Automated Home-cage System User Manual

The Automated Home-cage System is a fully autonomous mouse behavioral and optogenetic experiment system. This manual will guide you through the initial check and installation of your system. It also describes operation, maintenance, and diagnostic procedures of your system.

1. Hardware Checklist

- The Automated Home-cage System is a fully enclosed system as shown in Fig.1.1.

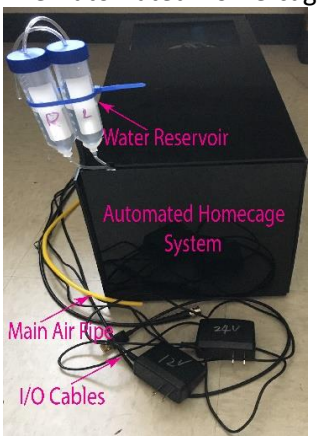

Fig.1.1

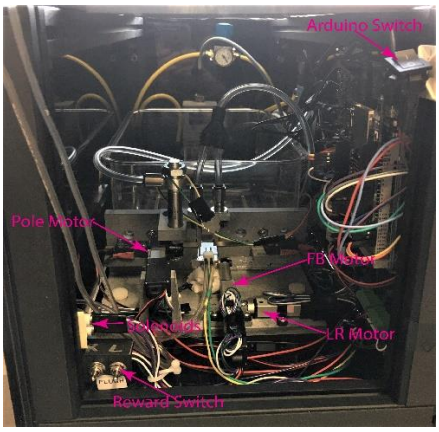

Fig.1.2

- There are four electric cables and one plastic tube come out from the left side of the enclosure. Refer to table1 for their specifications.

Table1. Home-cage I/O

	Item Name	Function
Input	12v AC/DC wall mount Adapter	Provide 12v power for Arduinos, motors and Lee solenoids
	24v AC/DC wall mount Adapter	Provide 24v power for 3/2-way solenoid valve and analog pressure regulator
	USB 2.0 to Micro B Cable	Connect Arduino_Master to PC
	Plastic tube	Provide Air supply (4-bar) to Home-cage
Output	BNC Cable	Send voltages (0-5 V) to control Laser

- The water reservoir, the Arduino switch, the reward switches, and the solenoids are located at the front frame of the enclosure (Fig.1.2).
- All the circuit boards are mounted on the right side of the enclosure (Fig.1.3).

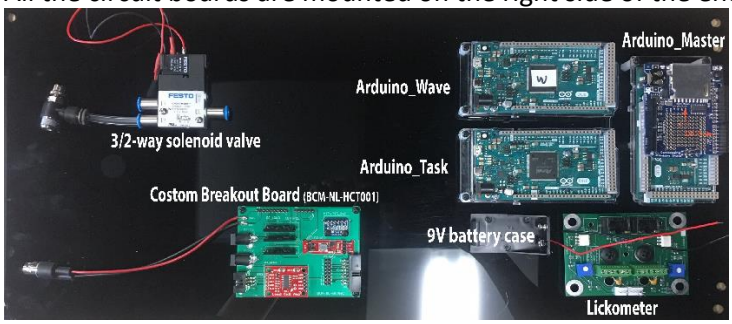

Fig.1.3

- The Home-cage and the Headport are fixed on the floor of the enclosure (Fig.1.4).

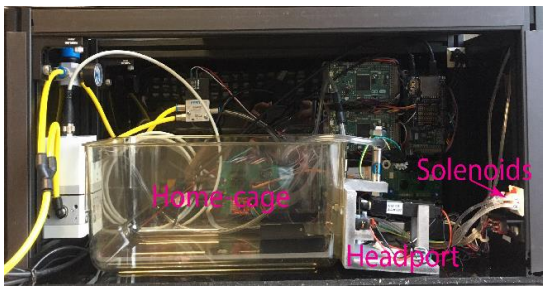

Fig.1.4

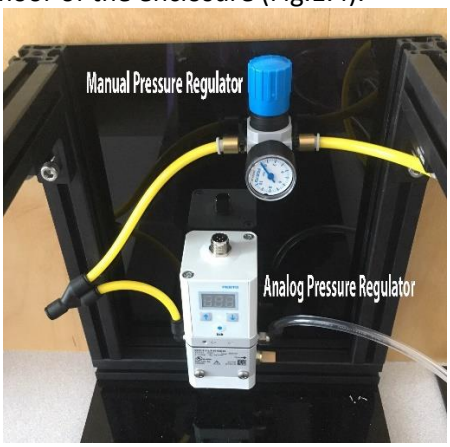

Fig.1.5

- Two pressure regulators are mounted on the back side of the enclosure (Fig.1.5).

2. Software installation

- Download the latest version of **MATLAB** from https://www.mathworks.com. After installed it, do the following steps to adjust **Java Heap Memory** to be half of the full Java Heap Size:
  - Start MATLAB.
  - On the **Home** tab, in the **Environment** section, click **Preferences**. Select **MATLAB** > **General** > **Java Heap Memory**.
  - Select a Java heap size value using the slider or spin box.
  - Click **OK**.
  - Restart MATLAB.
- Download the latest version of the **Arduino IDE** from https://www.arduino.cc. After installed it, with the Arduino plugged into a USB port, start up the IDE and make sure that the correct board name/version and COM port appears under the **Tools** menu.
  - The COM port can be seen in the Windows Device Manager.
  - A support package for Arduino DUE needs to be downloaded via the IDE:
    - Click the menu item **Tools → Board → Boards Manager**.
    - Search 'due' and find Arduino Due package.
    - Click on the **Install** button that appears.
  - Copy all the files in the folder "\Arduino programs\XXX task\Library\" to the PC folder:\Program Files (x86)\Arduino\libraries\".
  - To upload an Arduino program to Arduino board via the IDE, do the following steps:
    - Connect a USB cable from PC to the Native USB Port of the Arduino board
    - Click the menu item **File → Open**, select the program.
    - Click the menu item **Tools → Board,** select the Native USB port.
    - Click the menu item **Tools → Port,** select the correct COM port **.**
    - Click the button **Upload .**

- Upload Arduino program to Arduino board according to the task. In the folder ***"\Arduino programs\XXX task\"***:
  - Upload ***"\Arduino_Wave\Arduino_Wave.ino"*** to the Arduino_Wave board.
  - Upload ***"\Arduino_Task\Arduino_Task.ino"*** to the Arduino_Task board.
  - Upload ***"\ Arduino_ Master \Arduino_Master.ino***" to the Arduino_ Master board.

3. HomeCage GUI

HomeCage GUI is created for user to easily interface with the Arduino_Master board during experiment. To use HomeCage GUI, copy folder "\Matlab GUI\" to the PC, connect the system to the PC and run ***"\Matlab GUI\homecage_gui.m"*** program in MATLAB. HomeCage GUI shows up as Fig.3.1. The main functions are described as follows:

3.1. Cage block

- Set the mouse name, the experiment starting date and the COM port for the corresponding cage. Click **open** button to communicate to Arduino_Master board.
- Click **Msg** to get the experiment information (message) in last 24 hours.
- Click **Plot_ P/W** to plot the mouse behavior data in last 24 hours.
- The last 24 hours data/message stored in the local PC will be clear once click **close** button, disconnect the USB cable from PC, or Arduino_Master board is reset.

