**Feeding Experimentation Device (FED)**

Protocol updated Oct 5, 2016

**Contributors:**

Katrina P Nguyen

Mohamed A Ali

Timothy J O’Neal

Julia A Licholai

Ilona Szczot

Elecia White

Alexxai V Kravitz

**Funding:**

Development of FED was funded by the National Institutes of Health Intramural Research Program

**SHORT DESCRIPTION:**

Feeding Experimentation Device (FED) is an open-source device for measuring food intake in mice. FED can also synchronize food intake measurements with other techniques via a real-time digital output. Here, we provide a step-by-step tutorial for the construction, validation, and usage of FED.

**ACKNOWLEDGMENTS:**

Funding was provided by the Intramural Research Program at the National Institutes of Health. We thank the NIH Section on Instrumentation and the NIH Library for assistance with 3D printing, and all members of the Kravitz lab for general help with FED.

**DISCLOSURES:**

The authors declare no conflict of interests, financial or otherwise.

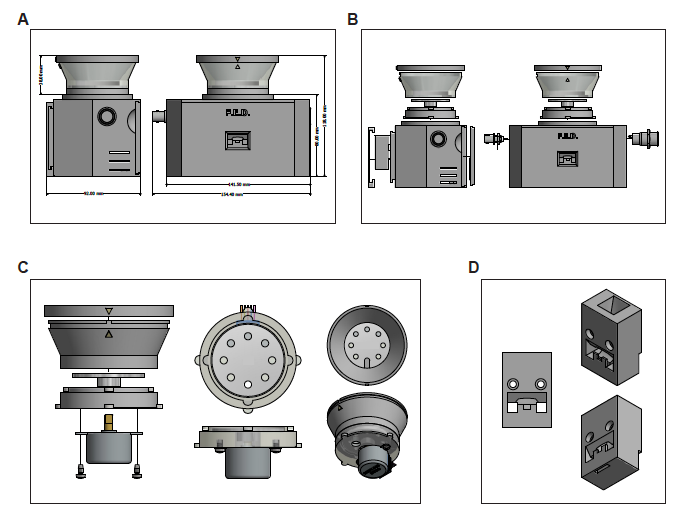
**PROTOCOL:**

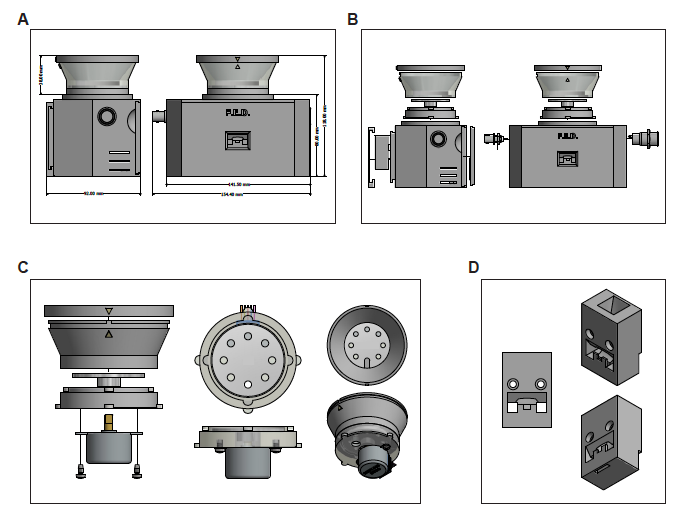
Note: This protocol is written for components specifically named in the Table of Materials. While similar functionality can be achieved using other hardware, FED is programmed with the Arduino Pro microcontroller and listed accessories. Other microcontrollers should work equally well, but will require the user to modify the code to support them. Offline data analysis was coded using the Python programming language.

1. **Preparation and software installation**
   1. Procure electronic components needed to construct FED (see **BoM** at: https://github.com/KravitzLab/FED/tree/master/doc).

Note: Alternative suppliers may be used for many parts on this table, provided they have equivalent specs.

