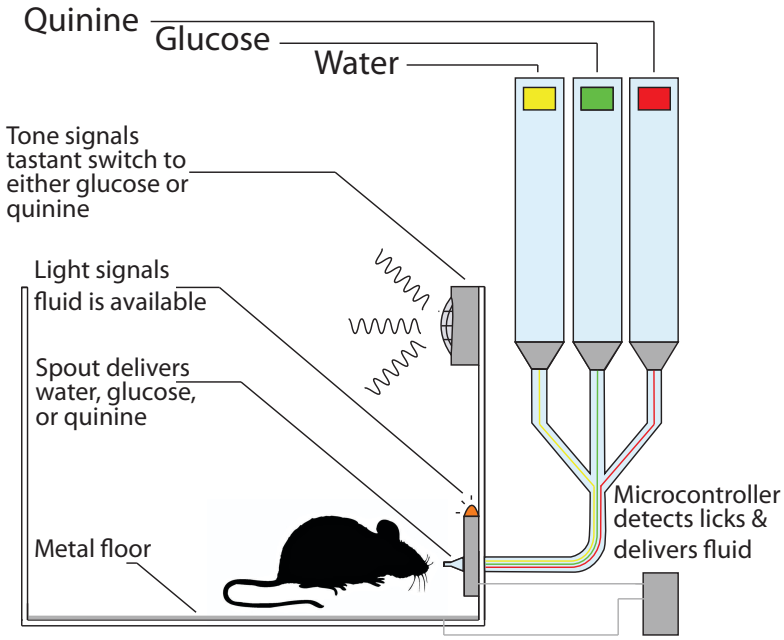
MultiTastant Spout
Construction Instructions
Nathan V-C, Ph.D.
3/9/2016

Overview:

This directory contains open source design materials for a rodent behavioral environment for learning from positive and negative events. These materials are for an Arduino controlled lick spout that controls delivery of small amounts of three different tastants, playing of 2 different audio cue tones (11KHz and 4KHz in this design), and timing of an in-cage light cue.



Overall the system contains three components:

1. **Spout Control Board (Arduino Uno Shield)**
   Eagle and Gerber design files contained in:
   3tastsp_v7_resistorcooldrive_battcheck/
   This is an Arduino shield contains hardware to control solenoid valves that deliver tastants, LED lights that inform the experimentalist about task events, a calibration button, location for analog-input signal that cues trial type, location for the raw analog input lick signal (through BNC connectors), and an HDMI plug that transmits 8 digital output data signals to your preferred data acquisition system. This control board is placed outside of the behavioral environment (typically a closed sound attenuated cupboard or box). This circuit board design can be sent out to a fabrication company. I use Seeedstudio and these gerber files match their production requirements.
2. **Spout Interface Board**
   Eagle and Gerber design files contained in: **sptinterface_v4_battcheck/**

This printed circuit board delivers cues and power to the solenoid valves that control tastant delivery, the internal cage cue light, some circuitry to protect the lick signal voltage from noise and current draw elsewhere in the system, and produces two cue tones using 555 timer circuits. This board needs to be placed near the spout assembly (typically inside the behavioral enclosure, near but not inside the cage). This design has a feature that if you wish to switch to IR lick detection, there is a plug and circuitry available for that. With this design you must choose between a raw voltage signal lick detection (where the mouse is on a grounded metal cage floor) and IR lick detection. The current spout design does not support IR detection but this is do-able, and this board is set up both to run 5V power to an IR emitter, and to detect analog output from an IR detector. Also, the lick detection circuit is consolidated to the left of the board, with separation possible from the right part of the board that generates tones and controls valves. I designed this board this way so the same system can be adapted to a system where valves need to be separate from the spout (for example, when combined with electrophysiological recording). This PCB can be sent out to a fabrication company. I use Seeedstudio and these gerber files match their production requirements.

3. **Spout Assembly**
   OpenSCAD and .stl design files contained in: **spoutassembly/**
   The spout itself is hand built from thin steel 24 gauge tubing arranged in a cylinder and that then interfaces with barbed connectors to link to solenoid valves that deliver the tastant. This spout assembly also holds the solenoid valves. 3 wires from the spout connect to the Spout Interface Board, the LED power, Ground, and a wire connected to the metal spout. I've included an OpenSCAD and .stl file of the parts needed for the spout assembly here

These three components are included in these design files. In addition to these three components, you will need a source to generate an analog input signal that cues trial type (I use a NIDAQ card), and Arduino UNO (loaded Arduino script **PCB_3tast_v29_3tnblank_public_switchpins.ino** or your own modification), a cage and behavioral enclosure, a set of computer speakers to play the tones, and a 9V battery to power the tone production.

At this time, an initial nearly complete parts list is available in this directory (excel file). Construction instructions are still being compiled, but if you would like to build a rig like this and need more complete information before I have updated, please don't hesitate to contact me via github.