

Behavioral Box - Setup Manual

Kwan Lab, Yale University

<http://alexkwanlab.org/>

Last updated: April 7, 2020

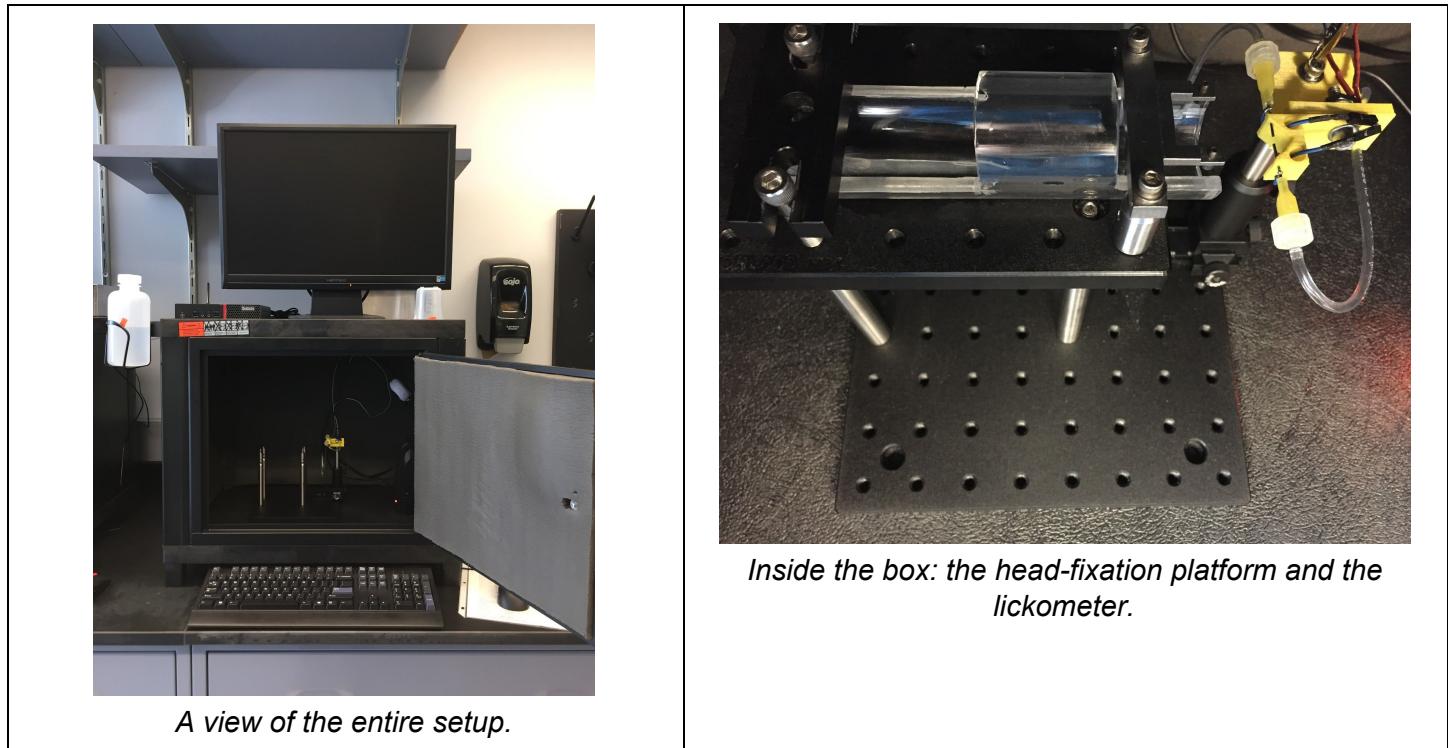
Overview	3
Section 1: Cabinet	5
Cabinet	5
Parts List	5
Assembly	5
Section 2: Head fixation	6
Head fixation platform	6
Parts List	6
Assembly	7
Notes	8
Section 3: Fluid delivery and lick detection	9
Lickometer	9
Parts List	9
Assembly	10
Notes	11
Lick detection circuit	11
Parts List	11
Assembly	12
Notes	13
Fluid delivery circuit	13
Parts List	13
Assembly	14
Notes	15
Fluid delivery circuit (obsolete)	15
Parts List	16
Assembly	16
Section 4: Steering wheel	18
Steering wheel	18
Parts List	18
Assembly	19
Notes	20

Section 5: Computer control	21
Control of the main behavioral rig	21
Parts List	21
Assembly	21
Notes	22
Control of the steering wheel	23
Assembly	23
Section 6: Pupilometry	24
Pupilometry	24
Parts List	24
Assembly	25

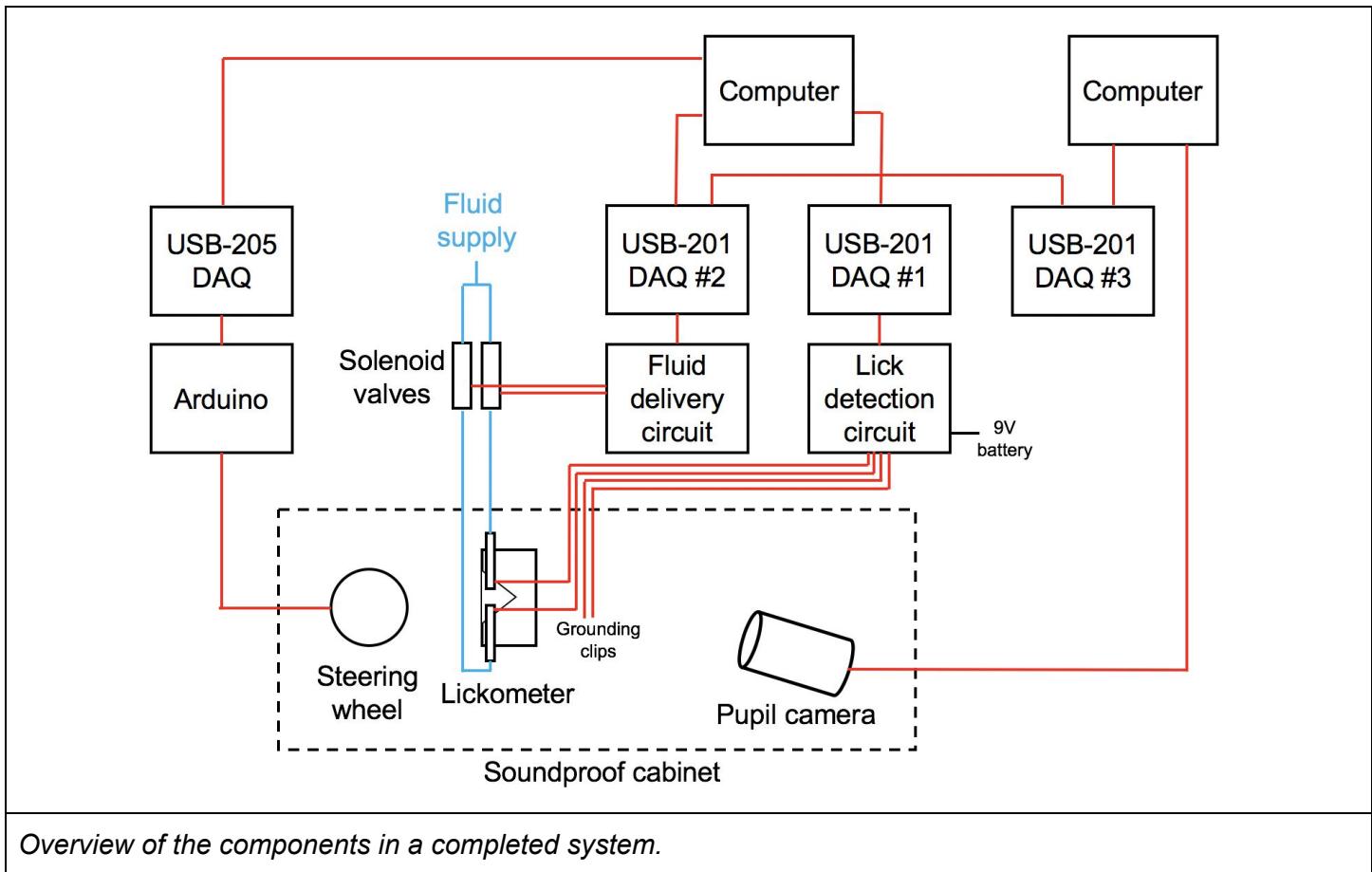
Overview

This document provides instructions for constructing a box to train a head-fixed mouse to perform two-choice decision-making tasks. The first version of the box was used for studies in the Kwan Lab at Yale, including [Siniscalchi MJ et al., Nat. Neurosci., 2016](#), [Siniscalchi MJ et al., Cereb. Cortex, 2019](#), and [Barthas F et al., Biol. Psychiatry, 2020](#). Newer versions incorporated additional features.

A completed box looks like this:



Each section describes a component of the system. The schematic below shows how the various components are connected in the final system.



Overview of the components in a completed system.

Contributors to this manual (in alphabetical order): Huriye Atilgan, Alex Kwan, Cayla Murphy, Heather Ortega, Neil Savalia, Michael Siniscalchi, Hongli Wang

Version history

Version 1 (2013): The initial setup for the two-choice head-fixed setup

Version 2 (2019): Major changes:

- Replaced parallel port with another DAQ
- New fluid delivery circuit
- New steering wheel response apparatus
- New pupillometry

Version 3 (2020): Organized the document and elaborated on the steps

Section 1: Cabinet

1. Cabinet

For behavioral training, the mouse sits inside a soundproof cabinet. For imaging and optogenetic experiments, the cabinet is typically not used. Instead, the head fixation platform and other components are integrated into the rigs. This section provides the details for building a soundproof cabinet.



