a USB Device to the STM32F4xx's Host.
- Headphone with male jack connector.

### 3.2 Running the application

To run the Audio playback and recording application, you should proceed as follows:

1. Program the firmware upgrade application in the internal Flash memory
2. Open the project (under Project\Audio\_playback\_and\_record) with a toolchain of your choice
3. Select "MEDIA\_IntFLASH" or "MEDIA\_USB\_KEY" workspace depending on the mass storage media memory used.
4. Compile, load it into target memory and run it.

Another option is to use the embedded Bootloader or any in-system programming tool to reprogram this application easily.

- Use STM32F4-Discovery\_Audio\_USB\_V1.0.0.hex or STM32F4-Discovery\_Audio\_IntFLASH\_V1.0.0.hex with any "in-system programming tool" such as STM32 ST-LINK Utility
- Use STM32F4-Discovery\_Audio\_USB\_V1.0.0.dfu or STM32F4-Discovery\_Audio\_IntFLASH\_V1.0.0.dfu with "DFUUse\DFUUse Demonstration" tool

For more details please refer to section 4 of UM1467.

If you select "MEDIA\_USB\_KEY" configuration, you should proceed as follows:

1. Load a wave file to the root of a USB key, this file should have the format described in [Section 2.3.1: Playback from USB key](#).
2. Plug the USB key into the STM32F4DISCOVERY board through 'USB micro A-Male to A-Female' cable (connector CN5)
3. Then follow the description provided in [Section 2.2: Firmware driver description](#)

## 4 Revision history

**Table 1. Document revision history**

Date	Revision	Changes
27-Oct-2011	1	Initial release.

**Please Read Carefully:**

Information in this document is provided solely in connection with ST products. STMicroelectronics NV and its subsidiaries ("ST") reserve the right to make changes, corrections, modifications or improvements, to this document, and the products and services described herein at any time, without notice.

All ST products are sold pursuant to ST's terms and conditions of sale.

Purchasers are solely responsible for the choice, selection and use of the ST products and services described herein, and ST assumes no liability whatsoever relating to the choice, selection or use of the ST products and services described herein.

No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted under this document. If any part of this document refers to any third party products or services it shall not be deemed a license grant by ST for the use of such third party products or services, or any intellectual property contained therein or considered as a warranty covering the use in any manner whatsoever of such third party products or services or any intellectual property contained therein.

**UNLESS OTHERWISE SET FORTH IN ST'S TERMS AND CONDITIONS OF SALE ST DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY WITH RESPECT TO THE USE AND/OR SALE OF ST PRODUCTS INCLUDING WITHOUT LIMITATION IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE (AND THEIR EQUIVALENTS UNDER THE LAWS OF ANY JURISDICTION), OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.**

**UNLESS EXPRESSLY APPROVED IN WRITING BY TWO AUTHORIZED ST REPRESENTATIVES, ST PRODUCTS ARE NOT RECOMMENDED, AUTHORIZED OR WARRANTED FOR USE IN MILITARY, AIR CRAFT, SPACE, LIFE SAVING, OR LIFE SUSTAINING APPLICATIONS, NOR IN PRODUCTS OR SYSTEMS WHERE FAILURE OR MALFUNCTION MAY RESULT IN PERSONAL INJURY, DEATH, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE. ST PRODUCTS WHICH ARE NOT SPECIFIED AS "AUTOMOTIVE GRADE" MAY ONLY BE USED IN AUTOMOTIVE APPLICATIONS AT USER'S OWN RISK.**

Resale of ST products with provisions different from the statements and/or technical features set forth in this document shall immediately void any warranty granted by ST for the ST product or service described herein and shall not create or extend in any manner whatsoever, any liability of ST.

ST and the ST logo are trademarks or registered trademarks of ST in various countries.

Information in this document supersedes and replaces all information previously supplied.

The ST logo is a registered trademark of STMicroelectronics. All other names are the property of their respective owners.

© 2011 STMicroelectronics - All rights reserved

STMicroelectronics group of companies

Australia - Belgium - Brazil - Canada - China - Czech Republic - Finland - France - Germany - Hong Kong - India - Israel - Italy - Japan - Malaysia - Malta - Morocco - Philippines - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States of America

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