* 1. Print all 3D designed components (available at: <https://github.com/KravitzLab/fed/wiki/Build-Instructions>). 3D printers with a 200 micron resolution should be capable of printing FED.





* 1. Download and install the Integrated Development Environment (IDE) platform to program the microcontroller.
  2. Download and install additional libraries to enable functionality of motor shield and data logger (available at: <https://github.com/KravitzLab/fed/>).
  3. Tools needed for assembly include: a soldering iron, heat gun, solder, wire strippers, needle-nosed pliers, and both flat-head and Phillips screwdrivers.

1. **Soldering electrical components**

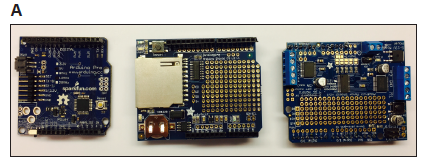
Note: Use heat shrink tubing to protect all soldered joints. Prior to soldering connections, slide a piece of shrink wrap tubing (~2cm) tubing around one of the wires. After soldering the connection, center the tubing on the connection point and use a heat gun to heat shrink the tubing.

* 1. **Preparing connectors**
     1. Prepare four 2-pin JST connector pairs and label both sides (male and female compliments) “A”, “B”, “C”, and “D”, respectively. Remove the red wire from both sides of connector pair “D”.

* + 1. Prepare one 3-pin JST connector pair and label both sides (male and female compliments) “E”.



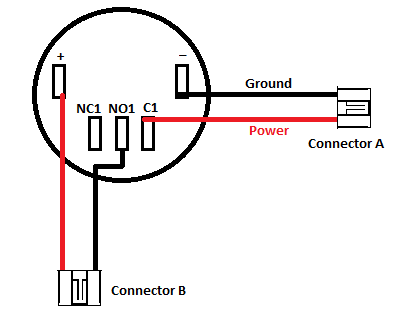
* 1. **Microcontroller and stackable shields:**
     1. Solder female stackable headers with sockets on the top side of the microcontroller. Clip protruding wire from headers on the bottom of the microcontroller.



* + 1. Solder female stackable headers with sockets on the top side of the SD data logging shield. Leave protruding wires at bottom of the shield.
    2. Solder male headers onto the motor shield with pins protruding from the bottom.
    3. Place coin cell battery into slot of SD shield to provide power to the real-time clock module.
    4. Jumper the Vin jumper on the motor shield (just above the power block), using the provided 2 pin jumper.
  1. **External power button:**

Note: A latching metal pushbutton has five connections: power, ground, normally closed (NC1), normally open (NO1), and common (C1).

