Getting started with osxAcousticEC real-time acoustic echo cancellation software expansion for STM32Cube

Introduction

osxAcousticEC software implements a real-time echo cancellation routine based on the SPEEX implementation of the MDF algorithm. The ancillary features of the SPEEX suite, namely the preprocess options and the Automatic Gain Control, has been included as well. Parameters can be modified at runtime, to grant adaptation to the varying environment conditions.

The osxAcousticEC library is integrated in binary format in a software package providing examples running on STM32F407 Discovery board (STM32F4DISCOVERY).

The example package is designed as an add-on for the standard STM32CubeF4 package; the library can be easily ported to any STM32F4 microcontroller with FPU unit.

The software is based on STM32Cube technology. Information about STM32Cube is available on www.st.com at *http://www.st.com/stm32cube*.

Contents

List of figures

1 Licensing information

This software library is licensed under ST OpenSoftwareX Limited License Agreement, (the "OSX-LLA License"). You may obtain a copy of the OSX License at:

OpenSoftwareX license agreement

Some of the library code is based on the CMSIS DSP software library by ARM, a suite of common signal processing functions for use on Cortex-M processor based devices. Licensing terms are available in the attached release_note.html file, in the next lines of this document and on the web at:

http://www.keil.com/pack/doc/CMSIS/DSP/html/index.html.

ARM license note:

Copyright (C) 2009-2012 ARM Limited. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of ARM nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL COPYRIGHT HOLDERS AND CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Some of the library code is based on the open Source patent-free SPEEX software, by Jean-Marc Valin/Xiph.Org Foundation, released under "revised BSD" license. Licensing terms are available in this document, in the release note for this software, in the header file of this software and on the web at:

http://www.xiph.org/licenses/bsd/speex/

SPEEX revised BSD license note:

© 2002-2003, Jean-Marc Valin/Xiph.Org Foundation.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the Xiph.org Foundation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

2 osxAcousticEC software library

osxAcousticEC software provides a real-time echo cancellation routine based on the SPEEX (*http://www.speex.org/*) implementation of the MDF algorithm.

3 osxAcousticEC software expansion for STM32CubeF4

3.1 Overview

osxAcousticEC software package expands the functionality of STM32Cube. It is an add-on for the standard STM32CubeF4 package and exploits its features for digital MEMS microphone acquisition and audio output with an STM32F407 Discovery board (STM32F4DISCOVERY).

For details on how to merge the two packages into a single package, refer to *Section 6.2: "Firmware and software"*.

The key features of this add-on package are:

• osxAcousticEC library middleware
• USB Audio input device class middleware.
• sample implementation available for the STM32F407 Discovery board (STM32F4DISCOVERY)
• detailed chm documentation of the library API

3.2 Architecture

The package extends the STM32CubeF4 by providing:

• an additional middleware component for the osxAcousticEC library
• a sample implementation running on the STM32F407 discovery board (STM32F4DISCOVERY)

Thus two packages must be merged in order to obtain a fully functional application based on the osxAcousticEC library, available to the user as a reference for software design and library integration.

The resulting application architecture includes the following layers, needed to access the microphones data:

• **STM32Cube HAL layer**: consists of a set of simple, generic, multi-instance APIs (application programming interfaces) which interact with the upper layer applications, libraries and stacks. These generic and extension APIs are based on a common framework which allows any layers they built on, such as the middleware layer, to implement their functions without requiring specific hardware information for a given microcontroller unit (MCU). This structure improves library code reusability and guarantees easy portability across other devices.
• **Board Support Package (BSP) layer**: provides software support for the STM32 Nucleo board peripherals, excluding the MCU. These specific APIs provide a programming interface for certain board specific peripherals like LEDs, user buttons, etc and can also be used to fetch individual board version information. It also provides support for initializing, configuring and reading data.

Figure 1: Resulting software architecture
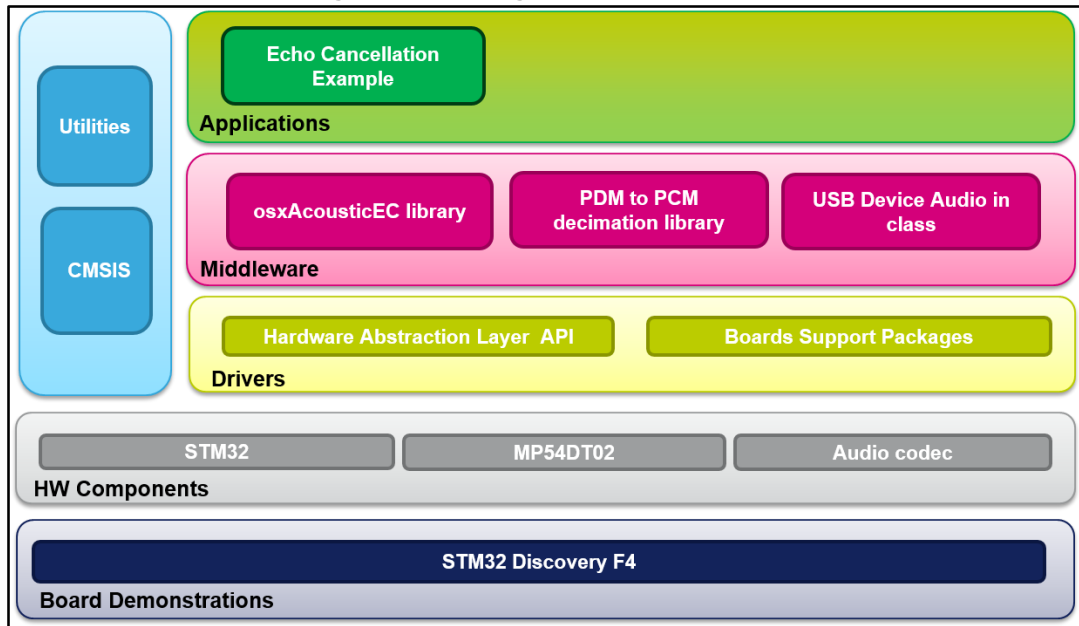
3.3 Folder structure

The following folders are included in the package:

Documentation: this folder contains a compiled HTML file generated from the source code and describing in details the software components and APIs.

Drivers: this folder contains the HAL drivers, the board specific drivers for each supported board or hardware platform, including the on-board components ones and the CMSIS layer which is a vendor-independent hardware abstraction layer for the Cortex-M processor series.

Middlewares: this folder contains the osxAcousticEC library binary code, documentation and license information.

Projects: this folder contains a sample application used to demonstrate library, provided for the STM32Discovery platform with three development environments (IAR Embedded Workbench for ARM, RealView Microcontroller Development Kit (MDK-ARM) and System Workbench for STM32 (SW4STM32).

Figure 2: Package folder structure
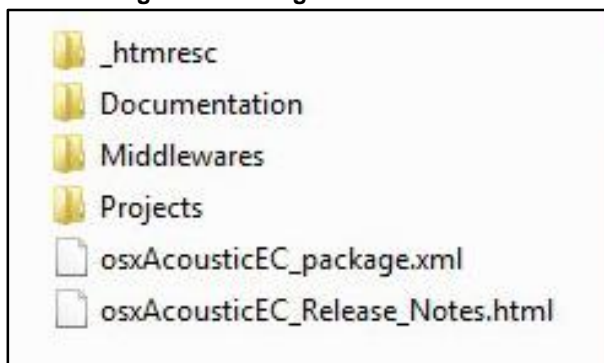
3.4 APIs

Detailed technical information fully describing the functions and parameters of the osxAcousticEC APIs can be found in the osxAcousticEC_Package.chm compiled HTML file located in the Documentation folder of the software package. In the same document, you can find general descriptions regarding all the osxAcoustic library APIs and the concepts behind their design.

The osxAcousticEC is provided as a node-locked library which allows derivative firmware images to run on a specific STM32 device only. Licensing activation codes must be requested from ST and included in the project (and become part of the build process) prior to attempting its usage. The resulting firmware binary image will therefore be node-locked.

For complete information about the open.AUDIO license agreement, please refer to the license file located in the Middlewares/ST/ STM32_OSX_AcousticEC_Library folder.

4 System setup guide

4.1 Hardware setup

The library sample application requires the following hardware:

- STM32F4DISCOVERY board.

Further details are available at www.st.com

4.2 Firmware and software

The main steps required to setup the environment are:

1. Additional package download and unpack
2. Library package installation
3. Node locking procedure
4. Compiling and running

4.2.1 Additional package download and unpack

The application is designed as an add-on project for the STM32CubeF4 package, which you can find at www.st.com.

After downloading, unpack the zip file in a directory of your choice. This will become the workspace used to test and work with the library. Avoid using paths names longer than 240 characters because it can cause problems in some tool chains during project build. In our example below, the workspace is placed in the "C:\Workspace" folder.

