

SwitchDoc Labs

MouseAir3 Assembly and Testing Guide

October 30, 2019

Version 1.1

Table of Contents

Errata	1
What is MouseAir?	2
MouseAir Block Diagrams	3
What is in the MouseAir Kit?	5
What Else is Required for the SolarMax LiPo Kit?	5
MouseAir Annotated Pictures	6
MouseAir Part Identification	9
Preparing and Learning your Raspberry Pi	17
Step by Step Assembly	19
MouseAir Grove Wiring	33
Testing Your MouseAir System	37
Installing Your Blynk App	41
What is Blynk	41
Step by Step Install (for iPhones – Android is very similar)	42
Installing the Blynk Authentication Codes into MouseAir	48
What is in the SkyWeather Blynk App?	49
General MouseAir Operation	52
Preparing your Pi	52
Starting MouseAir	53
Training the AI in Mouse Air	54
Understanding Machine Learning in Python	54
Doing your Own Training	54
The Science and Education Goals Behind MouseAir	54
The MouseAir AI Machine Learning Program	55
Support	56
Disclaimer	57

Errata

What is MouseAir?

This is a great introductory project for learning how to make the Raspberry Pi do physical tasks. Easy to build (no soldering needed) and easy to use. But you will get exposed to some very sophisticated hardware and software technology (including AI machine learning for Cat Detection, no less).

This project has it all. There have been few projects that make us laugh out loud every time we look at it, but this is one of those. MouseAir. This is an easy to build (no soldering!) project that will turn your Raspberry Pi into one of the funniest, smartest and cat like projects of all time.

MouseAir uses an advanced AI (Artificial Intelligence) machine learning algorithm to detect when a cat walks into the view of the supplied camera. When it spots a cat, MouseAir jumps into action. It starts the motors spinning and then activates a servo motor to push the mouse up into the motors and then be launched down the hall for the cat (note: not chickens) to chase. All kidding aside, this is actually a very sophisticated kit showing aspects of AI, robotics and Python programming.