3.2. Control panel

- Choose the cage that you want to interface with.
- Click **read** to get the on-line motor positions and mouse weight and show them in the corresponding text fields.
- Set a value in text field of **MotorFB**, click **Move& Set** to move the FB motor and set its position value in the Arduino code.
- Set a value in text field of **MotorLR**, click **Move& Set** to move the LR motor and set its position value in the Arduino code.
- Set a value in text field of **MotorPole**, click **Move** to move the Pole motor.
- Set a value in text field of **FinalFB**, click **Set** to set the final position value of the FB motor in the Arduino code.
- Set a value in text field of **Left/Right**, click **Set&Reward to** set the reward size and a drop of water comes out of lickport.
- Click **Tare** to initialize the weighting scale and save the offset of the weighting scale to the SD card at the Arduino_Master board.
- Set a value in text fields of **Struggle Threshold**, click **Set** to set the range of the mouse struggle threshold in the Arduino code.
- Set a value in text field of **Pole Anterior/Posterior**, click **Set** to set the anterior and posterior position value of the Pole motor in the Arduino code.

3.3. Save Paras and Load Paras Button

- Click **Save Paras** or **Load Paras** to save or load the settings of the mouse name, the experiment starting date and the COM port number for all the cages.

Fig.3.1

4. Operation Procedures

1. Move the Automated Home-cage system to the workstation where there are the compressed air supply and the power outlet accessible.
2. Connect the plastic tube from the system to the compressed air supply (4 bars).
3. Adjust the manual pressure regulator to 2 bars and check if there is air leaking.
4. Make sure the Arduino switch is switched to “OFF”.
5. Copy the two initial parameter files (***PARAF.TXT*** and ***PARAS.TXT***) in the folder ***“\Arduino programs\XXX task\initial config files\”*** to a blank SD card and insert the SD Card into the SD card shield.
6. Adjust the height of the lickport to make sure it is higher than the bottom edge of the **headport** opening.
7. Power up the 12V and 24V adapter and then connect Arduino_Master board to USB hub via USB cable.
8. The lickport will be moved into the headport opening.
9. Run “***\Matlab GUI\homecage_gui.m***” program in MATLAB. Set the mouse name, start date and the com port in the corresponding cage block of the HomeCage GUI. Click **open** button to start receiving data from Arduino_Master board.
10. Make sure nothing is on the weighting stage and click **Tare** button to zero the weighting scale.
11. Click **Read** button, the text field of **Weight** should show a value close to zero. Put an object with known weight **W** (g) on the weighting stage and click **Read** button again, the text field of **Weight** should show a value close to **W** (g).

12. After toggled the Arduino switch to "ON", trial 1 starts and the pole drops down. Touching the lickport by your hand will trigger another trial.
13. Put bedding, food, hiding tube into the home-cage.
14. Put the mouse into the home-cage.
15. Cover the home-cage with the lid and the system enclosure with the top panel.
16. Repeat the above steps if you have more cages. Click **Save Paras** button to save the cage setting parameters once you finished setting up the cages.

5. Maintenance

5.1. Connect to PC

- if you want to use HomeCage GUI to interface with Arduino_Master board and check mouse behavior while experiment is running, the Automated Home-cage System needs to connect to a PC, even though it can be fully function without connecting to a PC.
- In the event that MATLAB crushes while experiment is running, close MATLAB, and make sure that the Automated Home-cage Systems are still running fine. Restart MATLAB and run HomeCage GUI again.

5.2. SD Card

- For a new mouse, copy the two initial parameter files (***PARAF.TXT*** and ***PARAS.TXT***) in the folder ***"\Arduino programs\XXX task\initial config files\"*** to a blank SD card and start the experiment. While the trials are going on, there are six files stored in the SD Card. Refer to "\example data\README.txt" for the detail explanations of these files.
- When putting a different mouse to the same cage, make sure to change the SD card for the particular mouse.
- When changing SD Card, i.e., pulling one out from and pushing the other one into the SD Card shield, be sure to turn off the Arduino switch.

5.3. Weighting Scale Initialization

- When click **Tare** button**,** the program performs the weighting scale initialization and saves the offset of the weighting scale to the file ***PARAF.TXT*** in SD card. You must perform **Tare** in the following circumstances:
    - Using a new cage.
    - Using a SD card which is new to the particular cage.
    - After replacing the load cell for the weighting scale.