Figure 3: Directory content after unpacking the STM32CubeF4 package
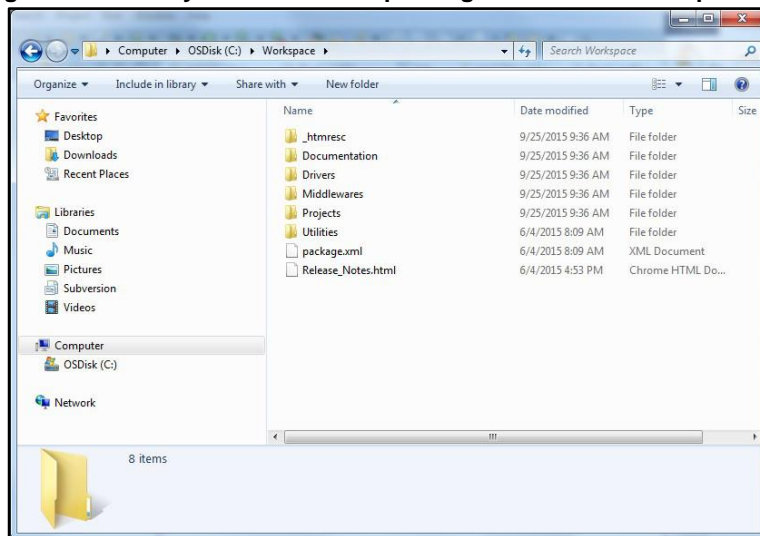
4.2.2 Library package installation

Double click the "osxAcousticEC_Setup" library installer available at *www.st.com* in order to start the installation procedure.
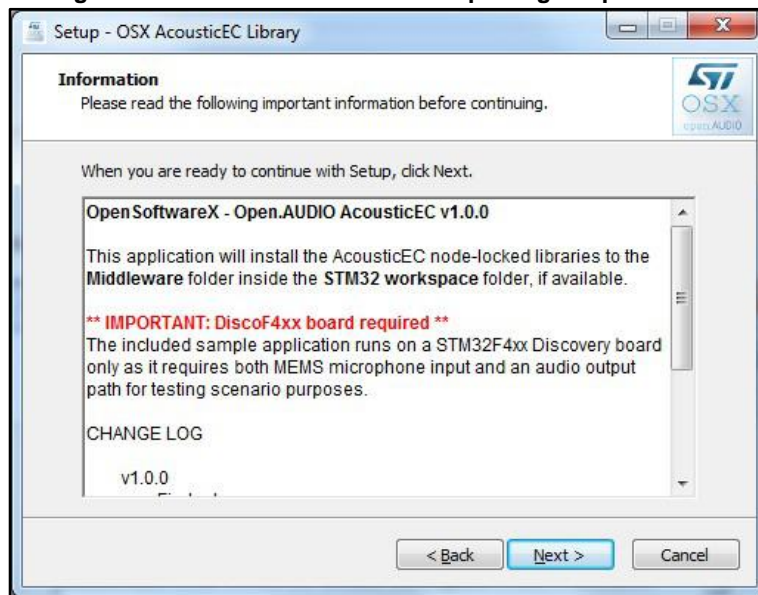
The first step is to accept the License Agreement:

Figure 4: License Agreement

Then the procedure details the additional packages required:

Figure 5: Information and additional package requirements

Now you can choose the directory where you want to install the library software. Note that this is not the path for the firmware, but where the PC software related information (uninstall and other information) shall be stored.

This folder does not necessarily have to be your workspace folder.

