Arduino Nanocourse Day 2 Project

Behavior Box Circuit Project

Many neurobiology experiments involve interacting with a behaving animal and monitoring its actions. In this project you will get a feel for the sensors and actuators relevant to a behavioral setup by building a ‘behavior box’. This is a device used to train mice on operant conditioning tasks.

Goal

Construct a rudimentary behavioral training setup that:

1. Triggers the start of a trial
2. Checks for some response
3. Delivers a reward

Constraints

Each trial should last 30 seconds (starting after the trigger is presented). If the mouse responds within three seconds after the trigger, a reward should be given.

Components

Wire color convention

To make debugging easier, we recommend using a consistent wire color convention. This is good practice not just for this class, but for future electronics projects as well.

BLACK or **GREEN** wire for GROUND nodes
RED wire for POWER
Pick some other color for signal lines
If your circuit uses a negative supply (nothing in this class does this), use WHITE wire.

In the kits, we provide an assortment of wires in different colors. You may not have exactly enough to strictly maintain the color convention above. Do the best you can.

Triggering a trial

To trigger the start of a trial, use a piezo buzzer to produce an audible tone. The buzzer is a small plastic can with two leads and a small opening at the top. Plug the buzzer into the prototyping board. Connect one lead to one of the Arduino digital pins. Connect the other lead to ground. As an option, you can use a resistor in series with the buzzer to limit the volume. See

the tone function for more information on how to use the buzzer :
https://www.arduino.cc/en/Reference/Tone

Detecting a response

There are multiple ways to detect a response. We are providing two options. Please select only one.

Photointerrupter

By selecting this component you've decided to detect a 'nosepoke' by using a photointerrupter to detect when the mouse pokes its nose into a small gap in the sensor. Inside the sensor is an LED paired with a photo-transistor. The LED provides infrared illumination, while the photo-transistor responds to the light and amplifies the photocurrent. The output is a digital "high" when not occluded and "low" when the light beam is blocked. The photointerrupter is mounted on a small circuit board which includes the current limiting resistor for the LED. To wire it up, please use a red jumper wire to connect PWR to the 5V pin. Connect GND with a green or black wire to one of the GND pins on the Arduino or to the GND bus on the breadboard. The SIG pin should be connected to whichever digital Arduino pin you've specified as the input in setup() The datasheet and connection details can be found here:
https://www.sparkfun.com/products/9299
https://www.sparkfun.com/products/9322

Temperature sensor

By selecting this component you've decided to detect the presence of the mouse by looking for a temperature change. In the instrument, this sensor might be attached to a metal plate on which the mouse stands, but for this course you'll only be using the sensor. The sensor you will use is TMP36 (https://www.sparkfun.com/products/10988); it outputs an analog voltage that is linearly proportional to temperature within its range (10mV per degree C). Connect the sensor power and ground pins. Connect the output pin to one of the *analog in* pins on the Arduino. Please note that the TMP36 output ranges from 100mV to 2V and consult the datasheet for additional details. Also, please note that changes in temperature occur slowly, and it may take several seconds for the sensor to register a response.

Administering a reward

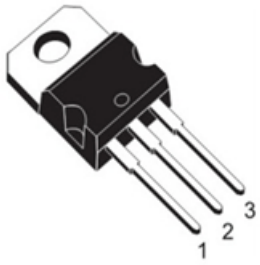
For a specific experiment, you may want to deliver either a liquid or solid reward. This constraint would, in part, determine what actuator you use to administer the reward. We provide you with a servo motor and a DC motor. Please pick one for reward delivery.

