# Cedar Grove AD9833 Precision Waveform Generator FeatherWing

The AD9833 Precision Waveform Generator FeatherWing is an Adafruit Feather-compatible module. The Waveform Generator produces an op-amp buffered sine, triangle, or square wave output with a practical frequency range of approximately 0 to 300KHz with 0.1Hz resolution.

* Frequency accuracy and stability is determined by the on-board 25MHz +/- 100ppm crystal oscillator (Abracon ASV-25.000MHZ-E-T).
* Waveform Generator control is via the Feather SPI bus; chip select is provided by the D6 GPIO pin.
* Output signal peak-to-peak amplitude is adjustable with a maximum unipolar output of slightly less than 3.3 volts.
* Output can optionally be connected to the A2 analog input pin via an on-board jumper for real-time output signal monitoring.
* The wing derives its power from the host Feather's 3.3V power.

The AD9833 Precision Waveform Generator FeatherWing was tested with a Feather M4 Express using CircuitPython version 4.1.0 rc-1. Example sweep generator code is provided in the repository (sweep example video: <https://youtu.be/O1vMfLoCWzg>).

OSH Park project: https://oshpark.com/shared\_projects/al6aPN0u

Refer to the Analog Devices' AD9833 Waveform Generator data sheet for device control protocol details.

#### Primary Project Objectives:

1. Generate useful waveforms
   1. Highly accurate and stable absolute frequency: ±0.5Hz, 1000ppm
   2. Moderately accurate waveform shape: 8-bit minimum
   3. Fixed 3.0 volt peak-to-peak output, centered at 1.5 volts
2. Implement with CircuitPython code
3. Increase SPI communications knowledge
4. Improve SMD soldering skills
5. Improve conceptual knowledge
   1. SPI communications
   2. Swept frequency control
   3. Digital signal generation
   4. CircuitPython device drivers
   5. ADSR envelope modulation (optional)
6. Fabricate physical equipment
   1. Workbench fixed and swept signal generator
   2. MIDI and CV controlled LFO and audio frequency Eurorack oscillator module

#### Deliverables:

1. Custom FeatherWing or smaller PCB
2. CircuitPython device driver
3. CircuitPython examples and test code
4. KiCAD schematic and board layout with Gerber code
5. OSH Park public shared project page
6. GitHub public repository page
7. Project description and report

#### Concepts Learned: