Microcontroller-based Mechanical Chaotic Oscillator

Eric Ayars and Brandon Thacker
California State University, Chico
eayars@csuchico.edu

We have built a microcontroller-based mechanical chaotic oscillator suitable for Advanced Lab use that allows complete control of all system parameters including drive frequency, drive amplitude, static field, rotational inertia, and damping. The onboard microcontroller synthesizes the drive signal, tracks position and time, and reports at synchronized intervals suitable for generation of multiple Poincare plots over an entire drive cycle. Control and communication is managed via USB through SCPI-compatible commands, making the instrument easily usable with LabVIEW or any other serial-capable language. The instrument can be constructed inexpensively with tools and construction techniques readily available to advanced undergraduates in physics.

Flywheel provides rotational inertia and conductivity for damping.

Magnetic Damping, adjustable with micrometer.

Optical quadrature detector for angular position tracking.

Field Coils provide an optional equilibrium point for the rotating dipole.

Control, drive signal, and USB communication is provided by a Cortex M4 processor on a Teensy 3.1 development board.

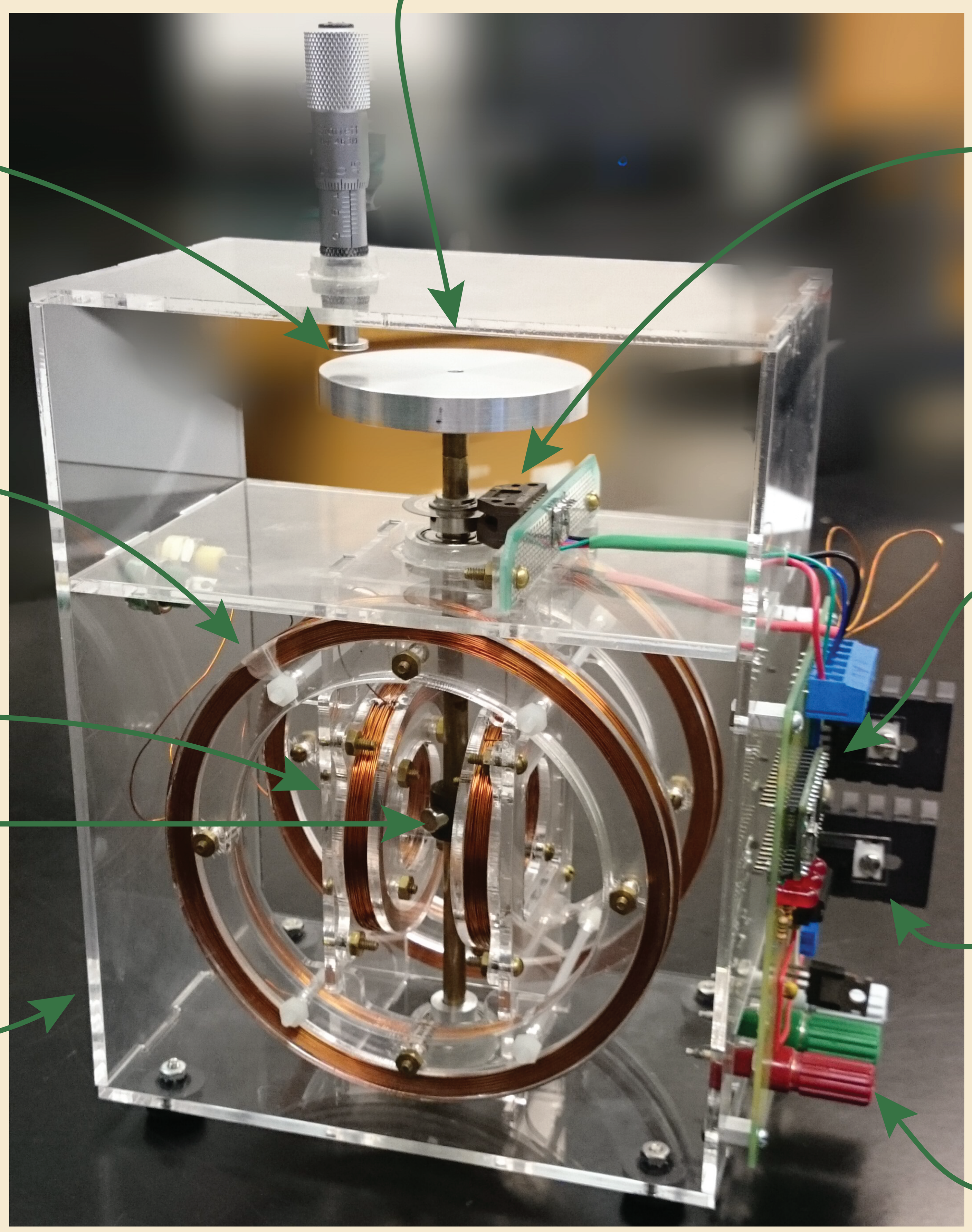
Oscillating drive coil current is synthesized by the microcontroller.