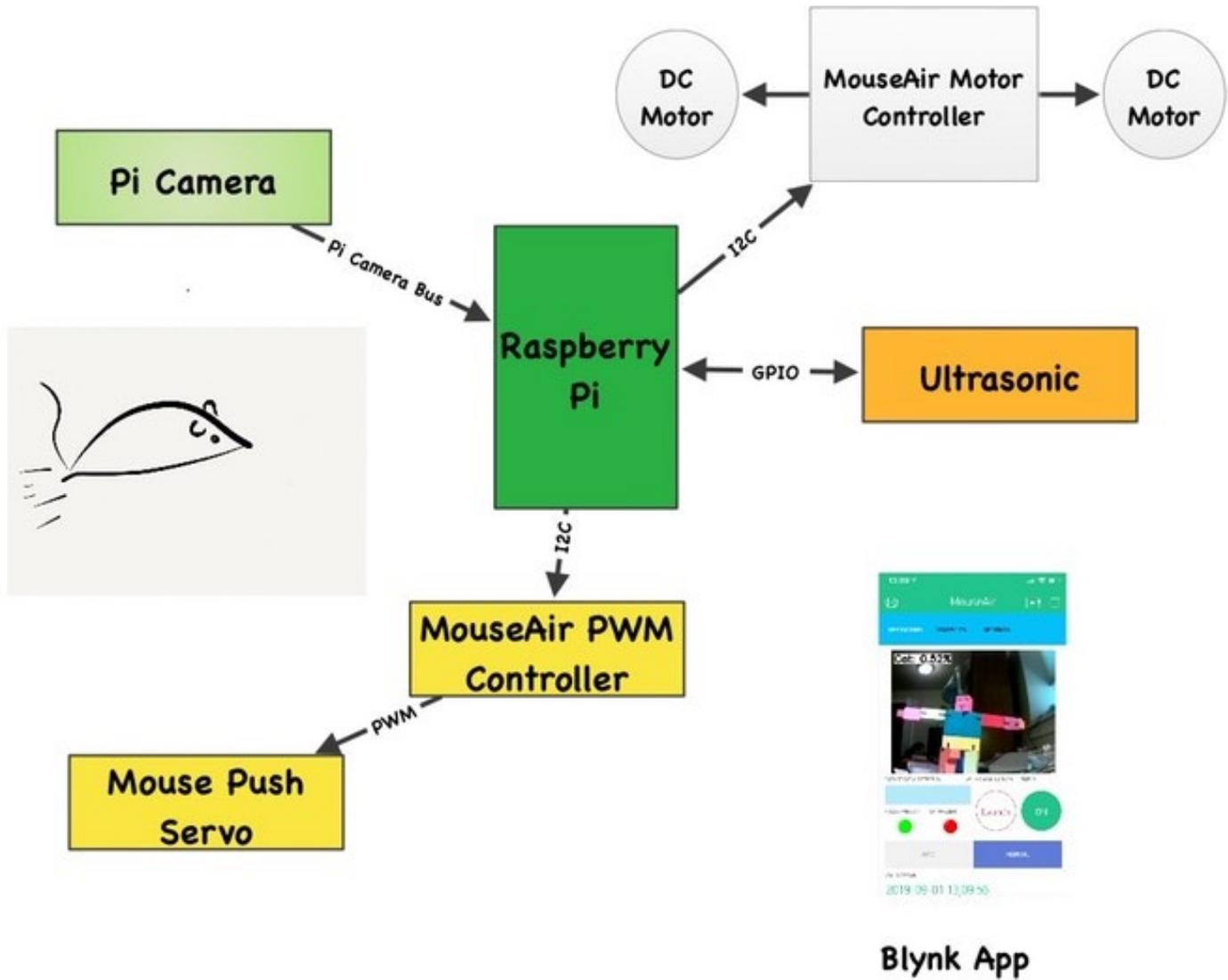
MouseAir is a unique Raspberry Pi based kit that you put together. We give you all the electronics and motors to build it. Have a 3D Printer? Print your own 3D parts. All STL files are included. If you don't a 3D printer you can buy them from many 3D vendors or even us. Don't like the way the program works? Change it. We supply all the software open source. Open source Python3. Even the Artificial Intelligence.

This is a kit built for your to modify and learn from. You build it. You test it. You modify it.

The MouseAir system is built around a Raspberry Pi controlling all the devices and connected to an app based Blynk control panel and also a simple webpage via a WiFi connection. We are using a Pi Camera capture the cat events, examine the video frames looking for a cat with a machine learning AI algorithm.