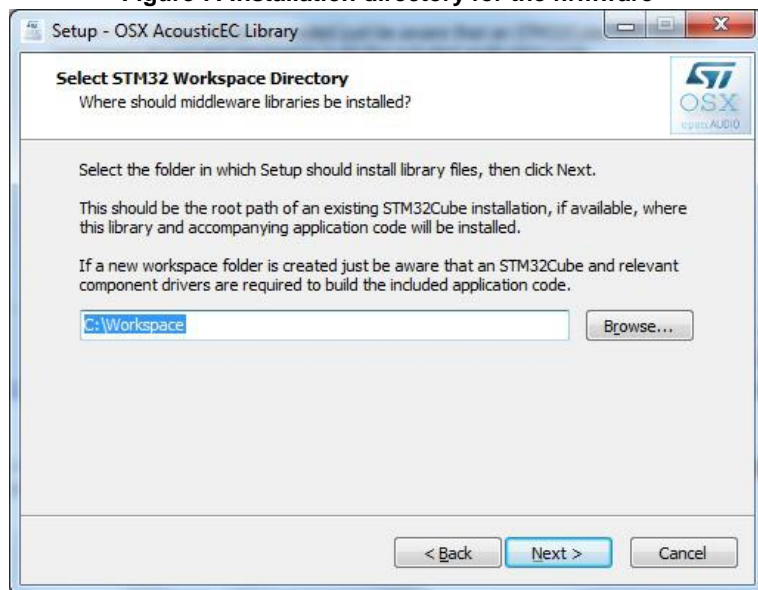**Figure 6: Installation directory for the PC software**
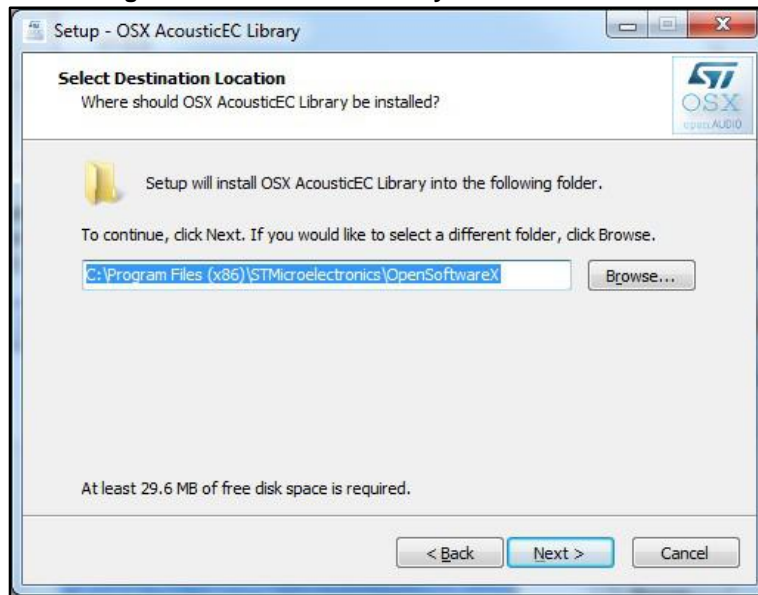
Then you choose the directory where the firmware will be unpacked.

> This folder must be your desired workspace directory ("C:\Workspace" in our example).

Figure 7: Installation directory for the firmware

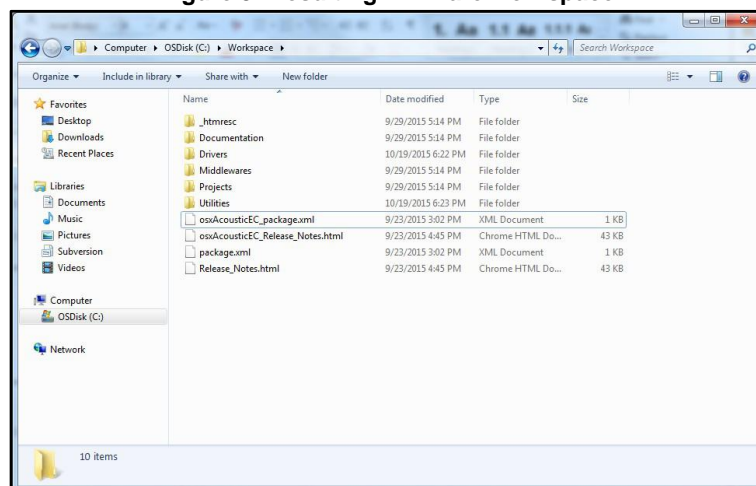Finally you can complete the installation procedure.

> The LicenseWizard software is required for the next steps.

Figure 8: OSX LicenseWizard tool installation

At this stage, your Workspace folder should resemble the image below.

**Figure 9: Resulting firmware workspace**

You will find the installed library in the Middleware folder, while the sample application projects are in the Project folder.

4.2.3   Node locking procedure

Connect a STM32DiscoveryF4 board to the PC, run the OSX LicenseWizard you installed in the previous steps and choose the library you want to activate.
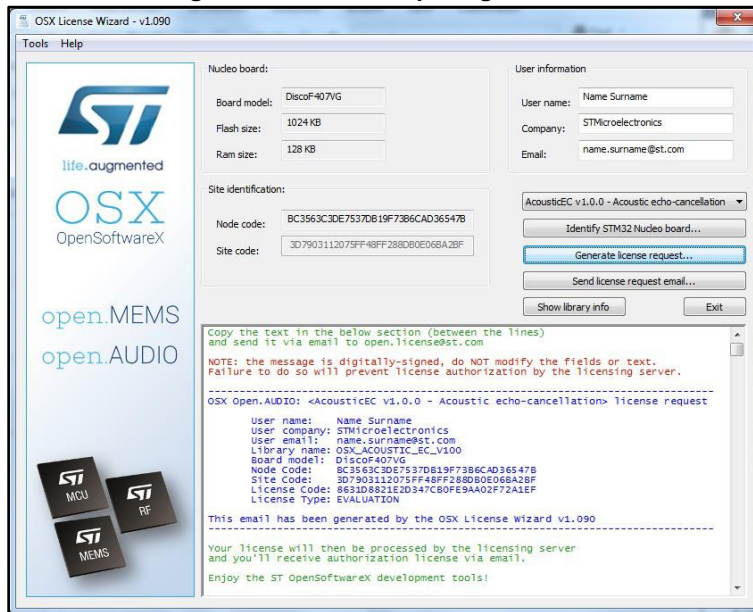
Figure 10: Library selection
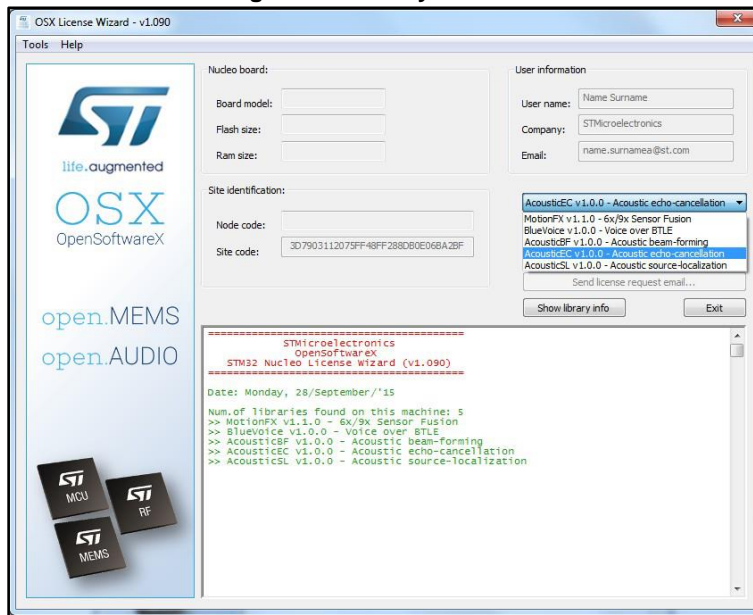
Now use the appropriate buttons on the wizard to:

- identify the STM32 Nucleo board
- generate the license request
- send the request e-mail
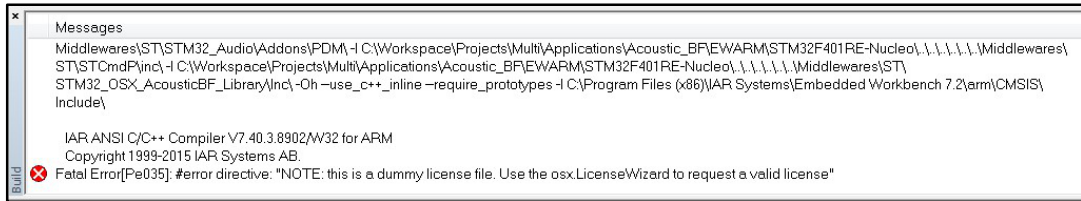
Figure 11: License request generation

The license mail should arrive after a few moments.

4.2.4   Compiling the sample projects

Open the project with your favorite tool chain. In this example, the projects for three IDEs are located in "C:\Workspace\Projects\Multi\Applications\Acoustic_EC".

Try to compile the project, you should receive the following error:

Figure 12: Error while building the original osx_license.h file

Open the osx-license.h file and substitute the dummy license array with the one you received by mail from the Open.Audio server. Also, comment the define directive depicting the error.

