Github

https://github.com/Pixelphonics/MediaPlayer

Audio

Libraries

portAudio
http://www.portaudio.com/

sndfile
http://www.mega-nerd.com/libsndfile/

Video

Libraries

FFmpeg
License: GNU Lesser General Public License (LGPL) version 2.1
https://www.ffmpeg.org/legal.html

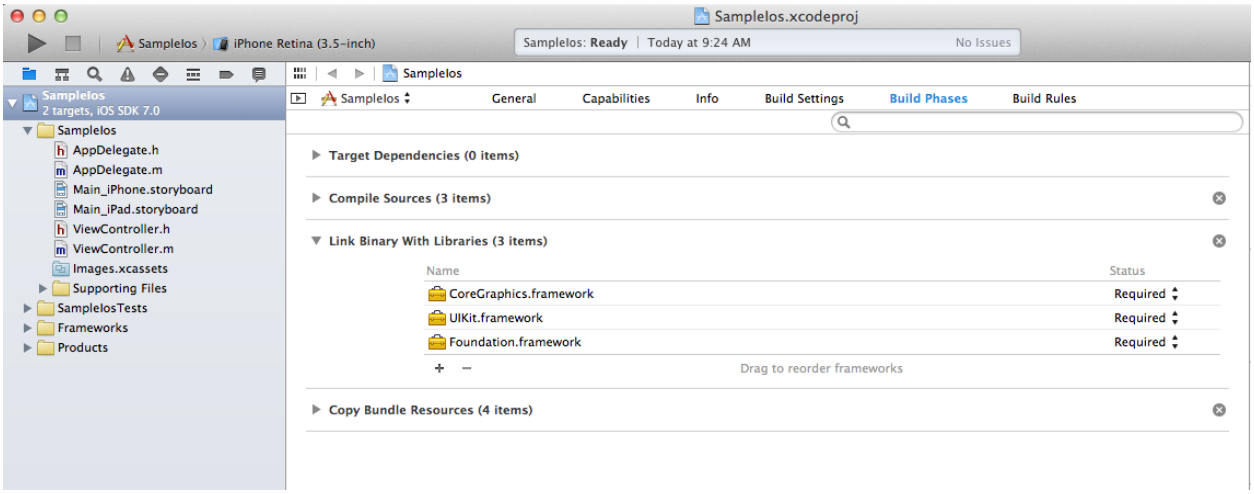
SDL
License: zlib license
https://www.libsdl.org/license.php

Mac:
This application was developed on the XCode 10.2.1

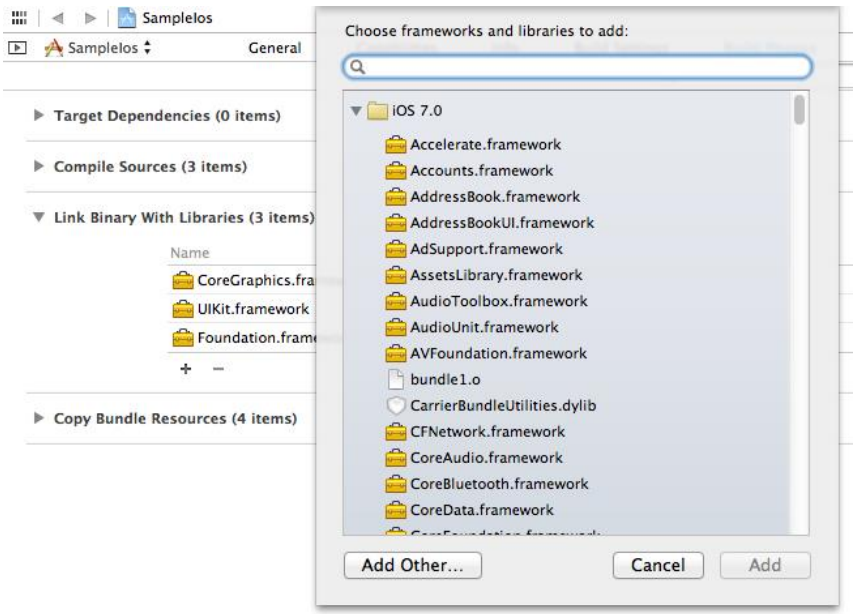
Static libraries have .a extension while dynamic libraries have .dylib extension.