The photo shows an immobile version of the cabinet without legs and wheels. The inside of the cabinet is lined with sound-absorbing foam, which is difficult to see except for the foam that is on the inner side of the door.

a. Parts List

- 1 of Audio-visual cart - Style E, enclosed polyethylene plastic, swivel casters (**McMaster-Carr, 4731T74**)
- 5 ft of Soundproof foam - Polyurethane foam with adhesive backing, 75% sound absorbed, 1" thick, 54" wide (**McMaster-Carr, 5692T49**)

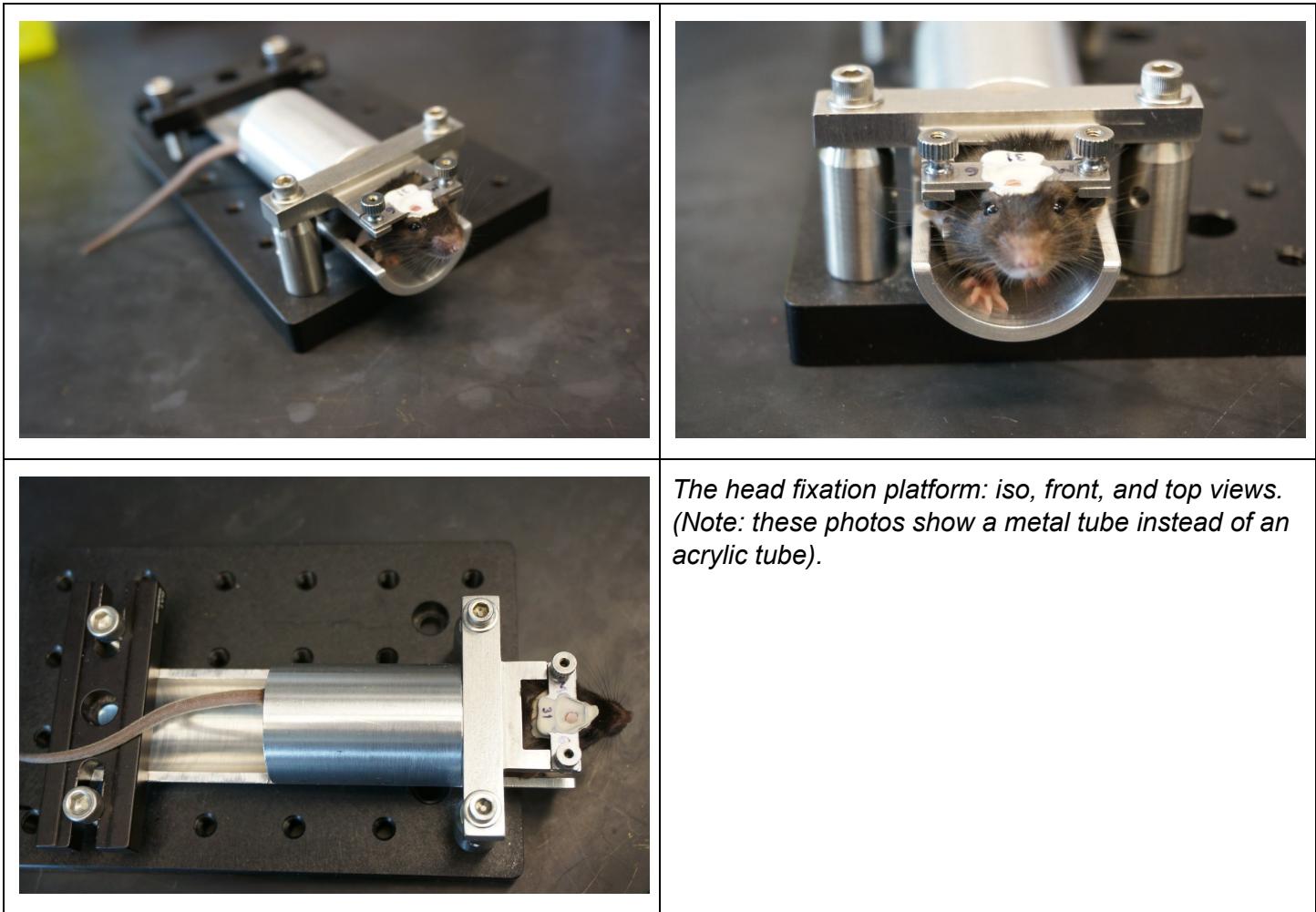
b. Assembly

- 1) Assemble the audio-visual cart.
 - For an immobile cabinet, assemble the enclosed portion of the audio-visual cart.
 - For a mobile cabinet, you can modify the intended configuration to put the enclosed portion on the upper level (instead of on the lower level as depicted by the manufacturer).
- 2) Dress the inside of the cabinet with soundproof foam. It is not advisable to put the foam on the outside of the cabinet, because there will be a lot of wear and tear on the foam.

Section 2: Head fixation

1. Head fixation platform

Each mouse is surgically implanted with a stainless steel head plate. The head plate is bolted onto a head plate holder bar that is mounted on two metal posts. The mouse sits inside an acrylic tube, restricting gross movements but allowing for some postural adjustments. All of these sit atop a small breadboard. Typically the mouse is head fixed on this small breadboard in a staging area, before the small breadboard is brought inside the cabinet and positioned on top of a larger breadboard.



a. Parts List

For the mouse:

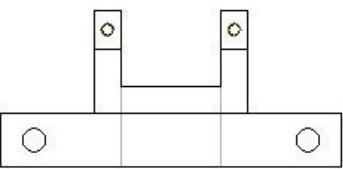
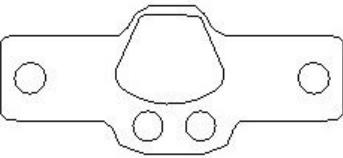
- - 1 of head plate holder bar, stainless steel 304, uncoated (eMachineShop.com, [/headplate/AK_holder.ems](#))
- - Head plates, stainless steel (eMachineShop.com, [/headplate/AK_PFCheadplate.ems](#))

For the small-breadboard:

- 1 of 6"x4" breadboard (**Thorlabs, MB4**)
- 2 of 1" post (**Thorlabs, TR1**)
- 2 of M2.5 screw (**McMaster-Carr, 91290A100**)
- 2 of thumb nuts (**McMaster-Carr, 96115A410**)
- 1 of mounting base, for clamping down the acrylic tube (**Thorlabs, BA1**)
- 1 of acrylic tube (**McMaster-Carr, 8486K433**)

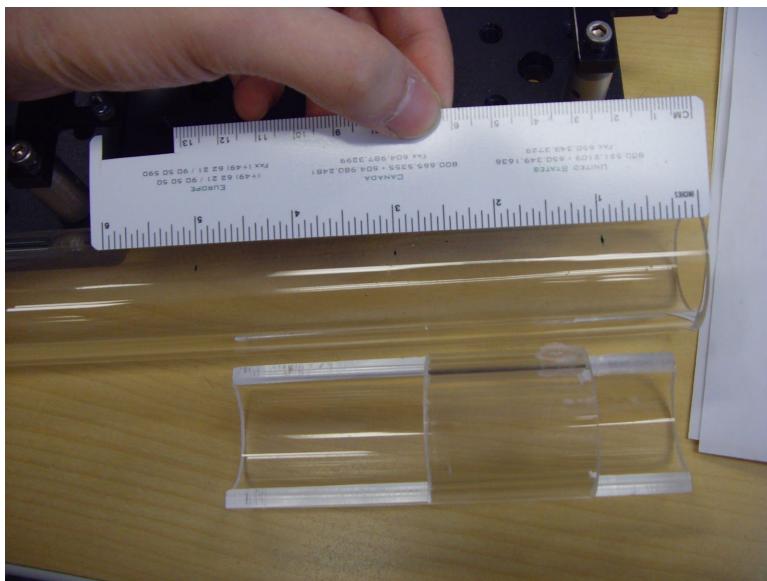
For the large-breadboard base:

- 1 of 8"x8" breadboard (**Thorlabs, MB8**)
- 4 of 4" post (**Thorlabs, TR4**)
- 4 of 1" post (**Thorlabs, TR1**)