5.4. Weighting Scale Routine Check

- In order to get the mouse weight accurately while the experiment is going on, it is important to check the weighting scale routinely via HomeCage GUI**:**
    - Make sure nothing is on the weighting stage and click **Tare** button to zero the weighting scale.
    - Click **Read** button, the text field of **Weight** should show a value close to zero.
    - Put an object with known weight **W** (g) on the weighting stage and click **Read** button again, the text field of **Weight** should show a value close to **W** (g).

5.5. Reward Solenoid Routine Check

- Use **Set&Reward** button in Homecage GUI to set the reward size for both left and right lickport. Make sure the actual reward drops are roughly equal on both sides.

- Check the reward size routinely. If noticing a much smaller reward size while the experiment is going, flush the water line first to see if the reward size is back to normal. If not, replace with a new solenoid.

5.6. Pole Motor

- Too much pole activity can cause the motor malfunction. A warning message appears in **Warning** box of HomeCage GUI and indicates which cage's pole motor is malfunction when the malfunction happened.
- In the case the warning message appears, be sure to verify if the pole motor is malfunction. Set a value in text field of **MotorPole** and click **Move** to move the Pole motor. If the pole motor does not move, carry on the following steps:
    - Turn off the Arduino switch to pause the experiment.
    - Refer to section 5.5 of "Automated Home-cage System construction Manual" and replace the pole motor.
    - Open "PARAS.TXT" file in the SD card, change parameter "is_motor_ok" from 0 to 1 and save.
    - Insert the SD card back to the SD card shield.
    - Turn on the Arduino switch to continue the experiment.

5.7. Arduino Board Reset

- In general, there is no need to reset the Arduino boards manually, i.e., by pushing the reset button at the Arduino board. When power up, the three of Arduino boards reset and handshake with each other automatically.
- In the event that Arduino_Master board hangs while experiment is running, the embedded watchdog timer will reset Arduino_Master board automatically, and the experiment should continue without issues.
- In case you have to reset the Arduino boards manually, be sure to reset Arduino_Wave board and Arduino_Task board first, and then reset Arduino_Master board.

5.8. Lickport Position

During Auto-training, check and adjust the Lickport position periodically:

- When the lickport retracts out of the headport, adjust its height to a bit higher than the bottom edge of headport opening (Fig.5.1) and make sure it is still at the center of the head port opening in left-right direction.
- When lickport reaches the final position, there is a one-time adjustment of the position to make sure the mouse can access both tubes of the lickport and the lickport is in the center roughly (Fig.5.2). The final FB motor position can be adjusted via the HomeCage GUI as described in section 3.

Fig.5.1

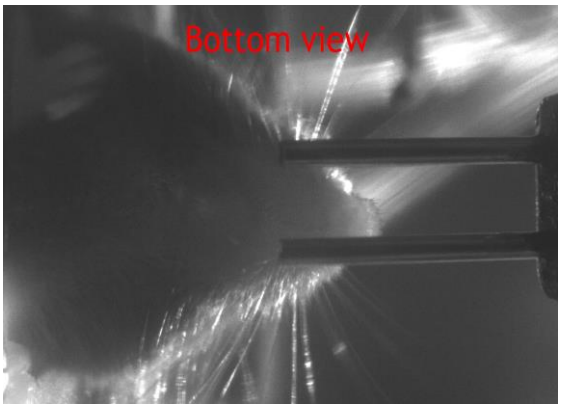

Fig.5.2

5.9. Clean Home-cage

The home-cage can be detached from the system to be cleaned:

- Unscrew the two screws (inside home-cage) that fixes the home-cage with the headport.
- Detach the 6pin connector that connects weighting stage to the circuit board on the right side of the enclosure.
- When clean the home-cage, be sure to take a good care of the weighting stage (no hard press to the stage), the 6pin connector and the ribbon cable.

