

Record players work by reading analog encoded waves generated by its needle dragging across the surface. The vibrations of the needle are amplified sometimes filtered and projected by a speaker. This just so happens to be the reverse of how originally records were transcribed. Original record transcribers used a microphone to collect vibrations and engrave the vibrations as a wave into the record. Thus the grooves are simply a physical scale model of the actual wave and thus the relations between the grooves and sound waves can be leveraged in applying the same formulas to both. 3D printing a record therefore will be the process of taking a digital audio recording and a blank record and transcribing the wave virtually by the removal of part of the record by the intersecting a 3d model of the wave as a spiral onto the record. First, before I get any further ahead of myself, a few more words about waves and digital audio encoding.

A sound wave is comprised of two main components, at least for our purposes. They are the amplitude and wave length. The amplitude is the height of the wave in relation to its center. The wavelength is the distance between the peaks of a wave. Since wavelength and frequency are related to each other in an inverse manner in relation to the phase velocity of the wave's medium we can ignore frequency during encoding since it will remain in proportion to the wavelength. The exact relation where λ is the wave length, f the frequency and v the phase velocity is given below.

$$\lambda = \frac{v}{f} \tag{1}$$

Most commercially available uncompressed audio formats use LPCM encoding. Linear pulse-code modulation, LPCM audio channels are stored

as vectors of amplitudes with a set sampling rate, times the wave is measured per second, and bit depth, max difference in amplitudes as a digital value. This makes the formulation of transferring the amplitude of a wave at a given time from a wav file, microsoft developed LPCM format, to a record trivial as the amplitudes stored in each channel can be normalized to the bitdepth and read as a scalable vector with values ranging from -1 to 1 inclusive. However, since modern audio encoding utilizes multi-channel sound it is necessary to combine the channels by averaging them element-wise as vector and thus creating a single wave consisting of an approximation of the combined sound of the original waves.

Translating the wavelength is a little less trivial.

$$ou(r, A, \beta, \theta) = (r + (r + A * \beta) * \cos(\theta), r + (r + A * \beta) * \sin(\theta)) \quad (2)$$

$$ol(r, A, \theta) = (r + r * \cos(\theta), r + r * \sin(\theta)) \quad (3)$$

$$il(r, A, \theta) = (r + (r - gW) * \cos(\theta), r + (r - gW) * \sin(\theta)) \quad (4)$$

$$iu(r, A, \theta) = (r + (r - gW - A * \beta) * \cos(\theta), r + (r - gW - A * \beta) * \sin(\theta)) \quad (5)$$