Linking libraries or Framework

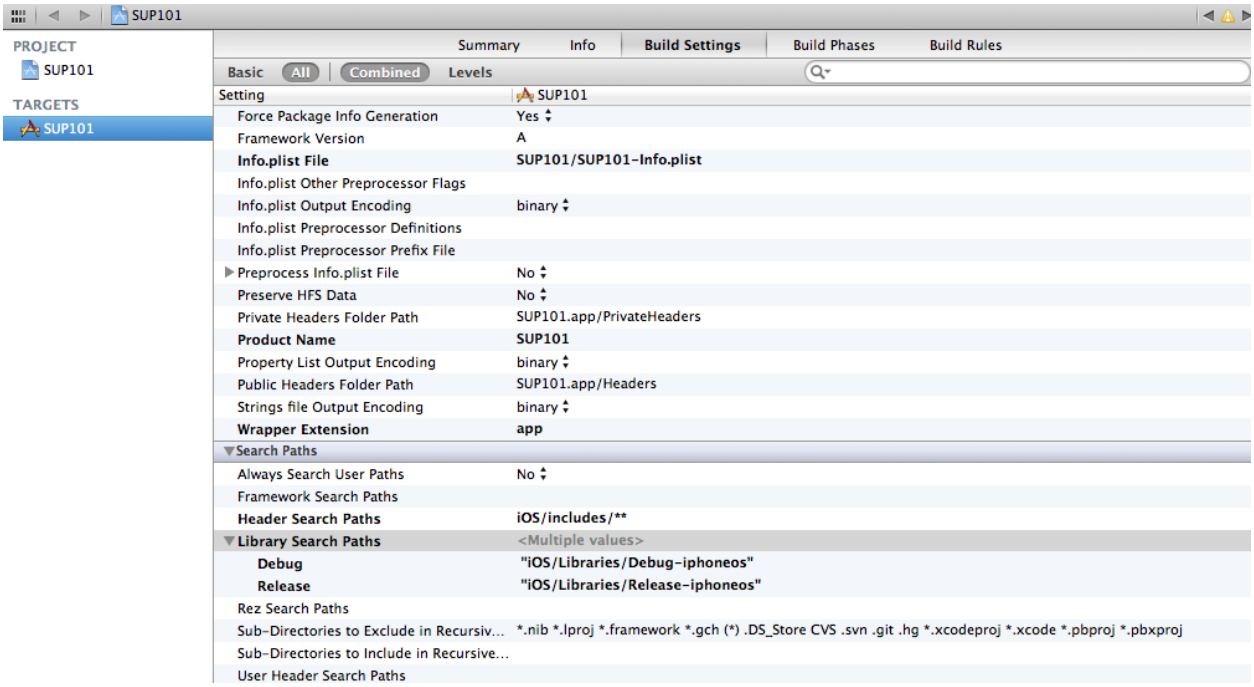
1. Go to your Project's Build Phases



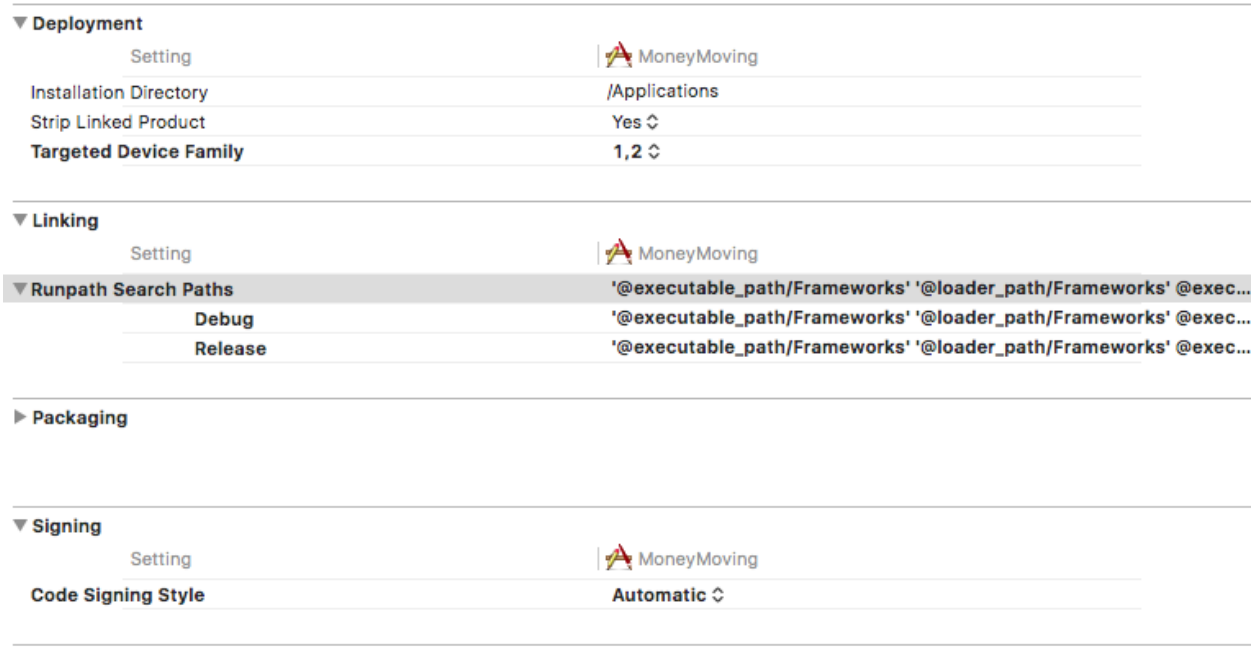
2. Click on the "+" under "Link Binary With Libraries" to add a new library. Then click on the "Add Other" button.
3. Navigate to the static library file (.a) or dynamic library (.dylib) or framework folder (.framework) and add it.