5.10. Power on/off

- In general, the system can be power on all the time.
- In the event you want to power off the system, follow the sequence: turn off the Arduino Switch**,** unplug 12V adapter and 24V adapter, unplug USB cable from PC, turn off the air supply.
- In the event you want to power on the system, follow the sequence: turn on the air supply, plug in 12V adapter and 24V adapter, plug in USB cable from PC, turn on the Arduino Switch**.**

5.11. Add/Modify protocol

To modify an existing protocol or add a new protocol, you should modify the existing states or create new states by editing the function “send_protocol_to_Bpod_and_Run()” in “\Arduino programs\XXX task\Arduino_Master\Arduino_Master.ino”.

- In function “send_protocol_to_Bpod_and_Run()”, modify/add new state using function “CreateState()”.
- The function ”CreateState()” constructs a state struct defined by the input arguments and returns a constructed state. There are 5 input arguments:
  - Name: The name of the state.
  - TimeOut: A float variable indicating the state timeout.
  - nChangeConditions: The number of state transitions defined.
  - StateChangeConditions: The pointer to the state transitions.
  - nOutputActions: The number of output actions.
  - Outputs: The pointer to the output actions.

6. Trouble Shooting

6.1. Mice do not enter trigger head fixation for extended period of time

6.1.1. Mouse cannot poke in the headport (Fig.6.1)

- Check the headbar position: anterior edge of headbar should line up with lambda suture; depressions face up.
- NO glue or dental cement should occlude either side of the headbar.
- Headbar position should be consistent from mouse to mouse.

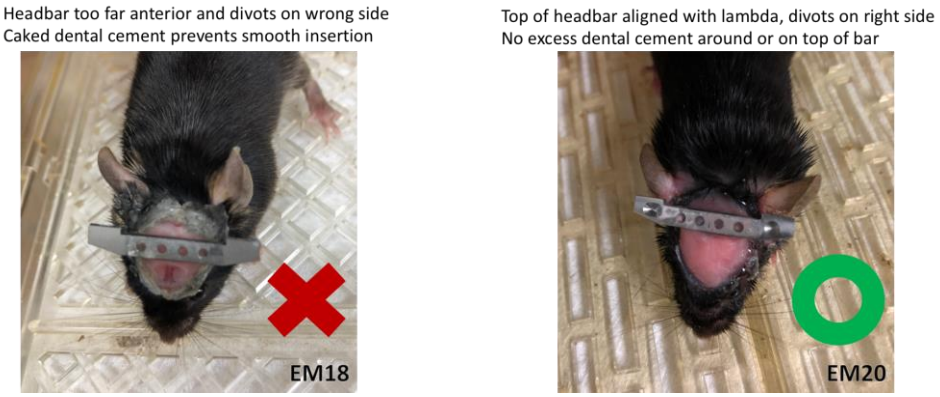

Fig.6.1.

6.1.2. Mouse cannot access the lickport because the lickport is far from mouse

- The lickport position needs to be adjusted, refer to section 5.8 for how/when to adjust the lickport position.

6.1.3. The program cannot detect licking when mouse licks

- Check if metal sheet on the weighting stage is electrically connected with the Lickometer to form a complete circuit. The resistance between metal sheet on the weighting stage and the aluminum tape on headport should be less 10 Ohm if the connection is good.
- Check if the **9v battery** is good and the voltage is more than 8 Volt.
- The Lickometer may be malfunction, refer to section 9.9 of "Automated Home-cage System construction Manual" for how to check Lickometer.

6.2. Mice do not advance in training for extended period of time

6.2.1. The pole does not drop down during the task

- The pole may be stuck. Increase the air pressure to 3-4 bar to let the pole drop down then go back to 2 bars as normal.
- Try manually press down the pole (together with the motor).