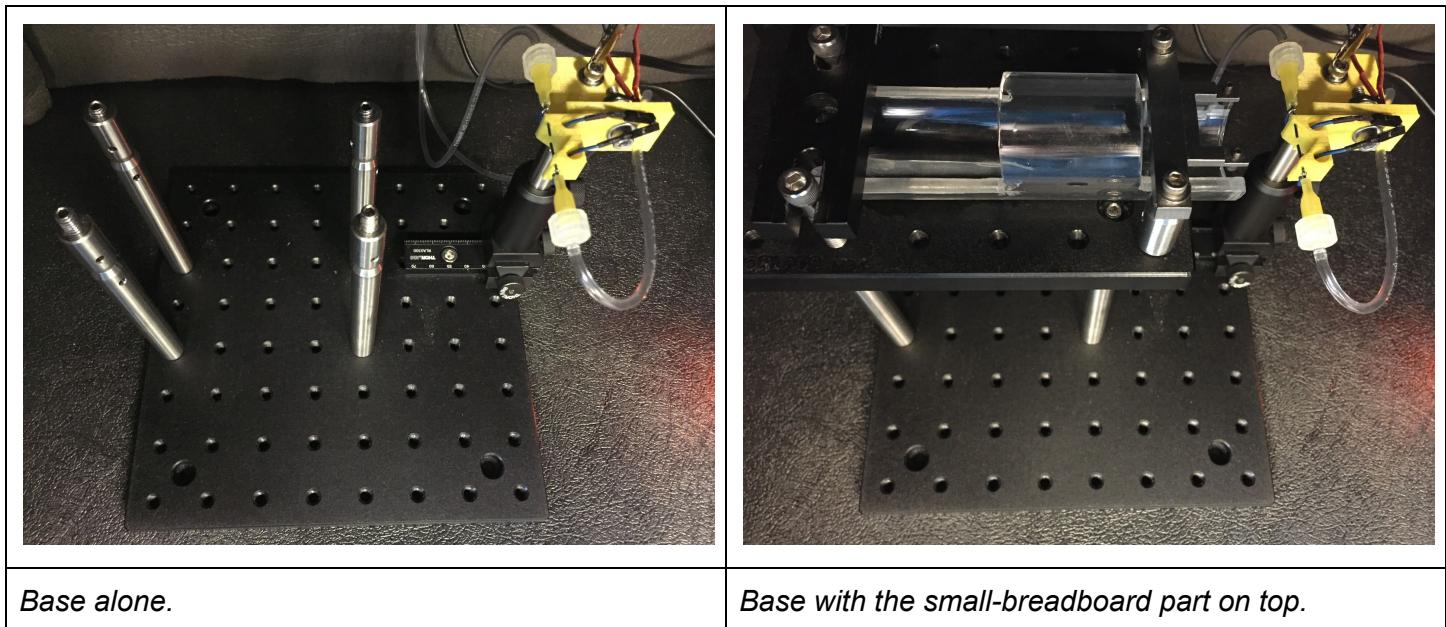
	
AK_holder.ems - <i>send the design file to eMachineShop.com to order multiple copies. Expect several weeks of lead time. Prices decrease when more units are purchased.</i>	AK_PFCheadplate.ems

b. Assembly

- 1) Cut the acrylic tube, e.g. using a bandsaw.
 - Cut a tube with length of 5"
 - For the end where the mouse's head goes, cut at 1" from tube's end, ~3/4 of the way in
 - For the end where the mouse's tail goes, cut at 2" from tube's end, ~3/4 of the way in

	<i>How the cut acrylic tube should look like.</i>
---	---

- 2) Put together the rest of the parts, as the photos show.
- 3) The small-breadboard part should fit onto the large-breadboard base like this:



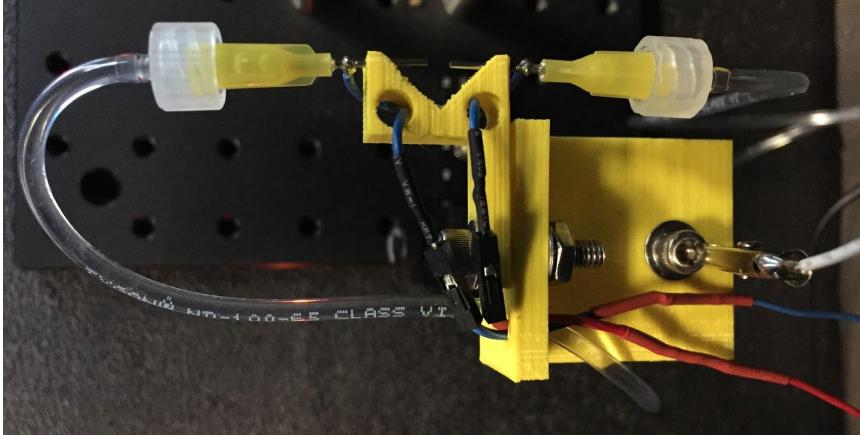
Notes

- Using a metal tube instead of an acrylic tube could help with grounding, and making the lick detection more reliable for some animals.
- If more stability is needed, screws should be used to bolt the small-breadboard part onto the large-breadboard base.

Section 3: Fluid delivery and lick detection

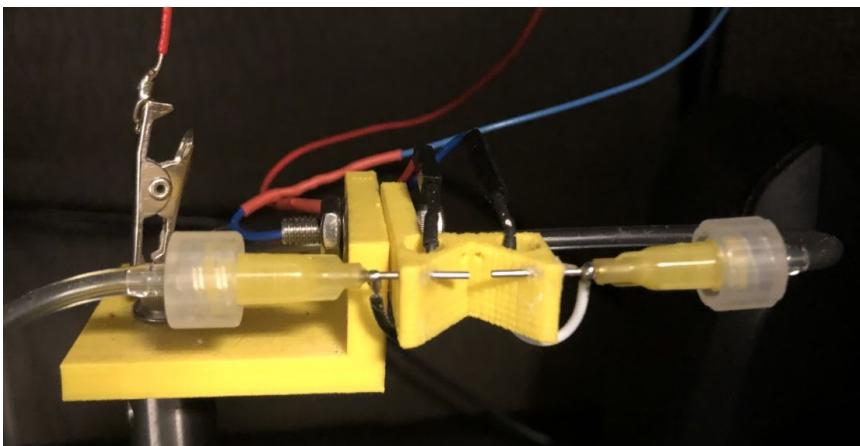
1. Lickometer

The lickometer is placed in front of the mouse to deliver fluid and detect tongue licks. The lickometer has two ports which are made of metal dispenser needles.



The lickometer: top view.

Note the grounding wire with the alligator clip is attached on the screw on the L-shaped piece - temporarily between uses. When the system is in use, that grounding wire should be clipped onto the head plate holder.



The lickometer: the mouse's view.

a. Parts List

- 1 of the V-shaped piece (**3D printed, /printed parts/Lickometer_Lpiece.STL**)
- 1 of the L-shaped adapter to connect lickometer to metal post (**3D printed, /printed parts/Lickometer_Vpiece.STL**)
- 1 of 2" post (**Thorlabs TR2**)
- 1 of 2" post holder (**Thorlabs, PH2**)
- 1 of XY dovetail rails (**Thorlabs, RC3 + RC1 + RLA0300**)
- 2 of metal dispenser needles (**Creative Hobbies, 20 Gauge, 0.5 inch, precision applicator dispenser needle, stainless steel blunt tip**)

<i>Lickometer_Vpiece.STL</i>	<i>Lickometer_Lpiece.STL</i>

b. Assembly

- 1) The dispenser needles have blunt tips at predetermined lengths, and therefore are preferred. If they are unavailable, it is possible to use hypodermic needles, but then should use cutting pliers to cut away the sharp tip for a flat opening, and use a file and sandpaper to further smooth the opening.
- 2) Solder a wire onto each metal dispenser needle. Use a flux pen to help with making a good connection. The other end of the wire will go to ‘Lickometer port (left)’ or ‘Lickometer port (right)’ terminals in the lick detection circuit.

	<p><i>Soldering a wire onto a metal dispenser needle. Using jumper connector for the wire helps with debugging later - if a part is broken, can be replaced quickly with fewer soldering steps.</i></p>
--	---

- 3) Insert the metal dispenser needles into the 3D-printed V-shaped piece. The needles should look centered, and spaced by 3 mm. Affix the metal dispenser needles onto the V-shaped piece using super glue.
- 4) Use screws, nuts, and washers to connect the V-shaped piece onto the 3D-printed L-shaped adapter, and then connect to the post. The post sits in the post holder, which rests on the XY dovetail rails. Using the post holder and the XY dovetail rails, the lickometer can be adjusted in x, y, and z to match the mouse’s tongue position.
- 5) Solder a wire onto an alligator clip. The alligator clip clamps onto the head plate holder. The other end of the wire will go to the positive terminal in the lick detection circuit.



Soldering a wire onto an alligator clip.

Notes

- Multiple metal dispenser needles can be soldered together: to create one composite lick port that can independently deliver multiple types of fluid. See [Barthas F et al., Biol. Psychiatry, 2020](#).

2. Lick detection circuit

Tongue licks onto the lickometer ports are detected by an electric circuit. Essentially, when the mouse's tongue makes physical contact with the metal spout, it 'completes a circuit' and a current is passed through the circuit. The design of the electric circuit is based on [Slotnick B, J Exp Anal Beh, 2009](#).

A few notes about the design:

- SPDT relay is used instead of SPST relay, so that the output voltage is pinned to 0 or 5V. In the original design using SPST relay, when the relay is open, the output voltage is floating, potentially causing uncertainty in the outcome.
- The SPDT relay contains a diode within the package, so there is no need for an external diode.

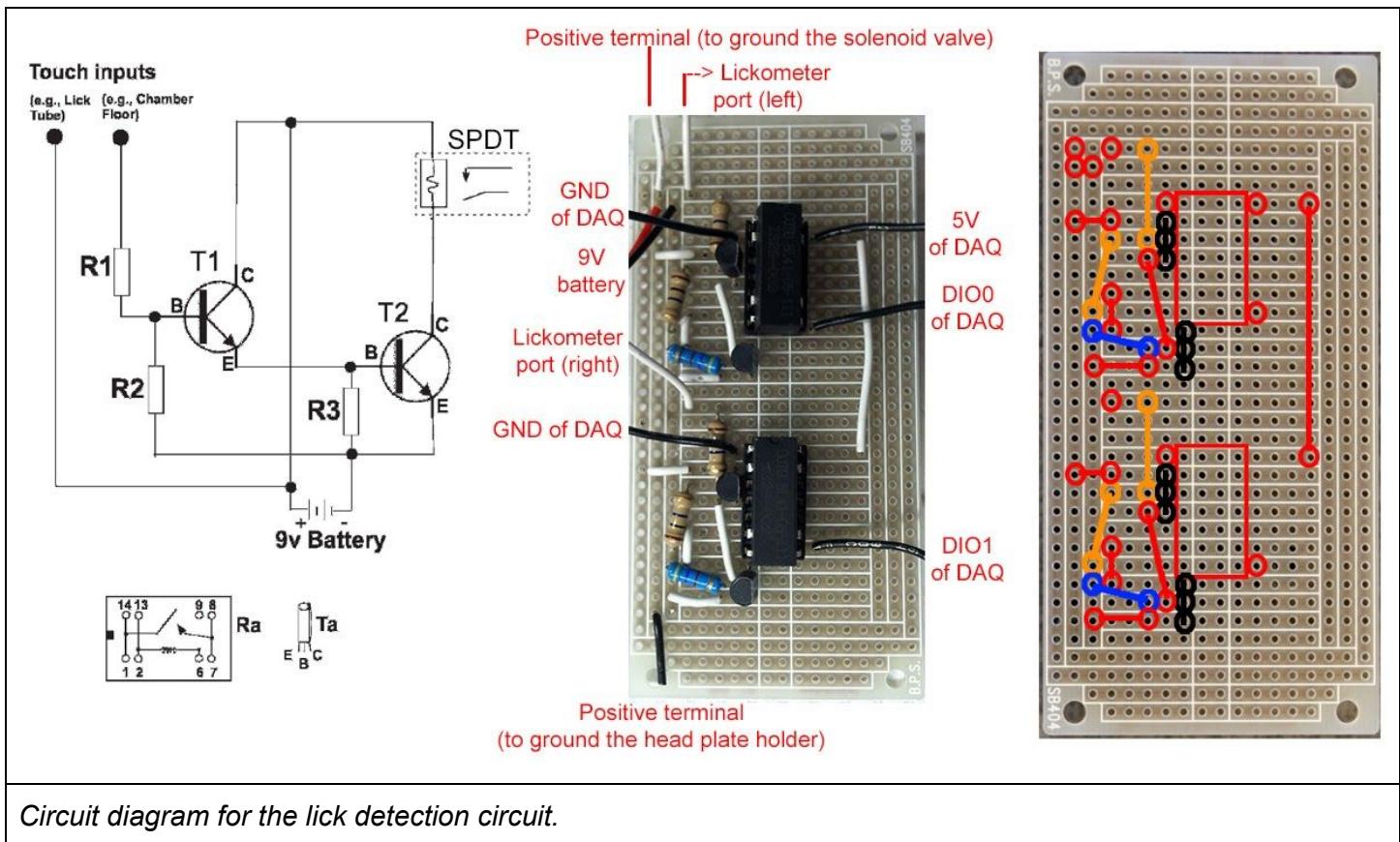
a. Parts List

- 1 of SB404 solderable prototyping board, with power rails, 3.75 x 1.85 inch (**e.g., from Amazon or Jameco, 2191402**)
- 1 of battery snap, 6", 26AWG (**Jameco, 109154**)
- 2 of relay SPDT 5VDC 1ms pull-in-3.8V DIP relay (**Digikey, 306-1046-ND**)
- 2 of 14-pin DIP socket (**Jameco, 526192**)
- 4 of 10MΩ ½ watt resistors (**Jameco, 662377**)
- 2 of 47kΩ ½ watt resistors (**Jameco, 662477**)
- 4 of 2N2222A transistors (**Jameco, 178511**)
- 9V alkaline battery (**Jameco, 198731**)
- 1 of plastic electric project enclosure

- DAQ #1, 8 analog inputs, 12-bit resolution, 100 kS/s, 8 digital I/O (**Measurement Computing, USB-201**)

b. Assembly

- 1) Solder the electronic components onto the prototype board, following the circuit diagram
 - Resistors: $R_1 = R_2 = 10 \text{ M}\Omega$; $R_3 = 47\text{k}\Omega$.
 - For the SPDT relay, connect ground to pin 1, 5V to pin 14, and output voltage to pin 7 or 8.
 - If a 9V battery is used, the wire for the battery snap can break from repeated use. Consider adding a PCB mount screw terminal block connector (2 pole, 2.54 mm pin distance), so the part can be easily replaced without soldering.



- 2) Connect the rest of the wires from the lick detection circuit to the USB-201 DAQ #1 device.
- 3) To reduce noise, two grounding wires are recommended. The first wire goes from the positive terminal to the metal casing of the solenoid valve used for fluid delivery. The second wire goes from the positive terminal to the head plate holder.
- 4) Connect to a 9V battery. When everything is connected, the voltage difference between C and E terminals for both transistors should be 9V.
- 3) The prototype board and the corresponding DAQ #1 can be housed in an enclosure. Ideally, use a plastic enclosure. If the enclosure is metal, make sure to use electrical tapes to cover the soldered side

of the prototype board -- otherwise direct contact with the metal enclosure could short the circuits. Another good solution is to use bone wax or light glue to attach the prototyping board directly on top of the plastic housing of the DAQ.



The lick detection circuit in a metal enclosure atop the cabinet.

Notes

- For more accurate positioning, the XY dovetail rails can be replaced by a XYZ translation stage with differential adjuster (Thorlabs, MT3). Positioning is slightly improved, but the cost is high (~\$1K).
- The lickometer can be motorized to move in and out of position. Control signal sent from the USB-201 DAQ device can be fed to an Arduino (e.g., Arduino UNO R3, DIP, 14DIO) programmed to drive a servo with 180-degree range for 4.8-6V (e.g., Jameco, 283039). The downside is the servo movement makes noise that the mouse can hear.
- Battery is excellent for reducing electrical noise to facilitate electrophysiology. However, over time, the battery will degrade. If the voltage for the battery drops below 8.8V, then lick detection becomes unreliable and the battery must be replaced.
- Instead of the 9V battery, power can be supplied via a wall adapter by modifying the circuit (xxx Huriye will add when the lab re-opens).

3. Fluid delivery circuit

The amount of fluid delivered to each lick spout is controlled by a solenoid valve.

1) Parts List

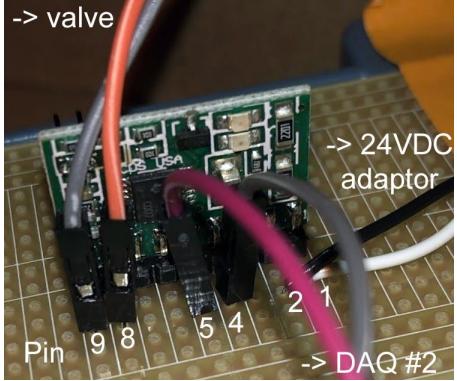
- 2 of solenoid valve for water, M valve, 2-way normally closed, Aluminum body, Viton plunger seal material, $\frac{1}{8}$ " barb body port, 24VDC (**Bordewieck Engineering Sales Co, Gems Sensor Solenoid Valve MB202-V-A-3-0-L-204**) - lead time can be up to a month
- 1/16" i.d., $\frac{1}{8}$ " o.d., Tygon tubing (**Cole-Parmer, EW-95666-01**)
- Various luer locks and hose barb adapters for connecting the plastic tubing, polycarbonate (e.g., the kit sold by Cole-Parmer, EW-45511-00)

- 1 of SB404 solderable prototyping board, with power rails, 3.75 x 1.85 inch (e.g., from Amazon or Jameco, **2191402**)
- 2 of valve driver board, input 8-28 VDC, max current 7A, trigger voltage 4.5-24 VDC (**APW Company, CDS-US-V2**)
- 1 of DC power jack connector (**Digikey, CP-002AH-ND**)
- 1 of 24V AC/DC wall mount adapter (**Digikey, T383-P5P-ND**)

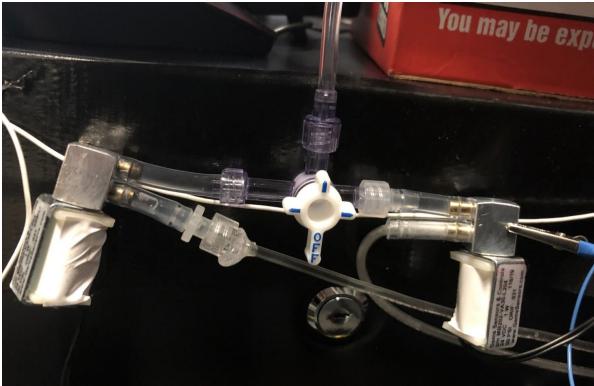
- DAQ #2, 8 analog inputs, 12-bit resolution, 100 kS/s, 8 digital I/O (**Measurement Computing, USB-201**)

2) Assembly

- 1) Solder the two valve driver boards and DC power jack connector onto the prototyping board.

 <p>The fluid delivery circuit. Pins for a valve driver board: Pin 1 = Positive line voltage of the 24VDC power supply Pin 2 = Ground of the 24VDC power supply Pin 4 = GND of the USB-201 DAQ #2 Pin 5 = DIO2 (if this is a board for left lick spout) or DIO3 (for right lick spout) of the USB-201 DAQ #2 Pin 8 = Connects to solenoid valve (to ground wire) Pin 9 = Connects to solenoid valve (to positive, red wire - however, the Gems sensor solenoid valves have no polarity requirement) The blue jumper connector regulates the fluid amount.</p>	<p><i>The fluid delivery circuit. Pins for a valve driver board:</i></p> <p><i>Pin 1 = Positive line voltage of the 24VDC power supply</i></p> <p><i>Pin 2 = Ground of the 24VDC power supply</i></p> <p><i>Pin 4 = GND of the USB-201 DAQ #2</i></p> <p><i>Pin 5 = DIO2 (if this is a board for left lick spout) or DIO3 (for right lick spout) of the USB-201 DAQ #2</i></p> <p><i>Pin 8 = Connects to solenoid valve (to ground wire)</i></p> <p><i>Pin 9 = Connects to solenoid valve (to positive, red wire - however, the Gems sensor solenoid valves have no polarity requirement)</i></p> <p><i>The blue jumper connector regulates the fluid amount.</i></p>
---	---

- 2) The fluid should be stored in a closed container (e.g., in a plastic wash bottle with a squirt neck). The fluid supply should be placed physically above the lickometer, as fluid flow requires gravity.
- 3) Hook up the fluid supply to the input of the solenoid valve, using tubings and luer locks. A single supply can feed multiple solenoid valves. The output of the solenoid valve goes to the metal dispenser needles in the lickometer.



Connecting the fluid supply to two solenoid valves.

- 4) Connect the fluid delivery circuit to the solenoid valves and USB-201 DAQ #2.
- 5) **Calibration!** Note that the amount of fluid delivered is not a linear function of the duration of valve opening. Also the same duration of valve opening could lead to different amounts of fluids in different system due to other factors (e.g., degree of gravity feed and resistance of tubing). The system should be calibrated often - e.g., by writing a script in Presentation to deliver many valve openings (with some inter-valve opening interval to separate individual openings - again because fluid delivered does not scale linearly with duration), collect that fluid using a weighing dish, and then weigh the fluid using a digital scale. Make sure to open each valve independently during each calibration pulse.

Notes

- The full setup requires two USB-201 DAQ units (one for lick detection and one for fluid delivery), because software cannot use one DAQ to generate both digital output (for fluid delivery) and receive digital input (for lick detection).
- We have mostly used the system to deliver water. However, it can also deliver other fluids such as 3% and 10% sucrose solutions. Make sure to flush the tubings and valves every day after use. See [Barthas F et al., Biol. Psychiatry, 2020](#).
- The solenoid valve may contribute to noise in the lick detection circuit, and could cause spurious signals that are detected as licks. Grounding the metal body of the solenoid valve to the positive terminal of the lick detection circuit mitigates the issue. Another possibility is to use a different, plastic-bodied solenoid valve (The Lee Company, LHDA2431415H, LHWX0320090A, LHWX0605450A), which works well in our hands but costs about 3 times more than the Gems sensor solenoid valves.

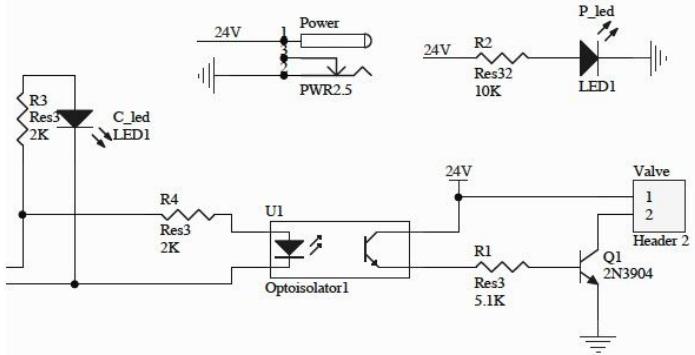
4. Fluid delivery circuit (obsolete)

This is the fluid delivery circuit for rigs built prior to circa 2019. The information below is kept for backward compatibility (for maintenance of existing rigs) and to provide an alternative.

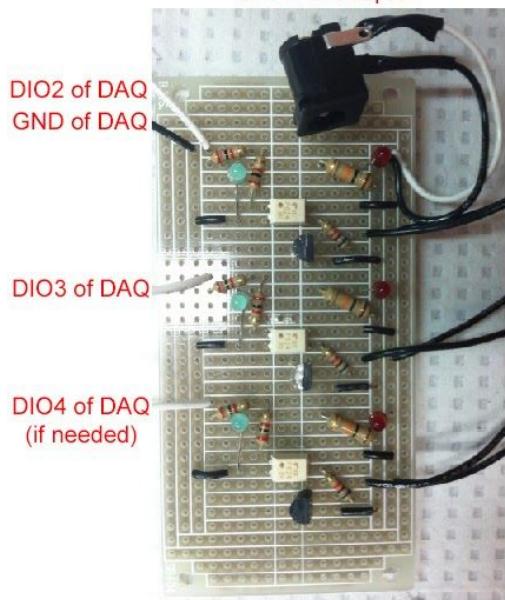
1) Parts List

- 1 of SB404 solderable prototyping board, with power rails, 3.75 x 1.85 inch (**e.g., from Amazon or Jameco, 2191402**)
- 1 of 24V Power supply (**Digikey, T383-P5P-ND**)
- 1 of DC Power connector (**Digikey, CP-002AH-ND**)
- 3 of Green LED
- 3 of Red LED
- 6 of 2 kΩ resistors
- 3 of 5.1kΩ resistors
- 3 of 10 kΩ resistors
- 3 of transistors 2N3904 (**Digikey, 2N3904TFCT-ND**)
- 3 of optoisolators, 4-DIP and 0.3" (**Digikey, TLP621BLTF-ND**)
- 1 of plastic electric project enclosure
- DAQ #2, 8 analog inputs, 12-bit resolution, 100 kS/s, 8 digital I/O (**Measurement Computing, USB-201**)

2) Assembly



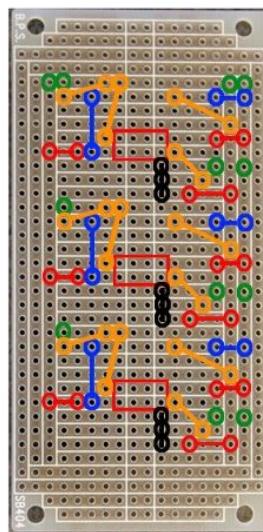
Power jack to
24V wall adapter



Valve
for left port

Valve
for right port

Third valve
(if needed)

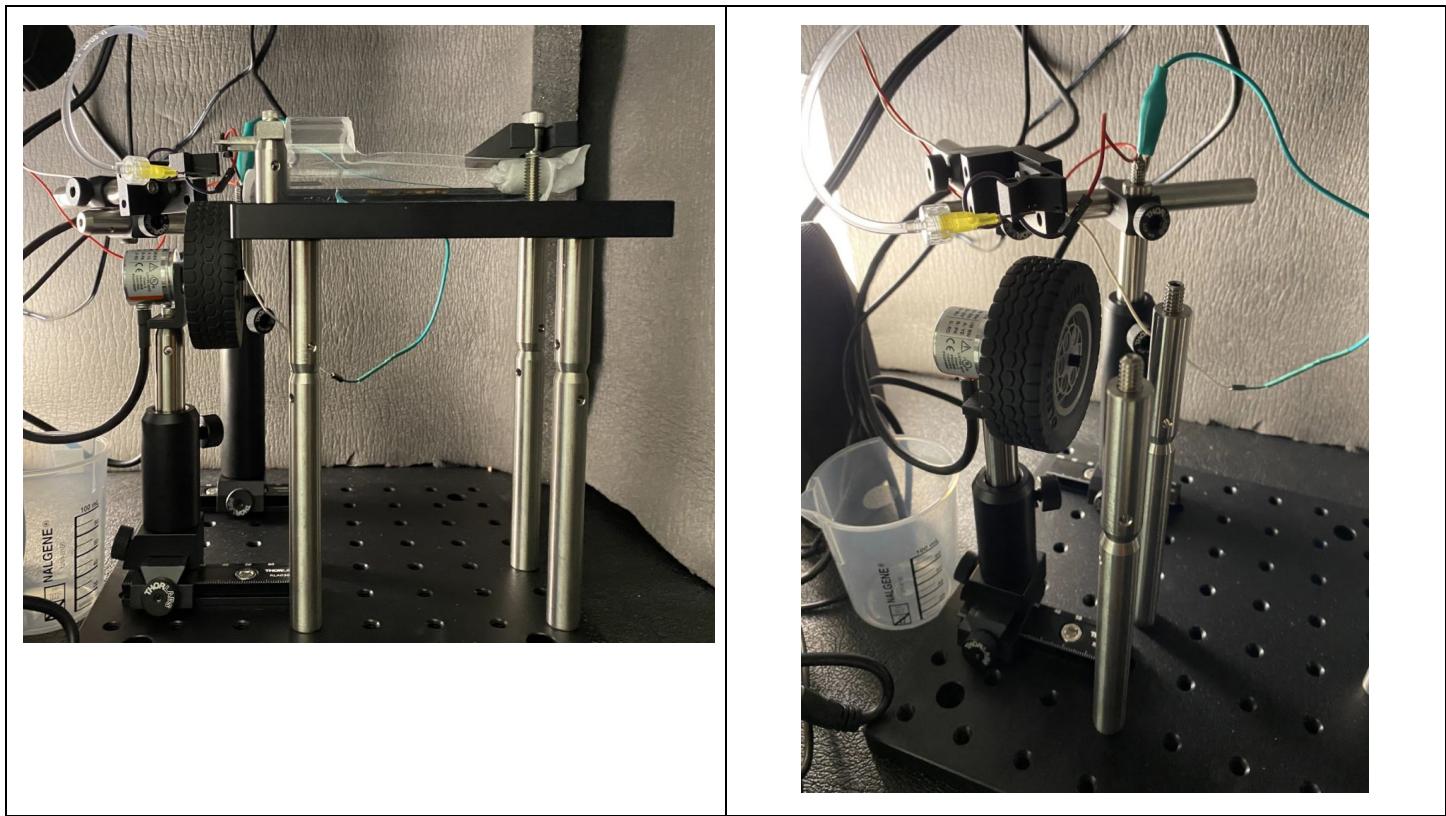


Circuit diagram for the old fluid delivery circuit.