Level-shifting and current-amplification circuit changes the microcontroller's 0-3.3V analog drive signal to ±10V at up to 5A.

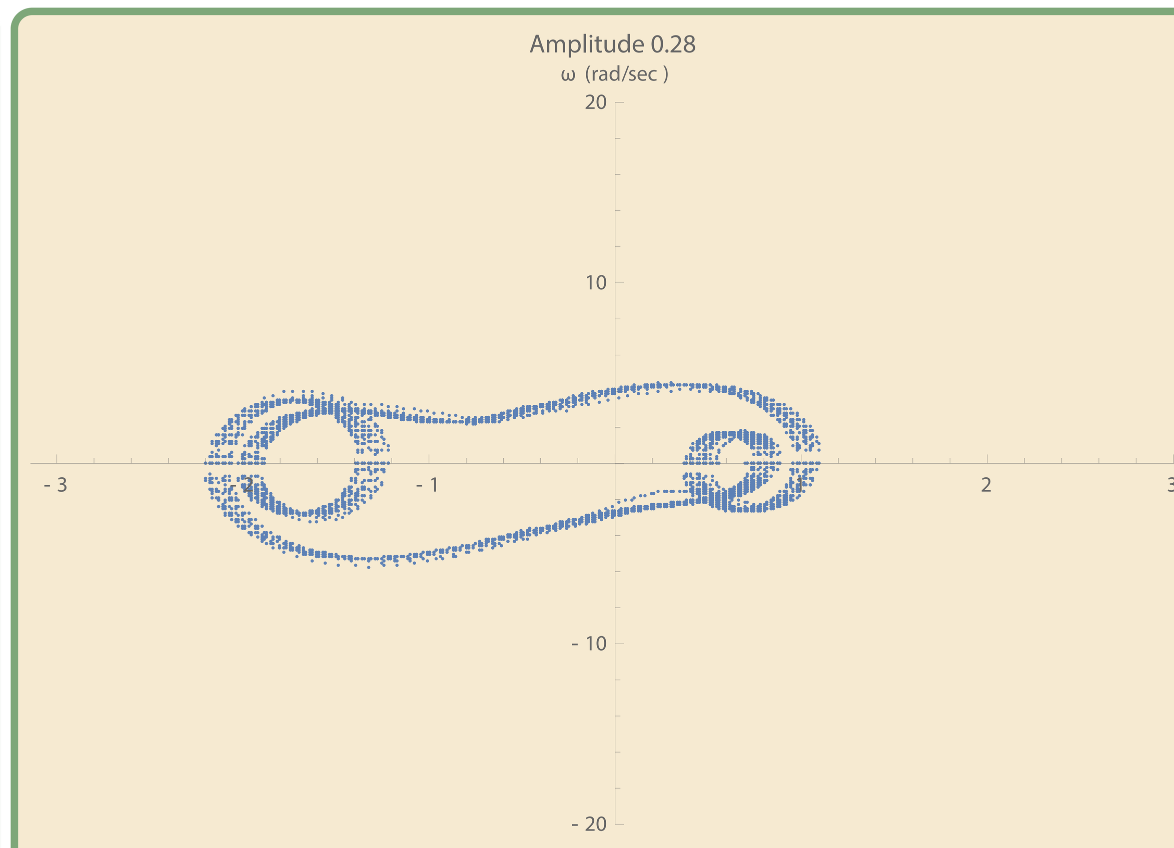
Magnetic dipole

Frame and coils are cut from 3mm acrylic with a laser cutter.

External DC supplies provide ±12V to run the amplifier, and (optionally) the Field Coil current.

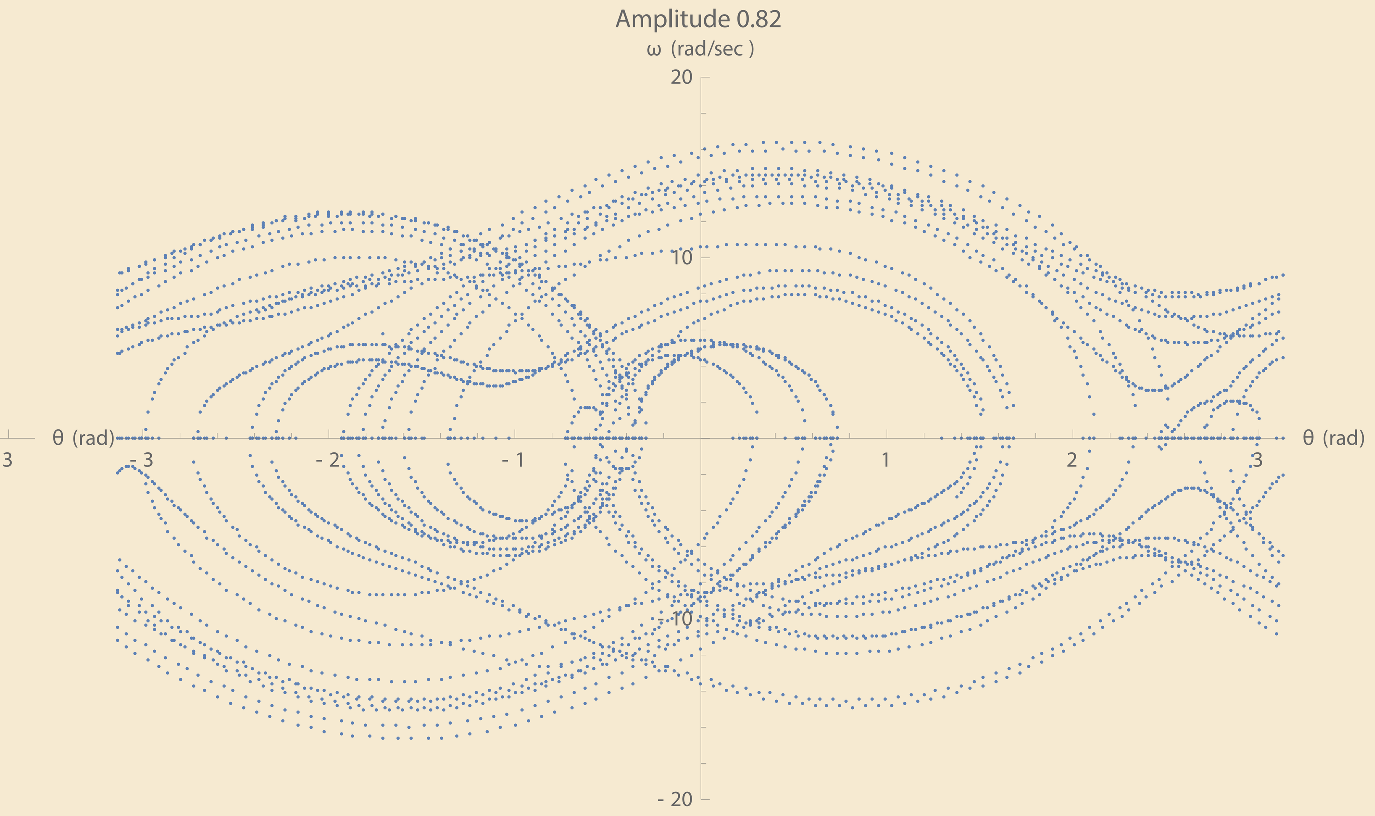


This is not a black-box experiment.