Figure 13: osx_license.h file after the required modifications
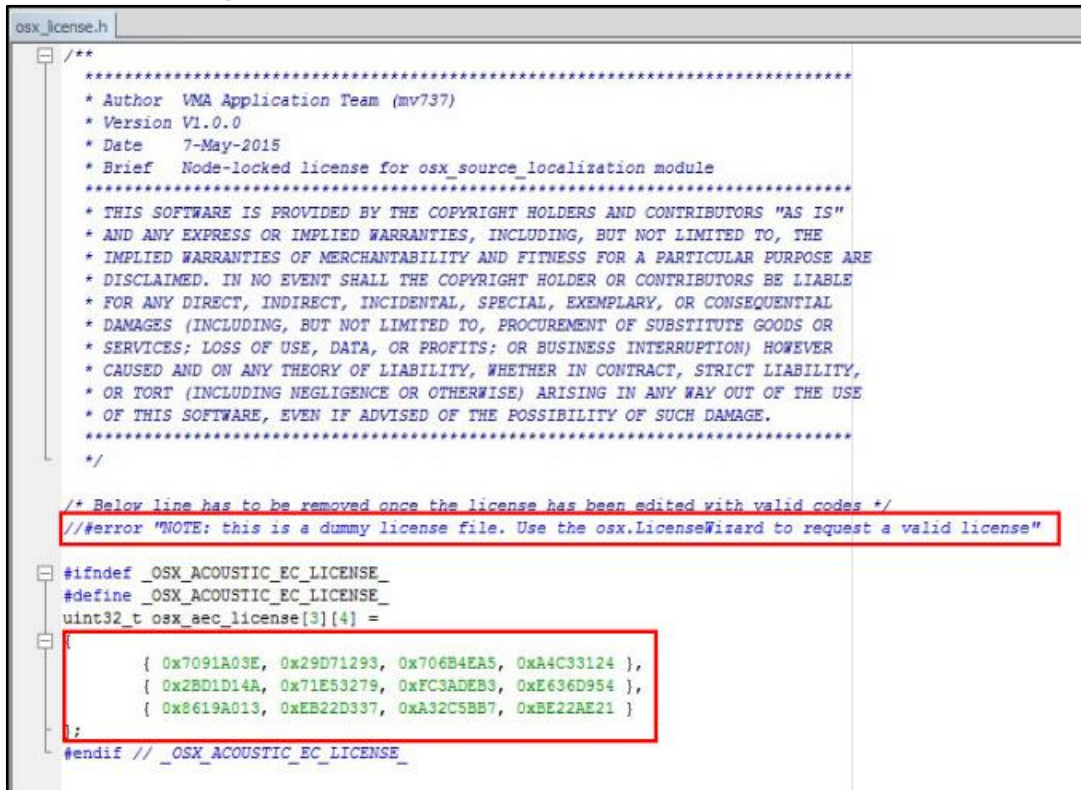
You should be now able to compile and flash the board for which you requested the license in order to use the library example.

5 Sample application description

A sample application using the osxAcousticEC library with the STM32F407 Discovery board (STM32F4DISCOVERY) is provided in the Projects directory. Ready to be built projects are available for multiple IDEs.

The application is designed to perform:

- Acquisition of the on board digital MEMS microphone
- Output of an audio fragment (saved in the STM32 FLASH) using the onboard codec
- Echo cancellation with the following inputs:
    - a primary microphone signal
    - the known audio fragment as the reference signal
- Streaming of two audio channels via USB to a host PC. The two audio streams contain:
    - the algorithm output (first channel)
    - a reference omnidirectional microphone (second channel).

5.1 Firmware steps

Follow these steps to obtain the desired behavior:

- Initialize and start the USB audio input driver and middleware. This allows a host PC to recognize the device as a standard multichannel USB microphone.
- Initialize microphones acquisition using the relevant BSP function.
- Initialize audio output using the relevant BSP function
- Initialize osxAcousticEC library.
- Start the audio acquisition that will trigger the library execution.
- Processed data as well as the reference omnidirectional microphone streams are passed to USB driver every millisecond.

All the operations related to the osxAcousticEC library are performed in dedicated functions in the "audio_application.c" file.

Further details about the library API can be found in the .chm help file in the Documentation folder.

5.2 Example program execution

In order to record audio from the device and evaluate beamforming, you need to install third party software, like the Audacity® freeware program, to save or play the streamed signal.

Detailed information on how to set up the host system for audio recording can be found in the X-CUBE-MEMSMIC1 user manual, available at www.st.com

Further information about Audacity® can be obtained at: *http://audacityteam.org/?lang=en*

6   Library profiling

Profiling was performed in order to evaluate the library resource consumption in terms of MIPS, RAM and Flash. Detailed information can be found in the osxAcousticEC_Package.chm compiled HTML file located in the Documentation folder.

7 Revision history

Table 1: Document revision history

Date	Version	Changes
02-Dec-2015	1	Initial release.

IMPORTANT NOTICE – PLEASE READ CAREFULLY