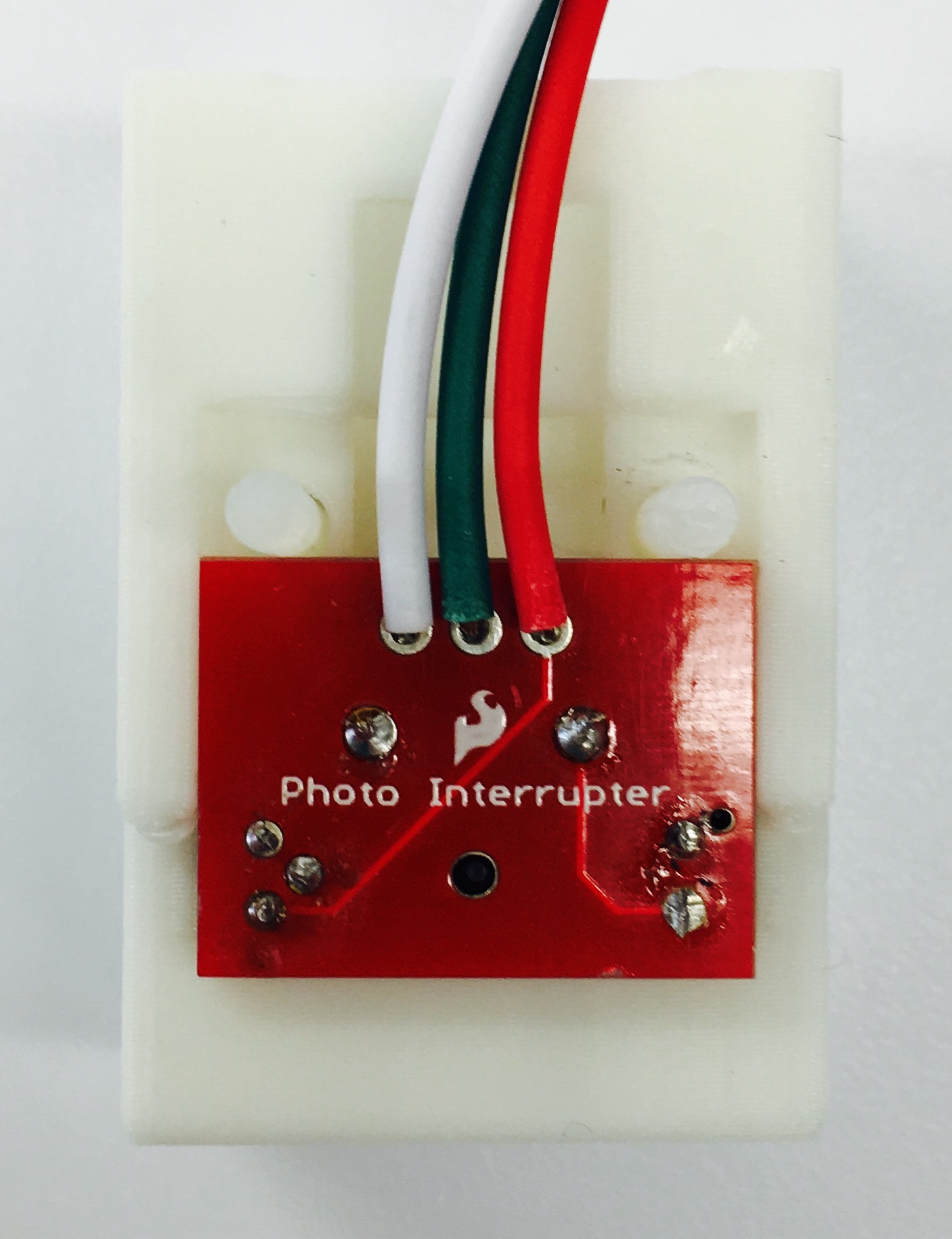
* + 1. Solder the 2-pin male connector “A” to C1 (use red wire) and ground (use black wire). Heat-shrink all connections.
    2. Solder the 2-pin male connector “B” to + (use red wire) and NO1 (use black wire). Heat-shrink all connections.



* 1. **Photo-interrupter :**
     1. Solder photo-interrupter (the black part) to breakout board.
     2. Solder a 4.7K resistor to the front of the breakout board.
     3. Solder the male 3-pin connector “E” to the back of the breakout board: Red wire to PWR, Green wire to GND, and white wire to SGL.

****

* + 1. Trim all loose wires on photo-interrupter break out board.

****

* 1. **Boost board**
     1. Solder the 2-pin female connector “A” to 5V and Ground pins on the boost board.

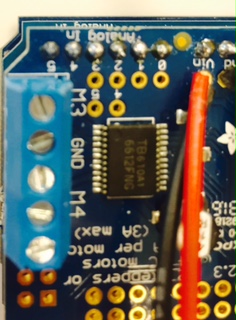
* + 1. Solder the black wire from male connector “D” to the additional GND pin on the boost board.



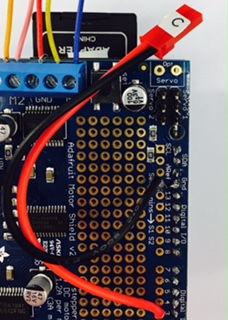
* 1. **BNC output cable (optional):**
     1. Solder the 2-pin connector “C” to the terminals of a BNC cable (red wire to central pin, black wire to outside pin). Note: for assembly, the 2-pin connector must fit through the nut on the BNC plug. We use a smaller connector, or shave down the JST connector with a razor blade to make it fit.



