

cedargrove_ad9833

A CircuitPython driver for the AD9833 Programmable Waveform Generator.

The AD9833 is a programmable waveform generator that produces sine, square, and triangular waveform output from 0 MHz to 12.5MHz with 28-bit resolution. The driver controls the waveform generator's frequency, phase, and waveform type.

The Cedar Grove AD9833 Precision Waveform Generator and AD9833 ADSR Precision Waveform Generator FeatherWings provide all the support circuitry for the AD9833. The ADSR (Attack, Decay, Sustain, Release) version incorporates the AD5245 digital potentiometer to provide output amplitude control.

- Author(s): JG for Cedar Grove Studios

Implementation Notes

Hardware:

Software and Dependencies:

- Adafruit CircuitPython firmware for the supported boards: <https://github.com/adafruit/circuitpython/releases>

```
class cedargrove_ad9833.AD9833(*, wave_freq=440, wave_phase=0, wave_type="sine", spi=board.SPI(),
select=board.D6, m_clock=25000000)
```

Class representing the AD9833 Precision Waveform Generator.

Parameters:	<ul style="list-style-type: none">• wave_freq – The 28-bit waveform frequency in Hz ranging from 0.09 to 12.5MHz (one-half the master clock frequency). Defaults to 440. Resolution is 0.09Hz.• wave_phase – The waveform integer phase offset = 2π Rad // 4096. Defaults to 0.• wave_type – The wave shape string parameter "sine", "triangle", or "square". Defaults to "sine".• spi – The <i>busio.SPI</i> definition. Defaults to <i>board.SPI()</i>.• select – The chip select pin designation. Defaults to <i>board.D6</i>.• m_clock – Master clock frequency in Hz. Defaults to 25MHz.
--------------------	---

wave_freq

A class set/get property.

Parameters:	Change or read the frequency value in Hz.
--------------------	---

wave_phase

A class set/get property.

Parameters:	Change or read the integer phase offset value.
--------------------	--

wave_type

A class set/get property.

Parameters: Change or read the waveform type.

pause()

A class helper function.

Parameters: Pause the wave generator and freeze the output at the latest voltage level by stopping the internal clock.

start()

A class helper function.

Parameters: Start the wave generator with current register contents, register selection, and wave type setting.

stop()

A class helper function.

Parameters: Stop the wave generator and reset the output to the midpoint voltage level.

reset()

A class helper function.

Parameters: Stop and reset the waveform generator. Pause the master clock. Update all registers with default values. Set sine wave mode. Clear the reset mode but keep the master clock paused.