Section 4: Steering wheel

1. Steering wheel

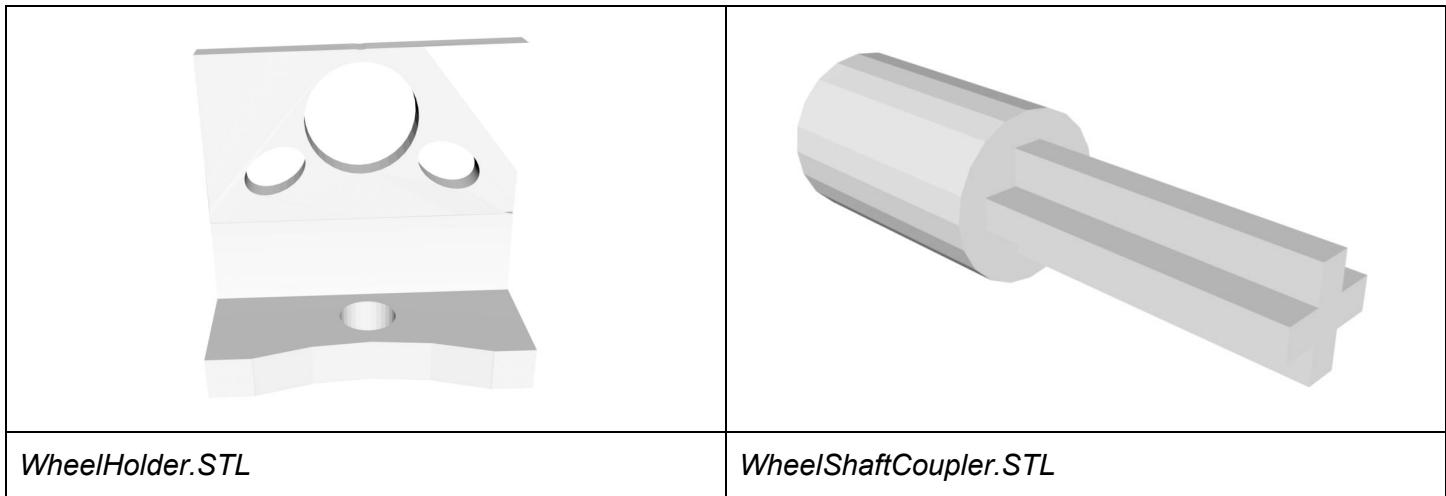
The mouse could use a steering wheel to report its choices in each trial. The steering wheel setup is adapted from the [Carandini lab](#). Briefly, the setup uses a LEGO wheel mounted on an incremental rotary encoder, which passes state information to an Arduino microcontroller. The Arduino board runs a serial decoding program based on rapidly refreshing encoder state, in turn outputting data to a USB-205 DAQ that NBS Presentation reads to detect clockwise and counterclockwise wheel rotations. The steering wheel setup looks like this:



1) Parts List

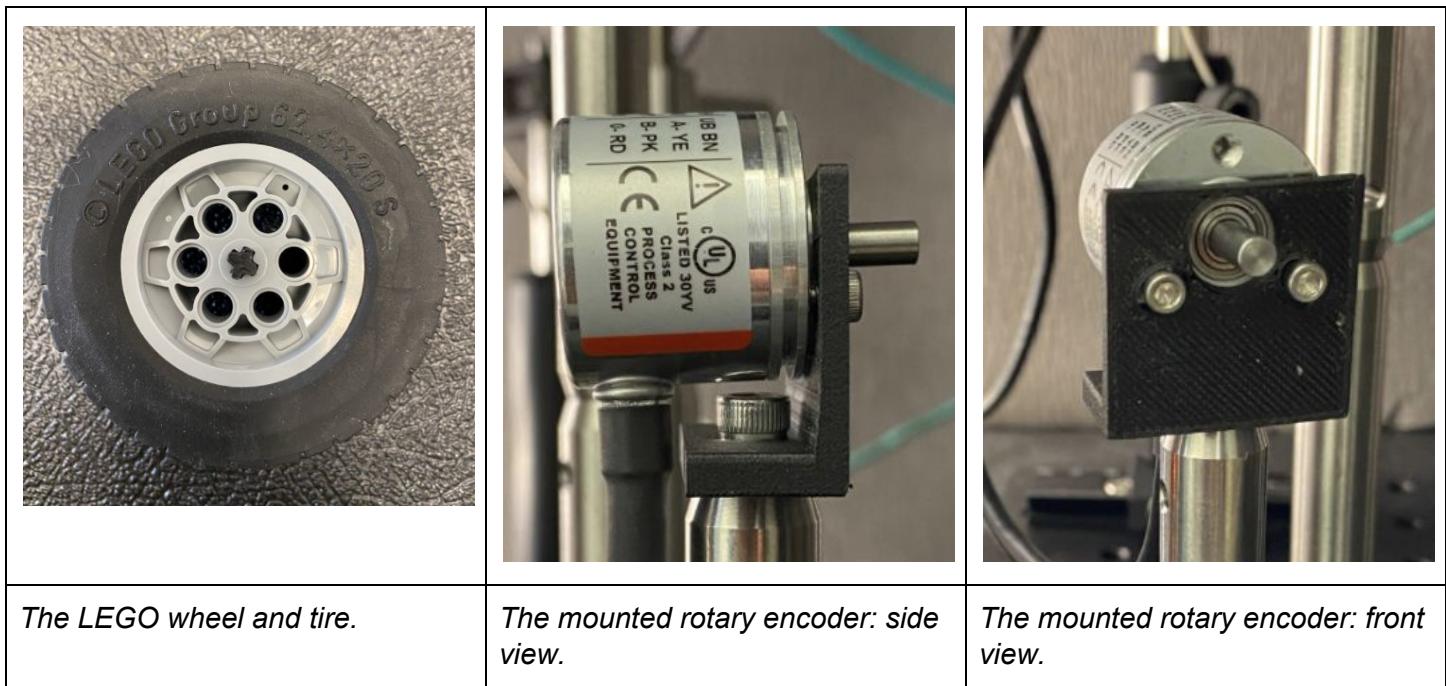
- LEGO wheel 43.2 mm dia. X 18 mm ([Lego, 86652](#))
- LEGO tire 62.4 mm dia. X 20 mm ([Lego, 32019](#))
- Rotary encoder, optical, incremental, 10 mm long 4 mm diameter shaft, push-pull, 1024 PPR resolution, 5 - 24V ([Kuebler, 05.2400.1122.1024](#))
- 1 of the wheel holder piece (3D printed, [/printed parts/WheelHolder.STL](#))
- 1 of the wheel shaft coupler piece (3D printed, [/printed parts/WheelShaftCoupler.STL](#))
- 1 of 1.5" post ([Thorlabs TR1.5](#))
- 1 of 1.5" post holder ([Thorlabs, PH1.5](#))
- 1 of XY dovetail rails ([Thorlabs, RC3 + RC1 + RLA0300](#))

- Arduino microcontroller (**Arduino, Uno R3**)
- DAQ, 8 analog inputs, 12-bit resolution, 500 kS/s, 2 analog output, 8 digital I/O (**Measurement Computing, USB-205**)

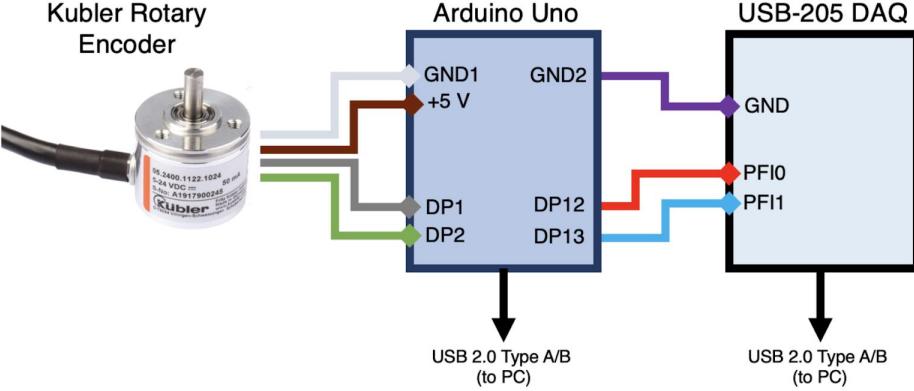
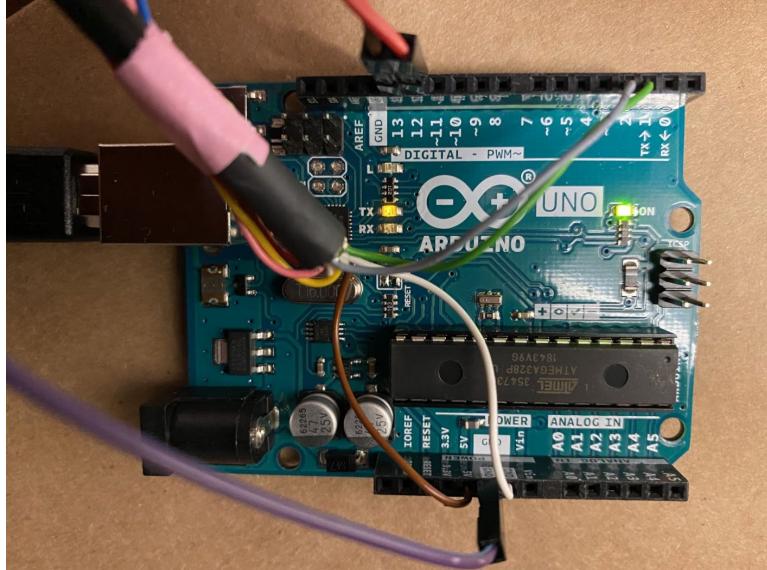


2) Assembly

- 1) The rotary encoder is fixed to the 3D printed wheel holder, which is fastened onto the Thorlabs post, post holder, and XY dovetail assembly.
- 2) The LEGO wheel goes into the 3D printed wheel shaft coupler, which connects to the shaft of the rotary encoder. To make a tight fit between the wheel shaft coupler and the shaft of the rotary encoder, you may want to apply tape onto the shaft.



- 3) The rotary encoder comes with an 8-wire output. Connect 4 of the wires to the Arduino microcontroller: white wire to GND pin, brown wire to +5V pin , gray wire to digital pin 1, and green wire to digital pin 2. The unused wires (red, yellow, pink, and blue) can be taped to avoid unintended connections.