4. Go to **Build Settings** tab, then scroll down to the **Search Paths** section. Check for **Header Search Paths** and **Library Search Paths** fields. Make sure the reference has a path to your /include header files and /lib for library files



5. Go to **Linking->Runpath Search Paths**, in here it specifies the **executable run-time** search path. This is very important because if you have .dylib file you want to search beside the executable you need to set this within build settings so it will find it. **@executable_path** means the directory executable is in at run time.



IMPORTANT:

If everything works fine in XCode but the standalone executable cannot find libraries, it may be the build settings were wrong or because **.dylib** libraries usually have embedded default install path of

usr/local/lib. To check the path where executable or libraries are looking for files use the **otool -L** <filename> command.

Example: (**After build**)

otool -L EmplaceAV

Gave me something like this in the list:

cmd LC_LOAD_DYLIB

cmdsize 64

name /usr/local/lib/libsndfile.1.dylib (offset 24)

time stamp 2 Wed Dec 31 16:00:02 1969

So I know that it's looking for **libsndfile.1.dylib** under **/usr/local/lib**

install_name_tool -change /usr/local/lib/libsndfile.1.dylib @rpath/libsndfile.1.dylib EmplaceAV

Sets it to **@rpath** which is **@executable_path** from our earlier build settings.

Windows:

This application was developed on visual studio 2017 using windows 64 bits version of FFmpeg version 4.1.3 and SDL 2 - 2.0.9.

Linking libraries

The entire visual studio solution was uploaded to github. (see above for link)

If relinking is necessary, go to the following links and download all the files and follow the steps below.

portAudio
http://www.portaudio.com/

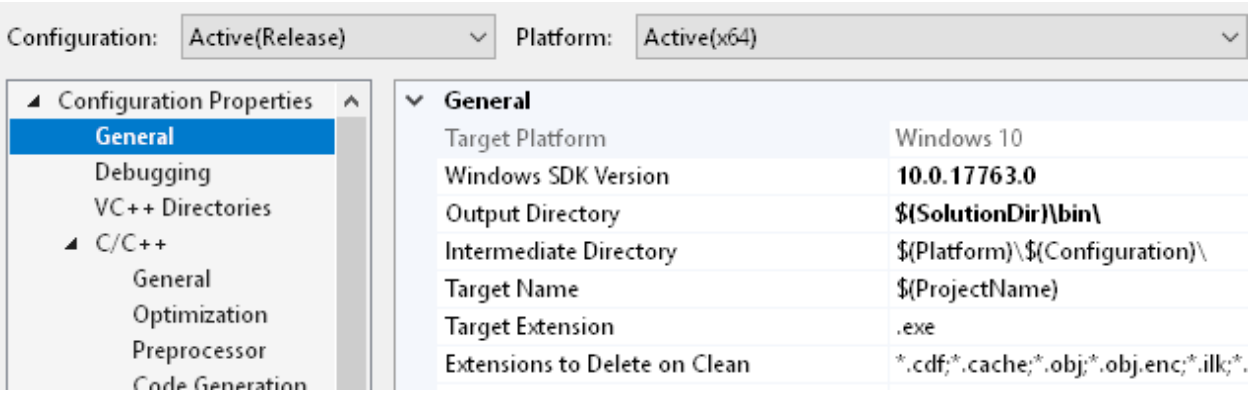
sndfile
http://www.mega-nerd.com/libsndfile/