* 1. **Motor Shield**
     1. Twist the red and black wires of the female connector “B” together and solder to Vin.

* + 1. Solder the black wire of the female connector “C” to the ground pin next to ARef, and the red wire of this connector to pin 3.



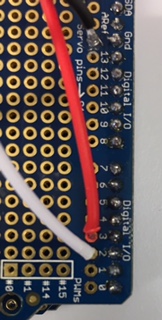
* + 1. Solder the black wire of the female connector “D” to the ground pin next to Vin.
    2. Solder the green wire of the female connector “E” to the ground pin next to 5v, the red wire of this connector to 5v, and the white wire of this connector to pin 2.



Connector E to 5V and GND

Connector D to GND

Connector B to Vin



Connector E to Digital pin 2

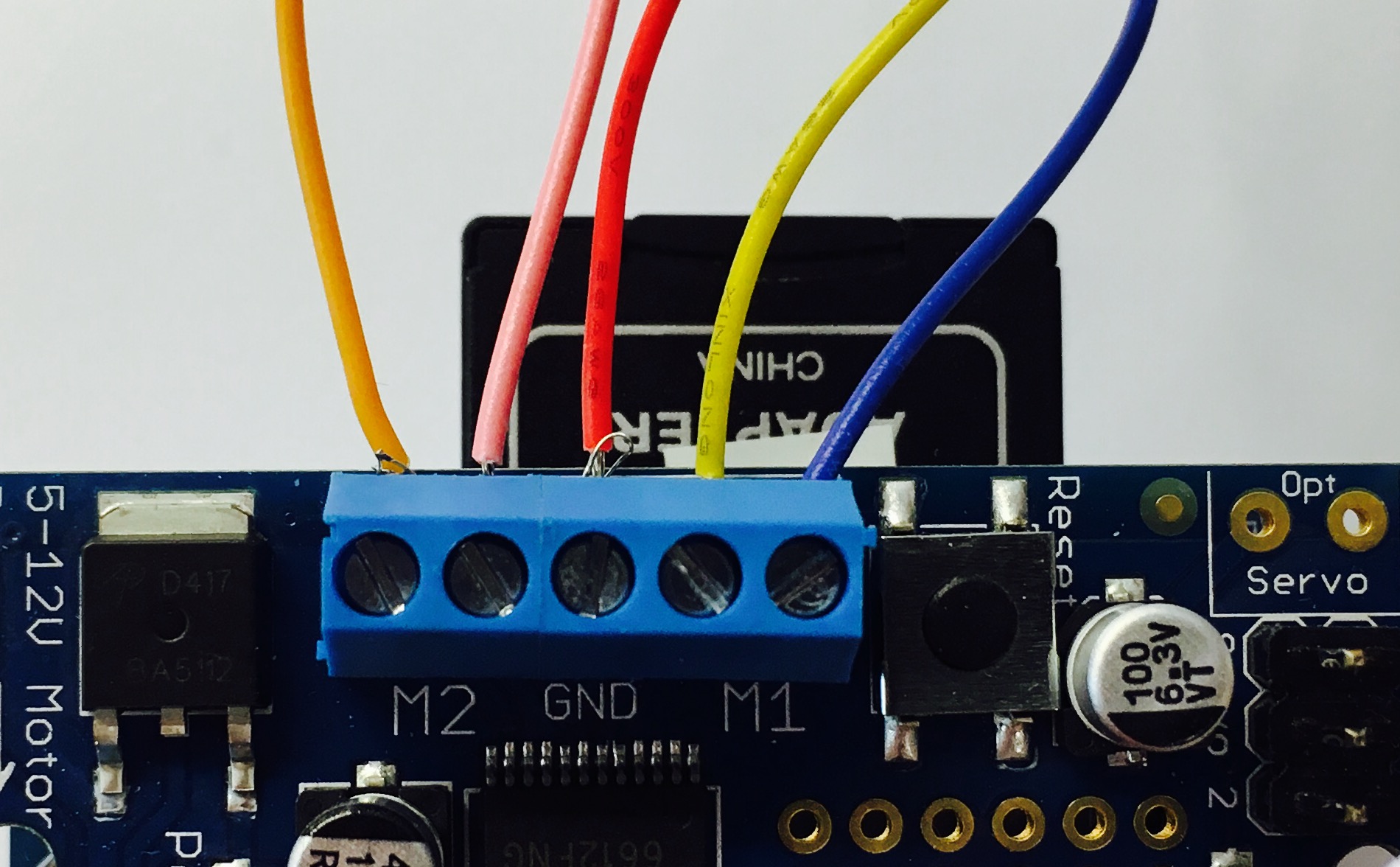
1. **Software upload**
   1. Connect the FTDI breakout board to the programming pins of the Arduino Pro, and then connect FTDI breakout board to computer via micro USB cable.
   2. Open the IDE (integrated development environment) program.
   3. Select Arduino Pro or Pro Mini through Tools > Board dropdown menu.
   4. Select ATMega 328 (5V, 16mHz) through the Tools > Processor menu.
   5. Select the port that the microcontroller is connected to through Tools > Port > COM# (will vary depending on which port is currently in use).
   6. Click the “upload” button to upload the FED sketch to the board (available at: <https://github.com/KravitzLab/fed/tree/master/fed-arduino>).
2. **Hardware assembly**
   1. **Stepper motor and motor shield (Figures 1C and 2E):**
      1. Secure the 5V stepper motor onto the 3D printed motor mount with two #6 x ¼” sheetmetal screws.