Servo

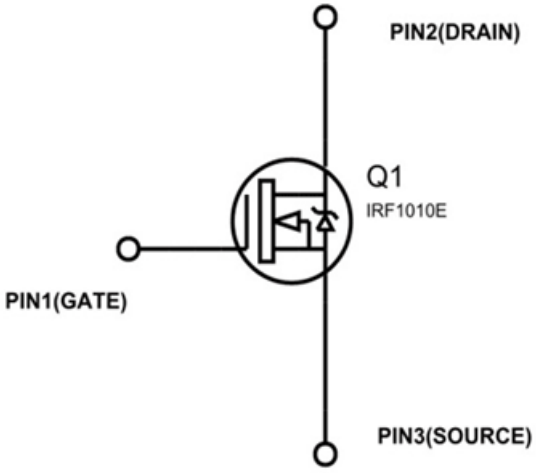
Servo motors are rotary actuators that use feedback to precisely control their position. In a real experiment the servo might be attached to a container of food pellets so that when the servo arm moves to a set location and then returns, a pellet is delivered. The servo we provide can rotate through a 180 degree arc. The servo has 3 terminals: Brown or Black wire should be connected to ground; Red to +5V; Orange, Yellow or White to the control signal. We recommend using the Arduino library <Servo.h> to control the servo. ~~(Note that this library will control servos only on pins 9 and 10.)~~ More information and examples can be found on the Arduino website.
https://www.arduino.cc/en/Reference/Servo