FFmpeg

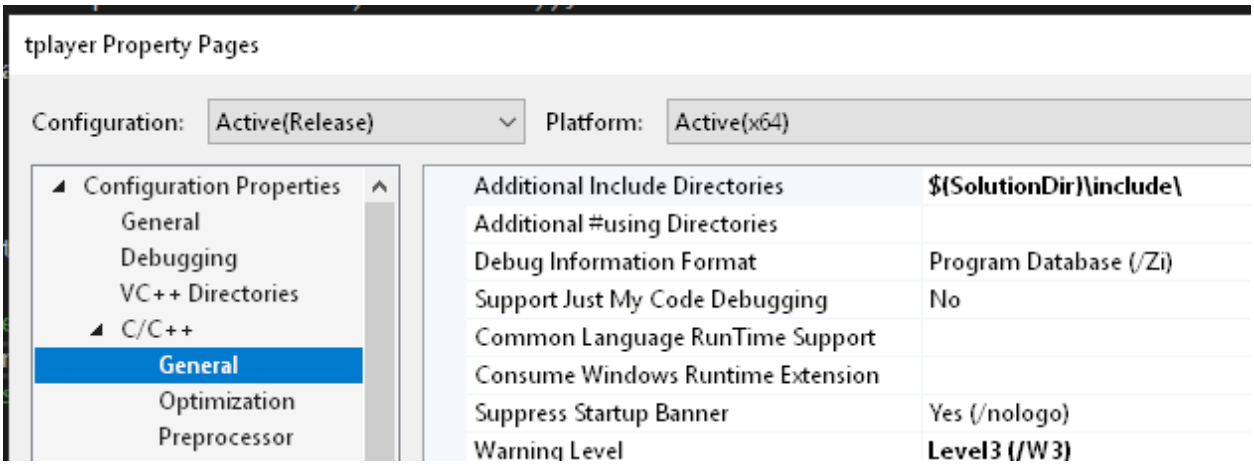
https://ffmpeg.zeranoe.com/builds/

SDL
https://www.libsdl.org/download-2.0.php

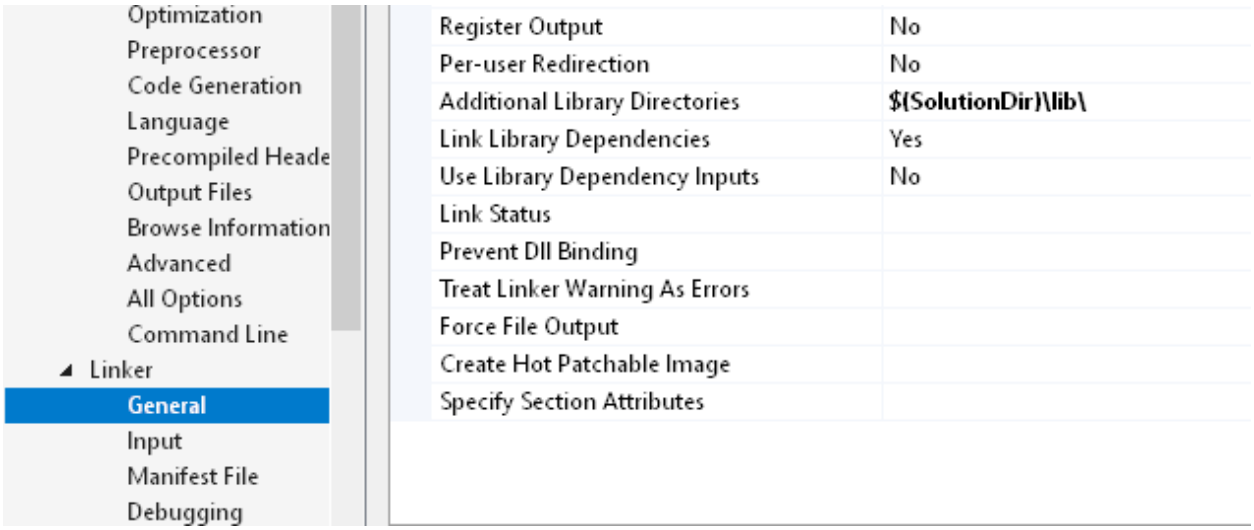
Step 1.
Specify where all the **dll** files are in **Output Directory.**

