

Playrec

Multi-channel Matlab Audio

[Home](#) **[Getting Started](#)** [Usage](#) [Documentation](#) [Download](#) [Contact](#)

Getting Started

Before the playrec utility can be used it must first be compiled. This is not ideal but allows the utility to be used on as many platforms and with as many APIs as possible. Provided Matlab is installed, no other software needs to be purchased to achieve this on any platform, and when using Octave then no software needs to be purchased. The following steps try to cover all scenarios, but if you have any problems please [contact me](#). After compiling, an overview of how to use Playrec is given in [usage](#) with the [documentation](#) giving information on all available commands.

1. Download and unzip the latest version of the Playrec source and examples, available from the [download section](#).
2. Download and unzip the latest PortAudio v19 stable release files, available from the [download section](#) of the [PortAudio](#) website. This can be unzipped anywhere, although unzipping into the 'playrec/lib' directory (created in step 1) should simplify compilation later.
3. If ASIO support is required and can be used on the target platform then the ASIO SDK needs to be downloaded and unzipped, available from the [3rd Party Developer](#) section of the [Steinberg](#) website. This can be unzipped anywhere, although unzipping into the 'playrec/lib' directory (created in step 1) should simplify compilation later.
4. The steps now differ slightly for each platform: [Windows](#), [Mac](#), [Linux](#).

Windows

The following steps are specific to compiling Playrec on Windows in either Matlab or Octave and

continue on from the general steps above:

5. To compile Playrec an external compiler with C++ support is required. This is easiest by installing the latest [Microsoft Windows SDK](#) listed on the Matlab [Supported and Compatible Compilers](#) pages. It is not necessary to install the entire SDK, but the following options must be selected during the SDK installation process: 'Windows Headers and Libraries', 'Visual C++ Compilers' and '.NET Development'.

Matlab will need to be configured to use the external compiler, using the command `mex -setup`. If this does not correctly detect the compiler then it can be configured manually by modifying the configuration file `mexopts.bat`. The base version of this file is in the preference directory (use `prefdir` in Matlab to find where this is). Alternatively this base version can be overridden by placing a file with the same name in the current directory. An example of this file that seems to work with a number of Microsoft Visual C++ compilers and versions of Matlab can be downloaded from the [download section](#).

Octave may need to be configured to use the external compiler, depending on how the compiler was initially installed. The script `configure_octave_mex` may be useful in doing this by supplying as arguments the path to the compiler install directory (eg 'C:\Program Files\Microsoft Visual Studio 8\VC') followed by the path to the Platform SDK (see later).

6. If DirectSound support is required then the DirectX SDK is required. This used to be a separate download from the [Microsoft MSDN](#) website, but is now included in the Windows SDK described above. When specifying the Direct Sound directory as part of the Playrec compilation process (see below), you will need to specify the installation path for the SDK (eg 'C:\Program Files\Microsoft SDKs\Windows\v7.1').
7. If WASAPI support is required then this will also require the Microsoft Windows SDK installed, as described above.
8. Open Matlab or Octave and set the current directory to that of the playrec directory created in step 1.
9. Run `compile_playrec` and select the required APIs to use. In Octave the default value is given in square brackets. Where possible the required paths will have been automatically completed

with paths to the relevant files (eg the PortAudio path to the path where the PortAudio files were unzipped). Where this has not happened the paths need to be completed manually. Depending on the configuration in mexopts.bat there may not be a need to specify a Platform SDK path so initially try without. If it is required then it will need to be something like 'C:\Program Files\Microsoft SDKs\Windows\v6.0' or 'C:\Program Files\Microsoft Visual Studio 8\VC\PlatformSDK'.

10. In Matlab click Compile to start compilation, in Octave compilation will start after the final prompt has been answered. If verbose was selected then the output from the compiler will be displayed in the Command Window.

If all is successful then there should be a file called playrec.dll, playrec.mexw32, playrec.mexw64 or playrec.mex in the current directory - this is the MEX file for the utility. As with '.m' files, this must be either in the current directory or a directory in the path list to be able to use it. To check the file has been compiled correctly type `playrec('about')` to view information on the utility, or run `test_playrec` (available as part of the [playrec scripts](#) download) to try outputting sound. If either works then everything should have built correctly. Please [contact me](#) if you are having any problems.

Mac

The following steps are specific to compiling Playrec on Macs and continue on from the general steps above. Playrec has not been tested using Octave on Macs, so please [let me know](#) if you have tried this.

5. Open Matlab and set the current directory to that of the playrec directory created in step 1.
6. To compile Playrec an external compiler with C++ support is required. Using the command `mex -setup` select the gcc compiler, installed as part of Apple's developer tools.
7. Run `compile_playrec` and select the required APIs to use. Where possible the text boxes will have been automatically completed with paths to the relevant files (eg the PortAudio text box to the path where the PortAudio files were unzipped). Where this has not happened the paths need to be completed manually.
8. Choose the required options using the checkboxes and then click compile. If verbose was

selected then the output from the compiler will be displayed in the Matlab Command Window.

If all is successful then there should be a file called `playrec.mexmac` or `playrec.mexmaci` in the current directory - this is the MEX file for the utility. As with '.m' files, this must be either in the current directory or a directory in the path list to be able to use it. To check the file has been compiled correctly type `playrec('about')` to view information on the utility, or run `test_playrec` (available as part of the [playrec scripts](#) download) to try outputting sound. If either works then everything should have built correctly. Please [contact me](#) if you are having any problems.

Linux

At the moment Playrec has only had limited testing on Linux, although it has successfully been used in both Matlab and Octave by following similar steps to those for [Macs](#). Using Octave on Ubuntu:

5. To be able to compile Playrec in Octave on Ubuntu will require the `liboctave-dev` package to be installed. This is not installed by default when installing the octave package.
6. Each of the APIs that you want to use requires other packages to also be installed. For example, to use ALSA then the package `libasound2-dev` will also need to be installed.
7. Open Octave and set the current directory to that of the playrec directory created in step 1.
8. Run `compile_playrec` and select the required APIs to use. Selecting just the ALSA API, having already installed the required package as described above, should provide suitable access to all sound cards.

If all is successful then there should be a file called `playrec.mex` in the current directory - this is the MEX file for the utility. As with '.m' files, this must be either in the current directory or a directory in the path list to be able to use it. To check the file has been compiled correctly type `playrec('about')` to view information on the utility, or run `test_playrec` (available as part of the [playrec scripts](#) download) to try outputting sound. If either works then everything should have built correctly. Please [contact me](#) if you are having any problems.

Software License

© 2008-2015 Robert Humphrey