cedargrove_chime

A CircuitPython class for generating wind chime and bell sounds using synthio..

Acknowledgement and thanks to:

- Lee Hite, '*Tubular Bell Chimes Design Handbook*' for the analysis of tubular chime physics and overtones.
- C. McKenzie, T. Schweisinger, and J. Wagner, '*A Mechanical Engineering Laboratory Experiment to Investigate the Frequency Analysis of Bells and Chimes with Assessment*' for the analysis of bell overtones;
- Liz Clark, '*Circle of Fifths Euclidean Synth with synthio and CircuitPython'* for the waveform and noise methods;
- Todd Kurt for fundamentally essential synth hints, tricks, and examples (https://github.com/todbot/circuitpython-synthio-tricks).

Also, special thanks to Jeff Epler for the comprehensive design and implementation of the amazing CircuitPython *synthio* module.

- Author(s): JG for Cedar Grove Maker Studios

Implementation Notes

Hardware:

Software and Dependencies:

- Adafruit CircuitPython firmware for the supported boards: https://circuitpython.org/downloads

class cedargrove_chime.Scale

A collection of wind chime musical note scales.

SubClasses:

- **Scale.Westminister**
- **Scale.Pentatonic**
- **Scale.CNine** (default)
- **Scale.HavaNegila**
- **Scale.CorinthianBellsA**
- **Scale.CorinthianBellsB**
- **Scale.CorinthianBellsC**
- **Scale.CorinthianBellsEb**
- **Scale.CorinthianBells**
- **Scale.Whittington**
- **Scale.Canterbury**
- **Scale.Trinity**
- **Scale.Winchester**
- **Scale.StMichaels**
- **Scale.HappyBirthday**

class cedargrove_chime.Material

The attack time, attack level, and release time for various chime/bell materials.

SubClasses:

- **Material.SteelEMT** (default)
- **Material.Ceramic**
- **Material.Wood**
- **Material.Copper**
- **Material.Aluminum**
- **Material.Brass**

class cedargrove_chime.Striker

The attack time, attack level, and release time for various chime/bell materials.

SubClasses:

- **Material.Metal** (default)
- **Material.Plexiglas**
- **Material.SoftWood**
- **Material.HardWood**

class cedargrove_chime.Voice

The pre-defined synth voices.

SubClasses:

- **Voice.Bell** – a single-capped tube with empirical overtones.
- **Voice.Perfect** – a dual-capped tube with algorithmically generated overtones equal to the length-related harmonics.
- **Voice.Tubular** — a traditional open-ended tube chime with empirical non-harmonic overtones. (default)

class cedargrove_chime.Chime(**, audio_out, scale=Scale.CNine, material=Material.SteelEMT, striker=Striker.Metal, voice=Voice.Tubular, scale_offset=0, loudness=0.5, debug=False*)

A CircuitPython class for generating wind chime and bell sounds using *synthio*.

Parameters:

- **audio** – An instantiated audio object to receive the output audio stream, typically an I2S connection, analog DAC output pin, or PWM output pin. No default.
- **scale** – The list of playable chime notes in Scientific Pitch Notation (SPN). Each element of the list is a single SPN string, such as "A#4" for the A# for Bb note in the fourth octave. The *Chime.Scale* class contains a number of chime scale lists. Defaults to *Scale.CNine*.
- **material** – A list of chime material note envelope parameters for *attack time*, *attack level*, and *release time*. The *Chime.Material* class consists of presets for a variety of materials. Defaults to *Material.SteelEMT*.
- **striker** – A list of striker material note envelope parameter ratios for *attack time and attack level*. The ratios are used to adjust chime material note envelope properties for a particular striker material. The *Chime.Striker* class consists of presets for a variety of materials. Defaults to *Striker.Metal*.
- **voice** – A string representing the pre-defined synth voices. The *Chime.Voice* class contains presets for *Voice.Bell* ("*bell*", a single-capped tube with empirical overtones), *Voice.Perfect* ("*perfect*", a dual-capped tube with algorithmically generated overtones equal to the length-related harmonics), and *Voice.Tubular* ("*tubular*", a traditional open-ended tube chime with empirical non-harmonic overtones). Defaults to *Voice.Tubular*.
- **scale_offset** – A positive or negative integer value of note pitch half-steps to offset the pitch of the scale. Defaults to 0 (no scale pitch offset).
- **loudness** – A normalized floating point value for output amplitude, ranging from 0.0 to 1.0. Defaults to 0.5 (one-half volume).
- **debug** – A boolean value to enable debug print messages. Defaults to False (no debug print messages).

scale

The chime scale list in SPN.

loudness

The current loudness value.

strike(**, root_note=69, amplitute=0*)

Strike the chime or bell. The midi root_note integer ranges from 0 to 128. The note_amplitude is a floating point value between 0.0 and 1.0. The note envelope and overtone values are determined by the chime/bell and striker materials.