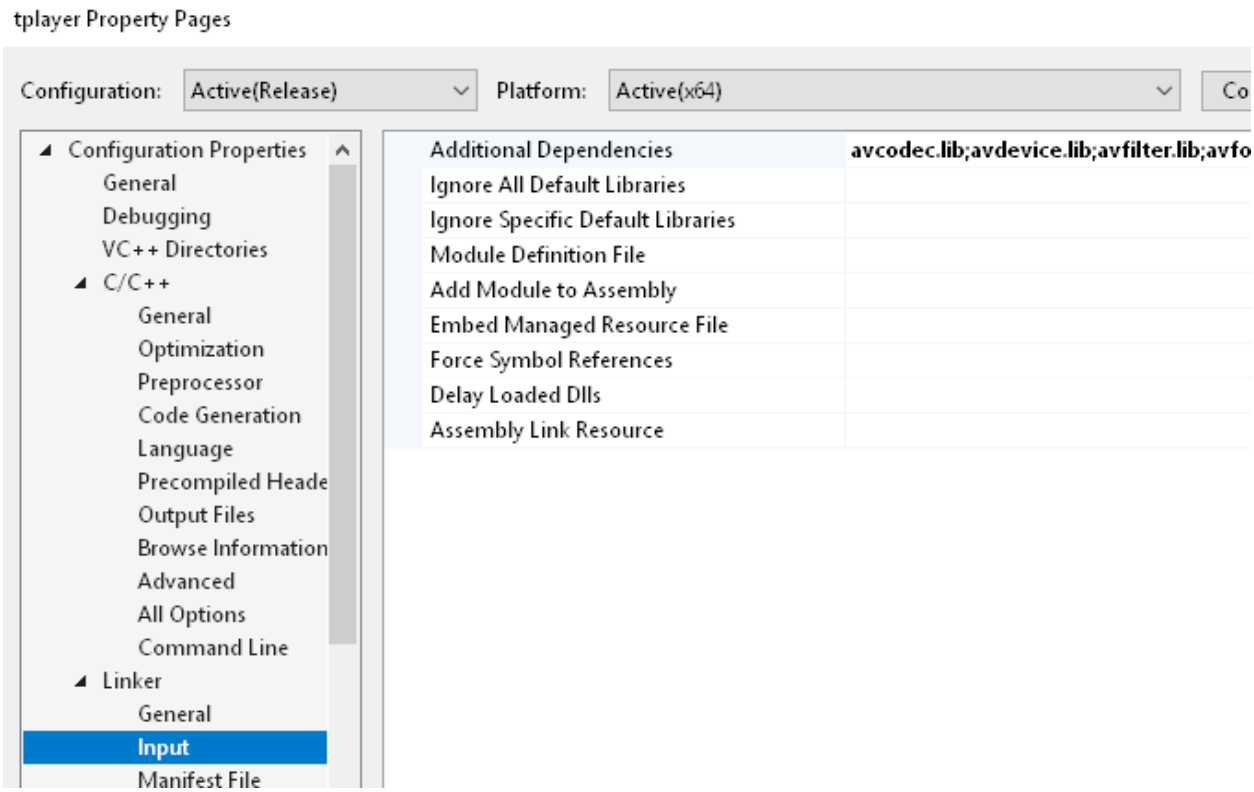


Step 2.
Specify where the **include** files are in **Additional Include DIrectories.**



Step 3.
Specify where all the **lib** files are in **Additional Library Directories** then add all of their names under **Additional Dependencies**.




tplayer Property Pages
Configuration:
Active(Release)
Platform:
Active(x64)
Configuration Properties
General
Debugging
VC++ Directories
C/C++
General
Optimization
Preprocessor
Code Generation
Language
Precompiled Heade
Output Files
Browse Information
Advanced
All Options
Command Line
Linker
General
Input
Manifest File
Additional Dependencies
avcodec.lib;avdevice.lib;avfilter.lib;avfo
Ignore All Default Libraries
Ignore Specific Default Libraries
Module Definition File
Add Module to Assembly
Embed Managed Resource File
Force Symbol References
Delay Loaded Dlls
Assembly Link Resource