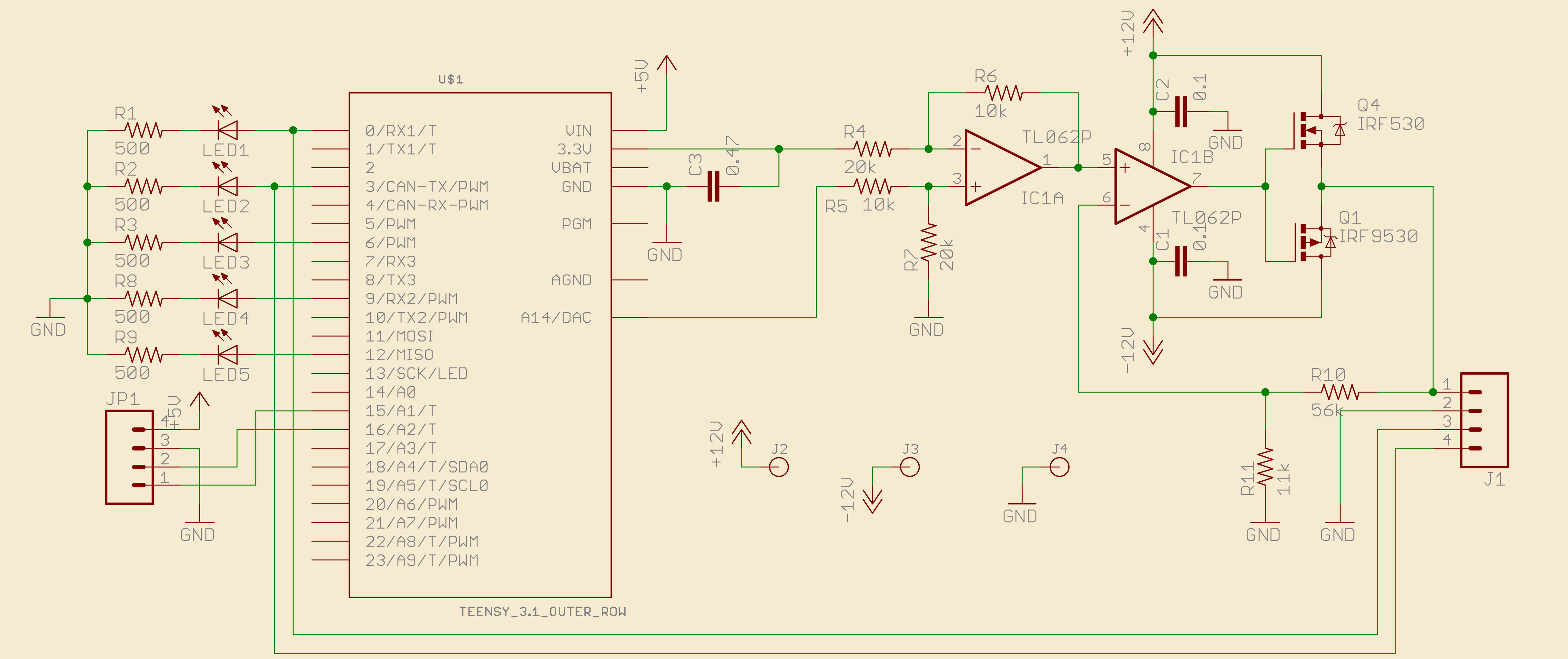


Above one can see the oscillator in a steady-state condition. The width of the trace is due to period-doubling: this is a period-4 (or possibly period-8) motion.

The width may also be partially due to drift in the center of the offset circuit, as this data was collected with an older version of the electronics which suffered in that regard. The newer version of the control circuitry, shown below, has not been observed to have this problem.



Adjusting the amplitude changes the rotor to a chaotic motion.
The center of both these graphs is offset slightly towards negative angles, as you may observe. The easy upgradability of the controller allowed us to add a "ZERO" command to fix this for future students.



The brains of the apparatus are provided by a Teensy 3.1 (ARM Cortex-M4, 96MHz) This board has vastly-improved capabilities compared to the more common Arduino, and includes a 12-bit D/A converter. The D/A goes from 0-3.3V, so we used a difference amplifier and class B current-amplifier stage to center the output, bring the voltage up to ±10V, and provide a useful drive coil current.

The Teensy does all the heavy lifting for this experiment: the external computer is used only for user interfacing and data plotting. The Teensy:

- Generates the sinusoidal drive signal (frequency 0-10Hz, amplitude 0-10V)
- Tracks the position and velocity of the rotating dipole
- Listens to the USB port for SCPI commands from the user. These commands can set frequency and amplitude, probe position and velocity of the rotor, toggle data reporting, define zero position, etc.
- Reports position and velocity of the rotor 256 times per drive cycle, for phase and Poincaré plots.
- Flashes various indicator LEDs to keep students from becoming bored

We tried to make the controller board as general-purpose as possible, and it may be useful for other experiments as well. Basically it's a USB-interfaced computer-controlled low-frequency programmable function generator with external sensor capability.

Want to build one of these for your Advanced Lab or Chaos class? We'd be happy to help. Complete plans are available, including circuit layout (we even have some extra boards) CAD drawings of the coils and frame, and the controller program. Contact Eric Ayars for more information.