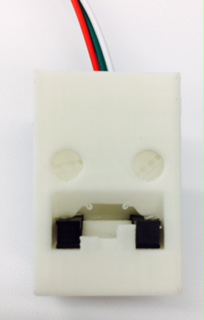
* + 1. Insert rotating disk into motor mount and push down to securely attach to stepper motor shaft.



* + 1. Twist on 3D printed food silo onto the motor mount making sure the pellet leveler arm is over the hole in the motor mount.
    2. Twist on connected pieces from above (steps 4.1.1 – 4.1.3) to the top of the printed base, with the stepper motor positioned towards the back of the base and the hole positioned in the front.
    3. Cut the 5-pin connector from the stepper motor wires and strip ~2mm from the end of each wire.
    4. Connect wires from stepper motor to the terminal block connectors on the motor shield: red to ground, orange and pink to one motor port (e.g., M1), and blue and yellow to the other motor port (e.g., M2).

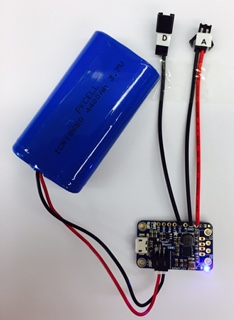


* 1. **External power button:**
     1. Remove the nut from the power button and insert the power button into the hole in the right side of the base. Secure button in place with hex nut.
  2. **Photo-interrupter :**
     1. Place the photo-interrupter into its 3D printed housing. Note: use a heat gun to heat up the housing if the photo-interrupter won’t seat all the way in.

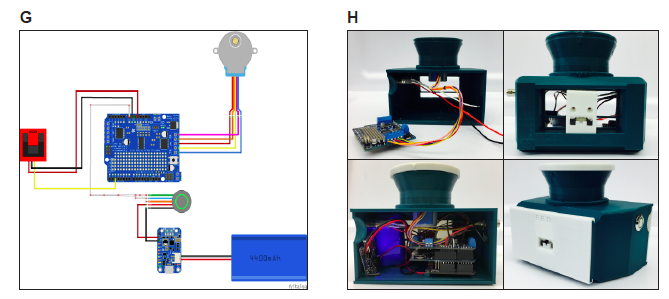


* + 1. String the 3-pin male connector “E” from the photo-interrupter (PWR, GND, and SGL) through the front middle hole of the 3D printed base.
    2. Secure the housing into the FED base with two 1” nylon screws and corresponding nuts.
  1. **BNC output cable (optional):**
     1. Insert BNC connector into hole on the left side of the FED base. Secure in place with nut.