	<p><i>Circuit diagram for the steering wheel.</i></p> <p><i>Note: The positions in this cartoon is not meant to be the position of the pins - read the pin labels.</i></p>
	<p><i>Arduino with the connections.</i></p>

- 4) Connect the Arduino microcontroller to the USB-205 DAQ: GND pin goes to GND of DAQ, digital pin 12 goes to DIO0 of DAQ, and digital pin 13 goes to DIO1 of DAQ.

Notes

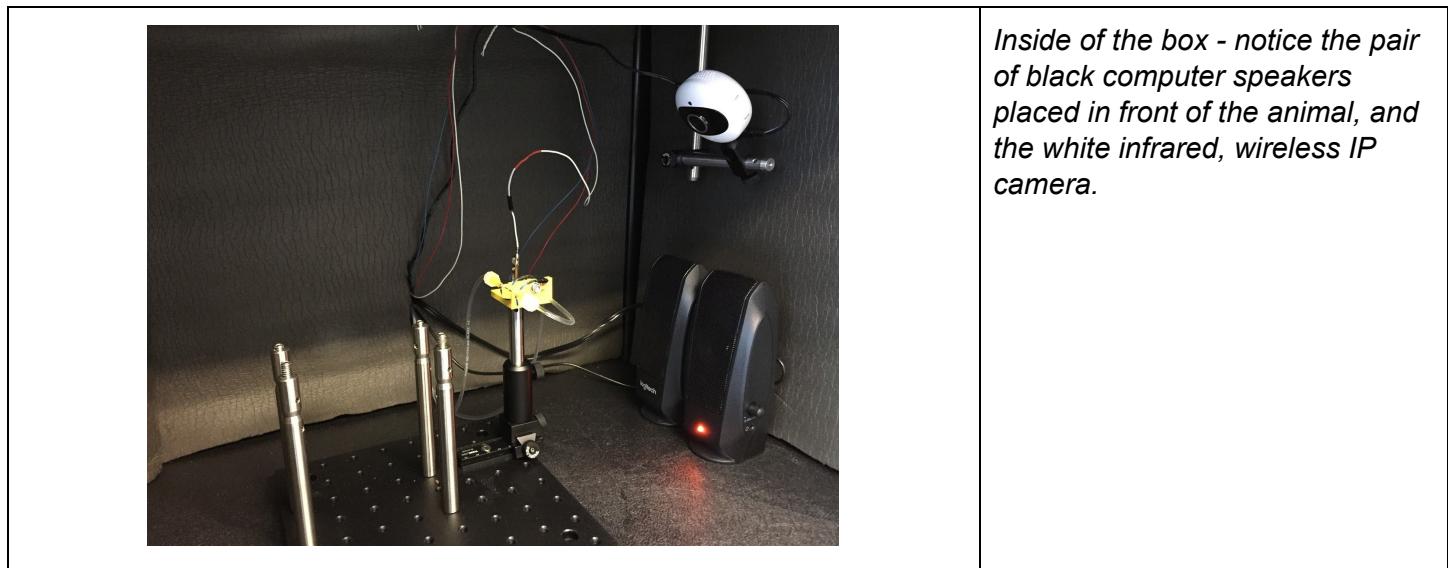
- For the DAQ, USB-205 was used instead of USB-201 because the latter device was out of stock at the time. The USB-201 should work as well?

Section 5: Computer control

1. Control of the main behavioral rig

a) Parts List

- 1 of desktop computer with monitor
 - If using visual stimulus, need a video card with 2 outputs (e.g., HDMI, VGA) for the Presentation software to simultaneously display the progress screen and to present the visual stimulus.
Visual stimulus may be delivered using a small monitor (e.g., for use inside a car)
- 1 of mini CCTV camera, low light or infrared light, manual focus, USB or wireless IP
- 1 of computer speaker (**e.g., Logitech, S120**)



b) Assembly

- 1) Position and mount the infrared camera to get a live view of the animal's head and lickometer. This is useful for debugging -- visually seeing what the mouse is doing versus what the computer is detecting.
- 2) Install - Instacal from [Measurement Computing](#). This software is required for the DAQs to communicate with Presentation. When you open Instacal, you should see the DAQs that are connected to the computer.
- 3) Install - Presentation from [Neurobehavioral Systems](#).

4) In Presentation, under the “Settings” tab:

- For lick detection:
 - Setting → Ports → Port Properties: You should see two MC Universal Library (Board 0 / Board1). Select and add the board that is associated with the lick detection circuit.
 - Settings → Ports → Input Ports panel: Port input channel: USB 201 #0 (data source: AUXPORT, register span: 1, status port, no mask, inversion mask = 0, no interrupt, check box for independent lines). Uncheck the box for Log Codes.
 - Setting → Response → Port device (USB- 201 #0) → Buttons. Add 1(start) and 2(start) as active responses. If you don't see the buttons on the input port, then the Input Ports were not configured properly. Set these parameters for both the default and the current .sce
 - Settings → Ports → Input Ports panel: Click Test, you should see value(s):1 when you activate the left port of the lickometer by touching it with the positive ground cable, and you should see value(s):2 for the right port of the lickometer.
- For fluid delivery:
 - Setting → Ports → Port Properties: You should see two MC Universal Library (Board 0 / Board1). Select and add the board that is associated with the fluid delivery circuit.
 - Settings → Ports → Output Ports panel: Port: USB-201 #1, data source: AUXPORT, register span: 1, check independent lines, no delay codes, inversion mask: 0.
 - Settings → Ports → Output Ports panel: Click Test, Enter code 4, pulse length 1000 (which is 1 second) - you should see fluid coming out of the left port of the lickometer. Enter code 8 for the right port of the lickometer. If you don't see any fluid, can you hear the solenoid valve clicking? May have to open the valve for a longer time to flush out all the air in the tubing. The code 4 and 8 work because we are using DIO2 and DIO3 on the DAQ for the trigger signals.
- Other Settings:
 - Setting → Advanced: Random number generator: Mersenne Twister
 - Setting → Logfiles: check Prefix subject name, check Append counter if exists, Subject Name/ID set to Prompt
 - Setting → General: uncheck Wait for return to start; check Never show report

Notes

- Example scripts to use with Presentation
 - **/presentation scripts/shaping** (common scripts used for initial habituation of animals)
 - **/presentation scripts/flexible auditory** (flexible sensory-motor coupling)
See [Siniscalchi et al., Nat. Neurosci., 2016](#)
 - **/presentation scripts/instrumental sucrose** (instrumental sucrose preference)
See [Barthas et al., Biol. Psychiatry, 2020](#)
 - **/presentation scripts/bandit** (probabilistic reward reversal)
 - **/presentation scripts/matching pennies** (matching pennies game)

- The computer can run a remote desktop software (e.g., TeamViewer), so the progress can be monitored elsewhere.
- To synchronize with other equipment (e.g., a two-photon microscope or patch clamp amplifier), some of the DIO pins in the USB-201 DAQ #2 are connected to the equipment. The scripts are programmed to send TTL pulses to trigger the other equipment. In this way, Presentation acts as the master, and the other equipment are the slaves.

2. Control of the steering wheel

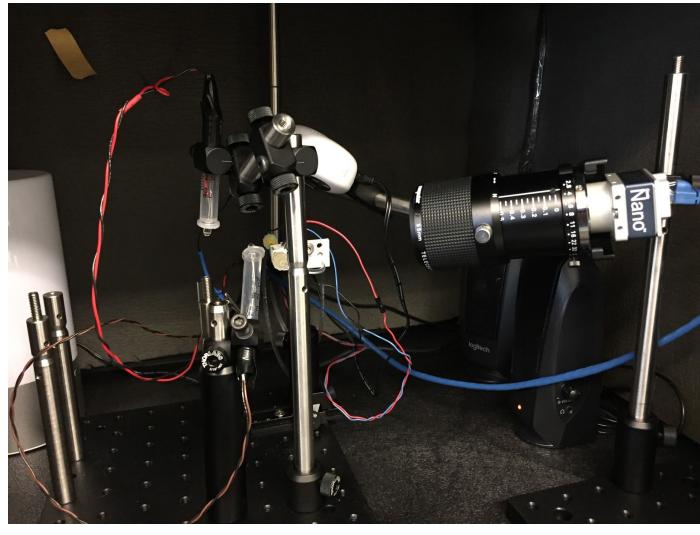
a) Assembly

- 1) Install - Arduino IDE software from [Arduino](#).
- 2) Transfer into the Arduino the code for the serial encoder (xxx to add - Neil). It is recommended you download some serial decoding software (e.g., PuTTy [recommended], CoolTerm; allows for live stream/reading Arduino output) for troubleshooting the code involving the Arduino board.
- 3) Verify that the COM port being used for the Arduino is reading 115200 bits per second with 8 data bits:
Open Control Panel -> Device Manager -> Ports (COM & LPT) -> Arduino Uno (usually COM3) -> right click -> Properties-> Port Settings -> Set Bits per second: 115200; Data bits: 8; Parity: None; Stop bits: 1; Flow control: None;
- 4) Set up USB-205 as an input port in Presentation.

Section 6: Pupillometry

1. Pupillometry

The pupillometry setup allows for recording of the mouse's eye movements and pupil size while it is performing behavior tasks. The main components of the pupillometry setup can be seen in the photo below.