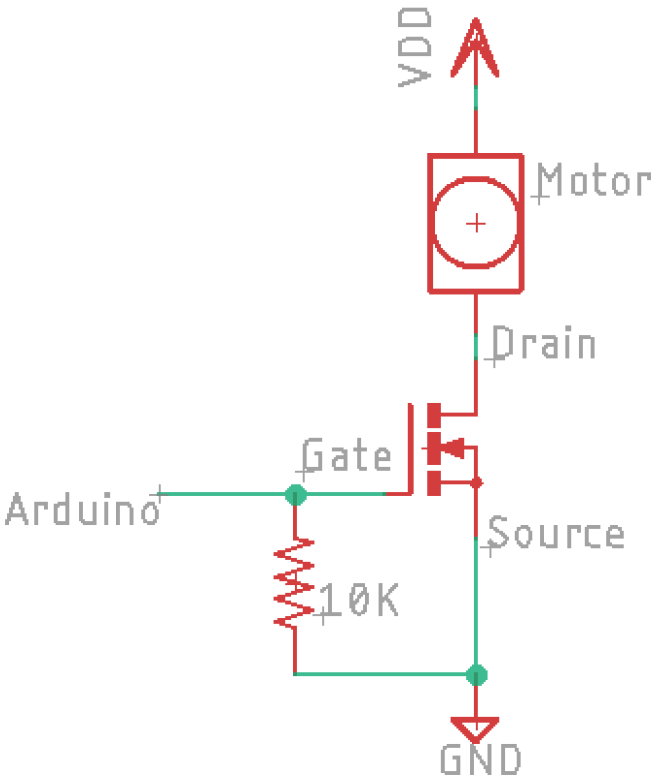
DC Motor

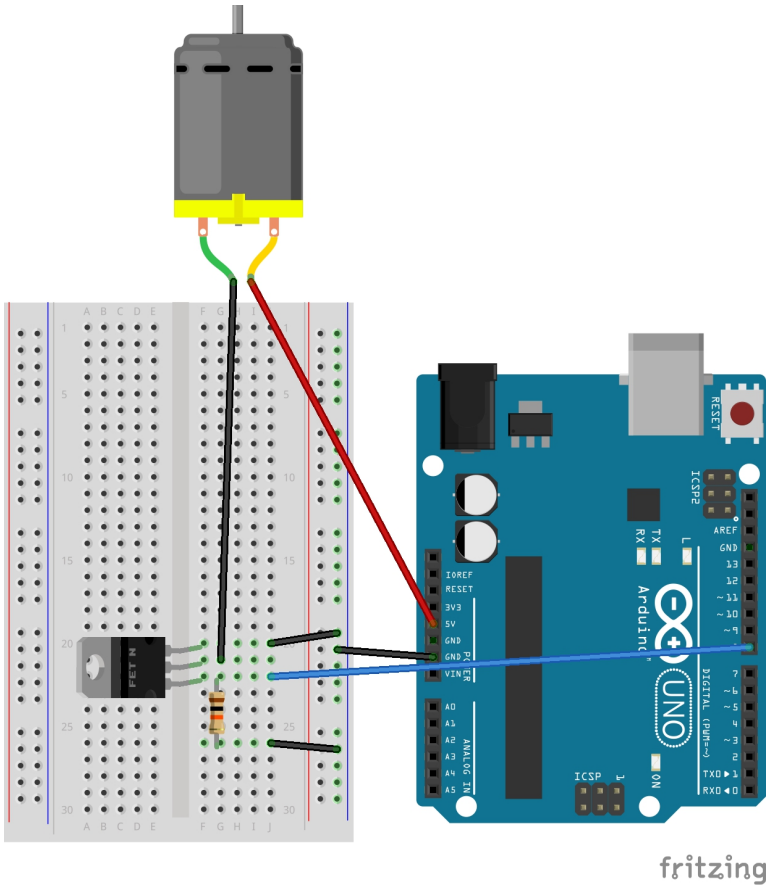
Unlike a servo, a DC motor does not allow for position control. When voltage is applied, it spins at a speed proportional to the voltage. In an experiment, a motor might power a small pump to deliver a liquid reward to the animal. The motor used in this class will spin from voltages ranging from about 1V up to 12V. However, even at 1V it draws over 100 mA of current, which is more than any single Arduino pin can source. To power this motor you should use a transistor (we provide you with DigiKey part 1727-5893-nd). Connect the *Gate* of the transistor to an Arduino IO pin that you plan to programmatically control. Connect the *Source* to ground. Connect the *Drain* to one terminal of the motor, and connect the other motor terminal to the 5V power rail. Please follow the schematics below to connect the motor and the transistor to the Arduino. The 10K resistor is optional, but it will prevent the motor from turning on accidentally.
https://www.sparkfun.com/products/11696



TO-220




VDD
Motor
Drain
Gate
Arduino
Source
10K
GND


fritzing

Control Logic

This project introduces several new ideas. You will need to time your events while your code is doing something else. You can utilize the built-in function `millis()`, which returns the time in milliseconds since the Arduino board started.

As with the previous project, it's best to build up your functionality in parts. Make sure to debug individual components before moving on. You should make use of the serial communication port to understand what the microcontroller is doing. Your general program flow might be something like this:

```
variable declaration

void setup(){
    specify input and output pins
    Serial.begin(9600);
}

void loop() {

    update trial counter
    tone (buzzerPin, pitch, duration_ms);
    trialStartTime = millis(); //current time

    while (within reward window){
      if (sensor is tripped) {
        deliver reward
        Serial.print("reward delivered at time ");
        Serial.println(millis());
      }
    }
    while (within trial){
      if (sensor is tripped){
      //unrewarded attempt
      Serial.print("unrewarded nose poke");
      }
    }
}
```

Bonus

If you have additional time or want experience with more components. Repeat the project using the other options for the sensor and actuator. Do you have to modify the control code? How might you write control code that would work for any combination of sensors and actuators?

Punishment option

Vibration Motor

Vibration motors are frequently used in cellphones for haptic feedback. In a behavioral experiment it might be used to punish the animal for an incorrect choice. The motor we provide is rated to work at no more than 3.6V. Because a digital HIGH from the Arduino is 5V, you should not connect the motor directly to a DIO pin or the 5V pin (connecting to 5V very briefly to confirm that it works should be OK). To control the motor we recommend using a transistor. Just like you did with the DC motor, connect the *Gate* of the transistor to one of the Arduino IO pins. Connect the *Source* to ground. Connect the *Drain* to one terminal of the motor, and connect the other terminal to the 3.3V pin. In this configuration, the transistor is controlled by one of the digital IO pins that output 5V, but it switches 3.3V to the motor. We recommend taping the motor to the breadboard to keep it from ‘walking away’.
https://www.sparkfun.com/products/8449