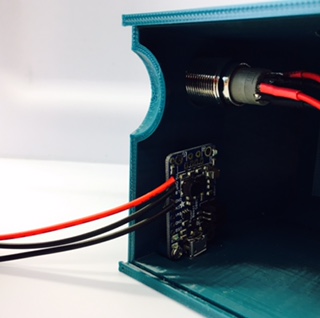
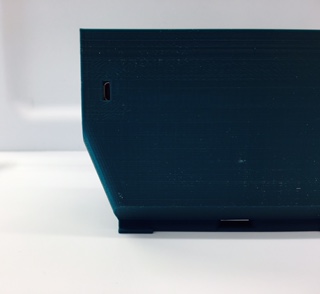
* + 1. If BNC connector is not used, plug hole with the 3D printed plug.
  1. **Battery and boost board:**
     1. Connect 3.7V battery pack to the DC/DC boost converter module via the JST 2-pin connection.



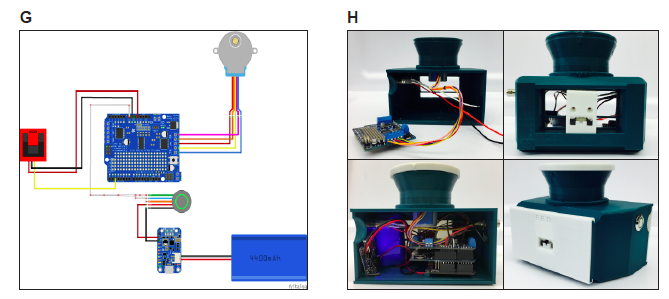
* + 1. Mount Arduino Pro inside of the base with FTDI connections facing the power switch, using #4 x ¼” steel sheet metal screws.
    2. Stack motor shield and data logging shield on top of the Pro.



* + 1. Screw the Boost board into the case using #2 x ¼” steel sheet metal screws. Mount Boost with the micro-SD slot pointing down – FED can be charged through this port without opening the case.

* + 1. Connect the five connectors, “A” male to “A” female, “B” male to “B” female, etc.
    2. Place the battery inside the 3D printed base and slide the back cover closed (**Figure 2G-H**).



* + 1. Slide on the 3D printed face plate.
    2. Fill food silo with 20mg food pellets

1. **Validation and data acquisition**

Note: Prior to powering on a FED system, ensure an SD card is inserted on the SD shield; otherwise FED will not dispense pellets. Additionally, ensure power jumper on the motor shield (just above the power block) is in place.

* 1. Power on FED system with the power pushbutton and test device functionality:
     1. Power switch should light up, as should LEDs on the microcontroller, SD shield, and motor shield. If there is no pellet in the well, one should dispense.
     2. Manually remove 5-10 pellets from food well and confirm that replacement pellets are dispensed.
  2. Remove SD card and verify that data was logged properly. Data should be acquired in a comma-separated value (.CSV) file named according to the date and start time of each device.
  3. Place FED unit inside experimental setting and power on. Ensure that a pellet is dispensed into the food well.
  4. Over the course of data acquisition, check FED daily to verify that it is working properly by confirming that (1) the LED light on the power switch is on, indicating that the battery has enough charge, and (2) a pellet is sitting in the food well, indicating that there are no problems with pellet dispensing.
  5. After data acquisition, retrieve SD card and access .CSV file
  6. We provide analysis scripts for meals and patterns of feeding at: <https://github.com/KravitzLab/fed/>.