The camera has a telecentric lens, and is mounted on a post and breadboard assembly. The infrared illumination is housed in a hard syringe casing near the lickometer. To the far left is the table lamp, providing ambient room light.

a. Parts List

For the illumination:

- 1 of infrared LED, 940 nm, 100 mA, radial (**Digi-key, 67-1000-ND**)
- 1 of 78Ω resistor
- 1 of DC power jack connector (**Digikey, CP-002AH-ND**)
- 1 of 9V AC/DC wall mount adapter
- 1 of hard housing for the LED and the wiring (e.g., a 5 mL syringe)
- 2 of 3" post (**Thorlabs, TR3**)
- 2 of rotating post clamps (**Thorlabs, SWC**)
- 1 of 2" post holder (**Thorlabs, PH2**)

For the camera:

- 1 of camera (e.g., **Dalsa, G3-GM11-M1920**, GigE, 1936 x 1216, 38 fps, CMOS sensor, C-mount lens mount)
- 1 of camera lens (e.g., **Computar, TEC-55**, 55 mm telecentric lens, C-mount, 140 mm - infinity working distance, fast F2.8 F-stop)
- 1 of 6" post (**Thorlabs, TR6**)
- 1 of 2" post holder (**Thorlabs, PH2**)
- 1 of rotating post clamps (**Thorlabs, SWC**)

- 1 of clamping fork (**Thorlabs, PF175**)
- 1 of breadboard (**Thorlabs, MB4**)
- 1 of lamp to provide ambient room light (e.g., **Aukey, LT-T6**, dimmable, 6W white light, 300 lm max)

For computer control:

- 1 of computer (**requirement: CPU 1.6GHz, RAM 16GB, a computer case large enough for the video capture card**) - this is a different computer than the one running Presentation
- 1 of video capture card (e.g., **Adlink, PCIe-GIE64+**, GigE frame grabber supporting power over ethernet)
- 1 of ethernet cable (**Category 5e or higher**)
- DAQ #3, 8 analog inputs, 12-bit resolution, 100 kS/s, 8 digital I/O (**Measurement Computing, USB-201**)

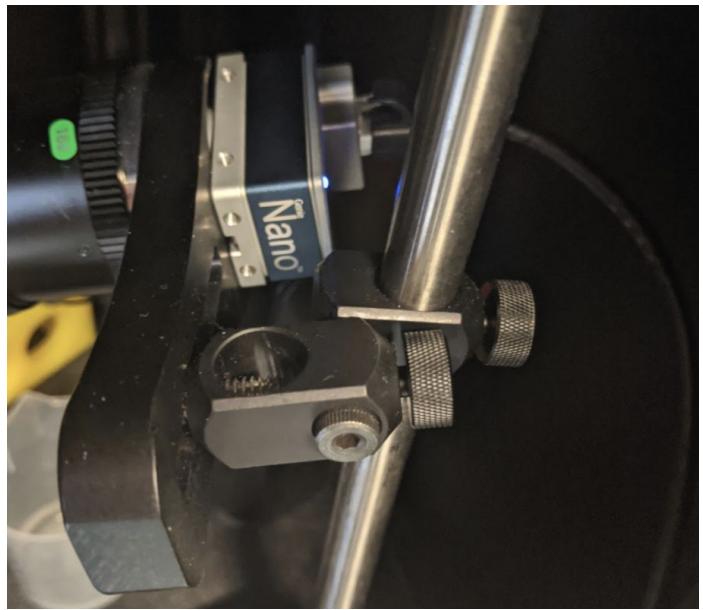
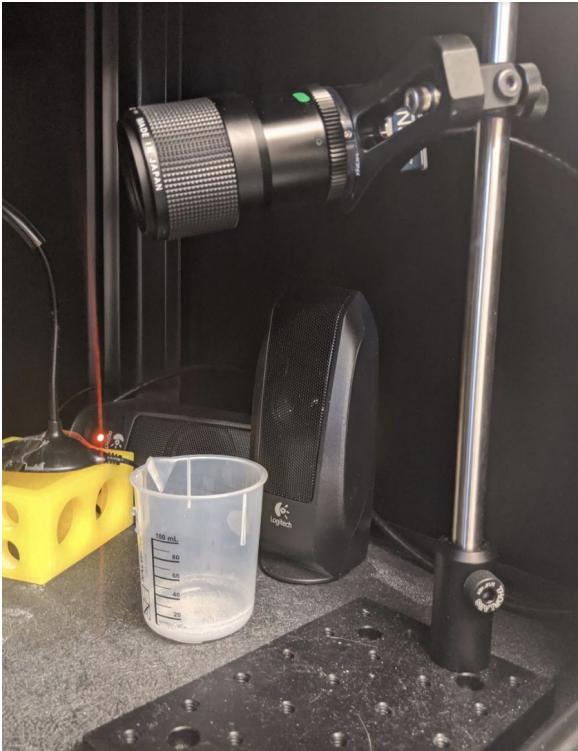
b. Assembly

- a) Solder the LED, 78Ω resistor, and the DC power jack connector in series. Protect the LED and wiring with a hard casing, e.g. using a syringe (see photo below). Power it with the 9V AC/DC wall adapter. Use posts and rotating post clamps so the angle and position are adjustable.



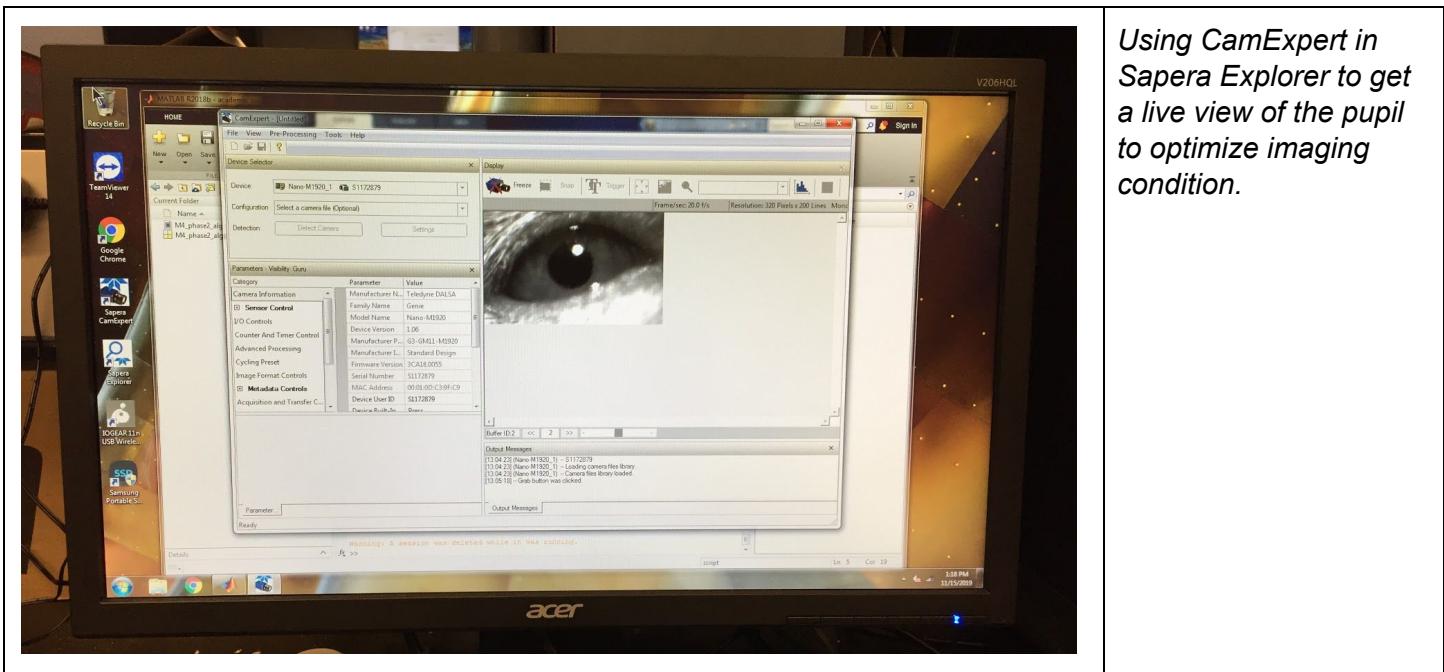
The LED illumination.

- b) For the computer, install the video capture card. Connect the camera to the computer using the ethernet cable.
- c) Put the telecentric lens on the camera, and mount the camera using the parts as shown in the photos below.



- d) Connect the USB-201 DAQ #3 to the computer for pupillometry.
- e) Connect 2 DIO pins and a GND pin of the USB-201 DAQ #2 (for fluid delivery and equipment triggering) to 2 analog input pins and a GND pin of the USB-201 DAQ #3 (for pupil), respectively. Note that the analog input channels have different scan rates (refer to the [USB-201 manual](#)).
- f) In Presentation, in the scenario file that runs the task, in the end of each trial, send one output code through one of the DIO pin of the USB-201 DAQ #2 for 10 ms (later we will use this to align the pupil recording to the behavioral times, the duration of the code may be adjusted depending on the system and hardwares), and also at the end of the session, send another output code through the other DIO pin of the USB-201 DAQ#2 (to stop the pupil recording).
- g) Set up the computer for pupillometry: Install MATLAB (2017b or above), Instacal, and Sapera Explorer with SDK (from Teledyne Dalsa). In MATLAB, download the Image Acquisition Toolbox Support Package for GigE Vision Hardware, and the Measurement Computing Data acquisition Toolbox. Follow the steps in the guide '[GigE vision Quick Start Configuration Guide](#)' to set up parameters for GigE.
- h) Optimize the imaging condition on the computer for pupillometry: Open Sapera Explorer to grab an image. Adjust the magnification and the camera's focus until you can see a reasonable image. Place

the mouse's pupil in the middle of the field of view. Adjust the infrared LED so that the pupil is lit clearly. Trim the whiskers if they obscure the pupil. Finally, close Sapera Explorer.



Using CamExpert in Sapera Explorer to get a live view of the pupil to optimize imaging condition.

- i) To acquire the pupil video: In MATLAB, run the acquisition code which is available at **/code/MATLAB/pupil**. Run the file [startPupilDataRecording.m](#).