<https://wiki.python.org/moin/GuiProgramming>

<https://docs.python.org/3/library/tkinter.html>

<https://docs.scipy.org/doc/scipy/reference/generated/scipy.io.wavfile.read.html>

<https://docs.python.org/3/library/wave.html>

The built-in wave library might be better, because it reads only as many frames as we specify at a time. But should be able to use the scipy library to ensure that we're interpreting the frames correctly (endianness, for instance).

Not quite sure how 2-channel is stored in LPCM, but my guess is a sample from left, sample from right, second sample from left, second from right, etc.

Wikipedia: "LPCM encodes a single sound channel. Support for multichannel audio depends on file format and relies on synchronization of multiple LPCM streams."

Ref: https://www.loc.gov/preservation/digital/formats/fdd/fdd000011.shtml

"Multiple channels Not applicable; LPCM encodes a single channel of sound. Stereo and two-channel audio is supported by interleaving two LPCM streams; see [AES3](https://www.loc.gov/preservation/digital/formats/fdd/fdd000142.shtml)."

AES3 <https://www.loc.gov/preservation/digital/formats/fdd/fdd000142.shtml>

"Audio Engineering Society interface format for serial digital transmission of stereo or two-channel [LPCM](https://www.loc.gov/preservation/digital/formats/fdd/fdd000011.shtml) (Linear Pulse Code Modulated) sound. The data is sent in audio blocks, each of which is made up of 192 frames numbered 0 to 191. Each frame is divided in 2 subframes (or channels): A (left) and B (right). Each subframe contains the information for one single sample of the PCM audio."