6.2.2. Reward could be running slow

- Check periodically that water is coming out and the flow on the two lickspouts is evenly.
- Check if there is any clog in tubes and solenoid. Clean the water lines if necessary.
- Check if the solenoid is malfunctioned.

6.2.3. The mouse triggers self-release too frequently.

- Make sure the mouse is not on the platform, and then click the **Tare** button in HomeCage GUI**.**
- Check if the top panel of the weighting stage is block by bedding or other stuff.

6.2.4. The pole motor is not firmly fixed at the end of the piston.

- Check if the screws are loosed or the 3D printed bracket is broken due to too much piston activity. This should be checked periodically.

7. Training data Example

- Fig.7 shows example learning curve of number of switch trigger (A), number of head-fixation (B), number of reward (C), weight (D) and histogram of number of re-extend of lickport (E). These data are from ~35 example mice and only first 10 days are showed to illustrate the trend.

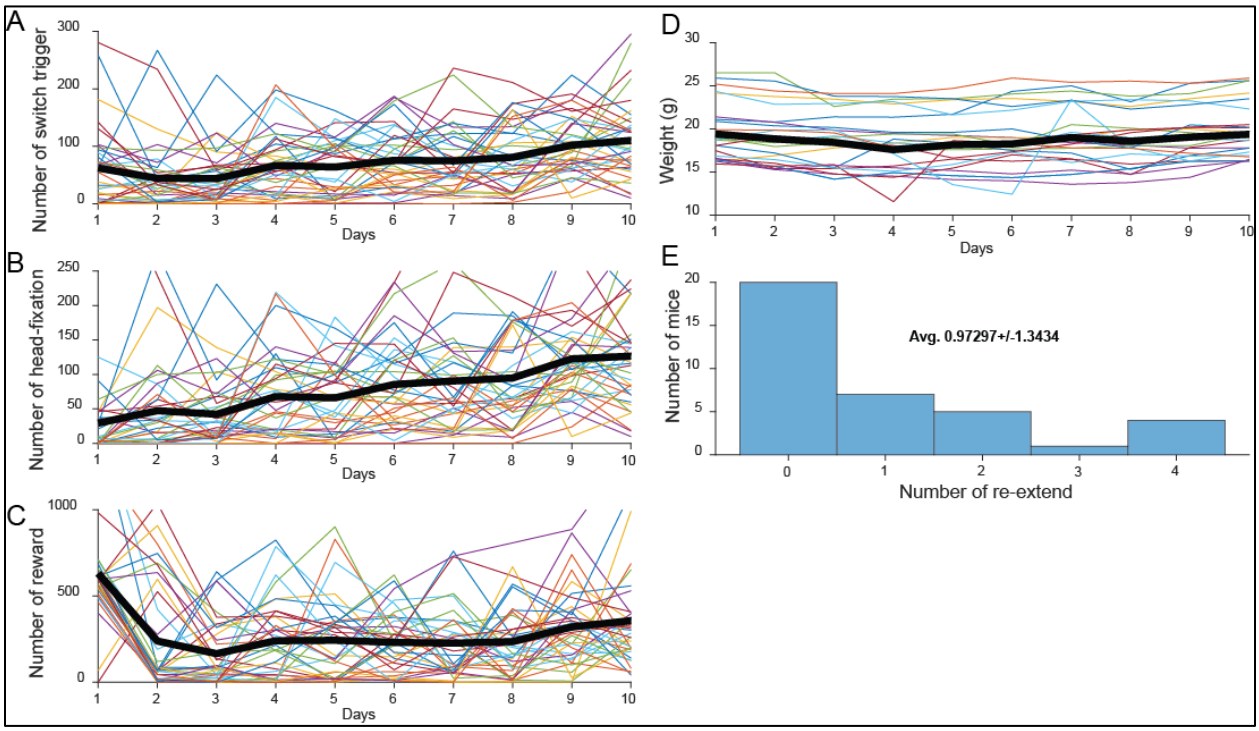

Fig.7