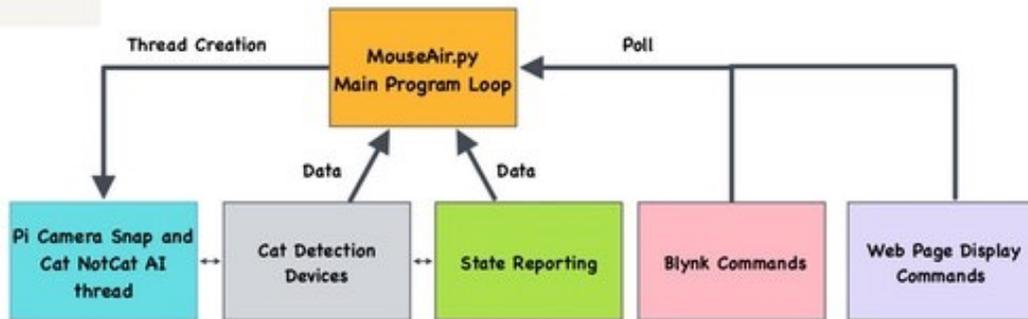
MouseAir Block Diagrams

MouseAir System Diagram





Overall MouseAir Software Architecture



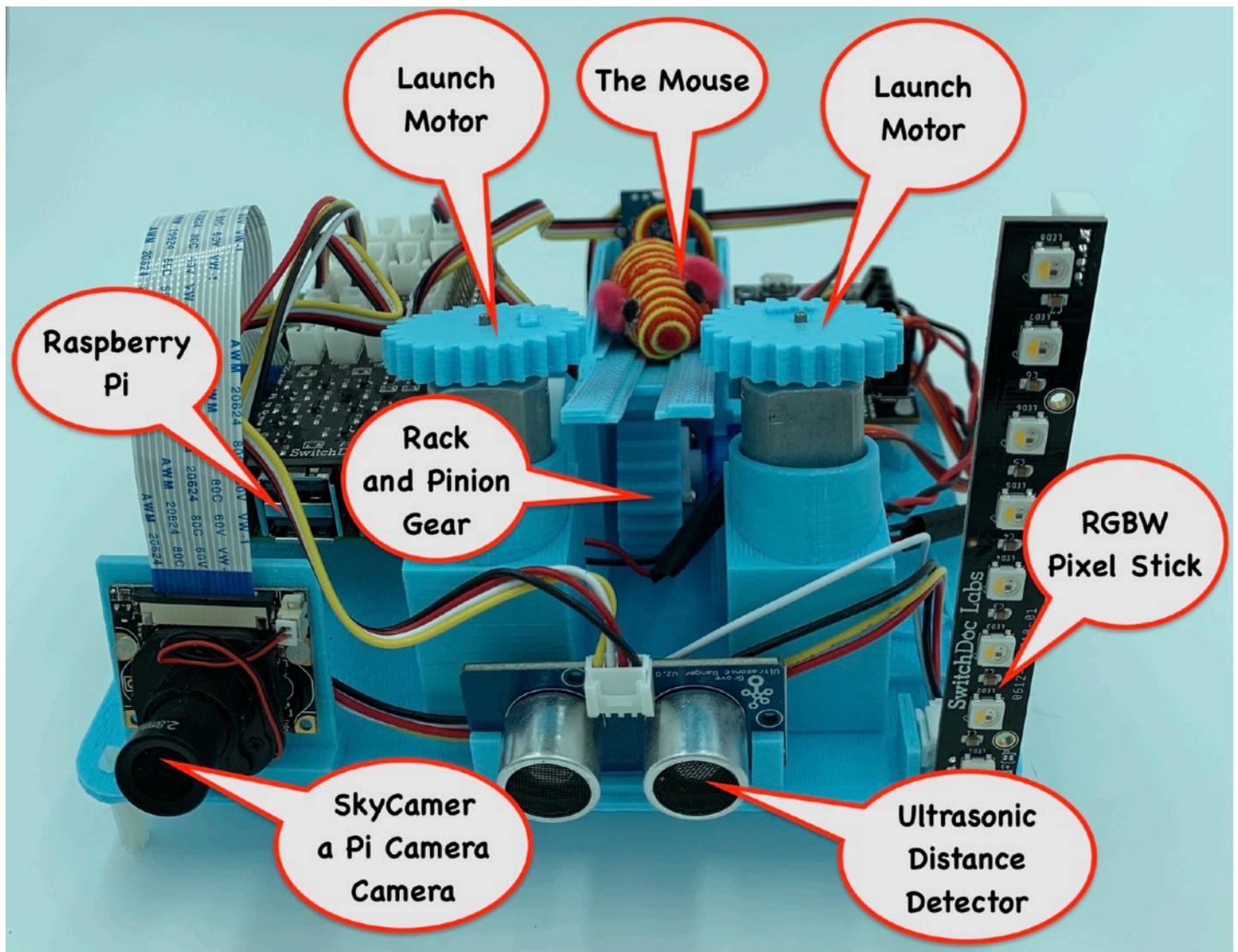
Blynk App

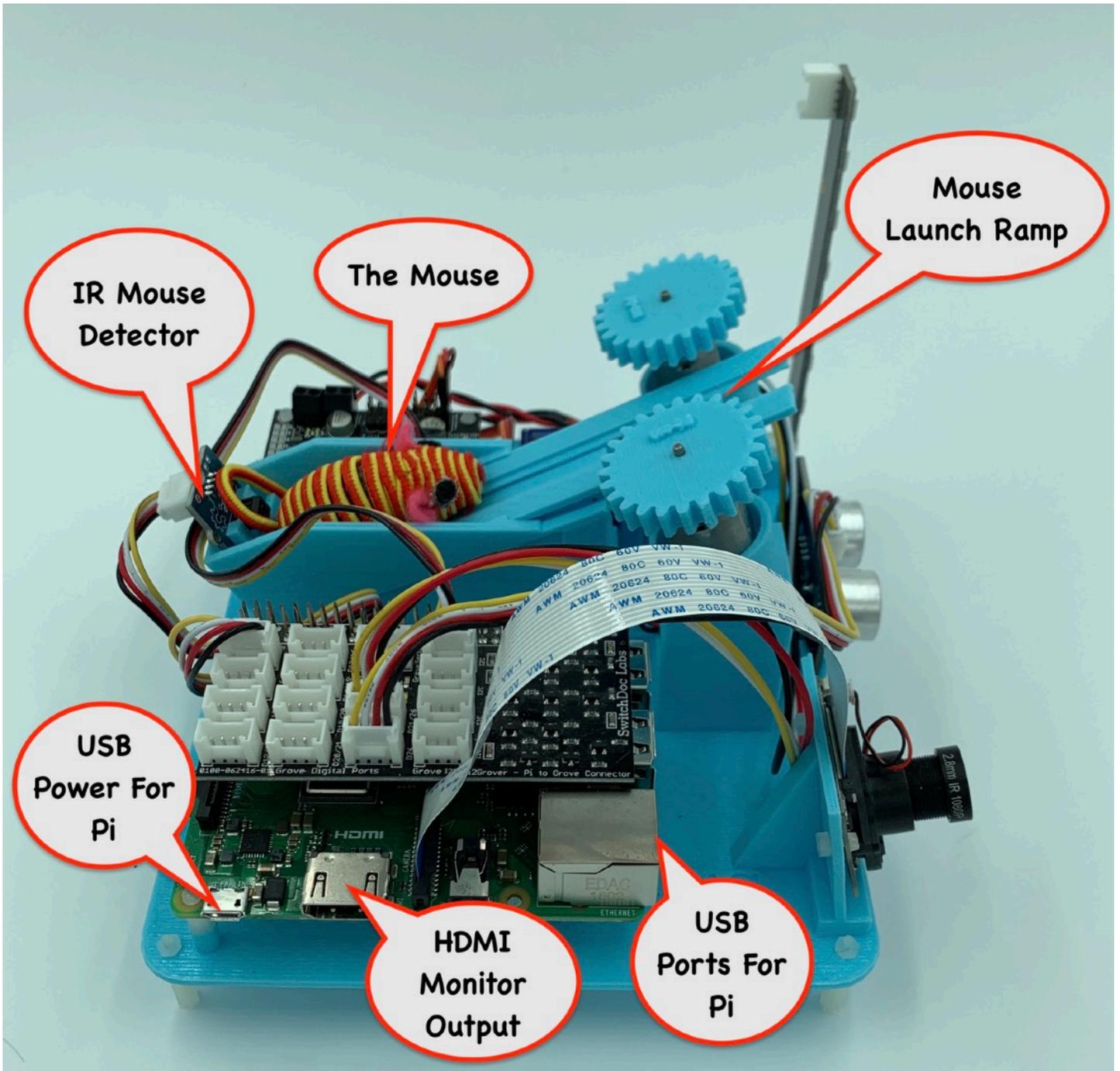


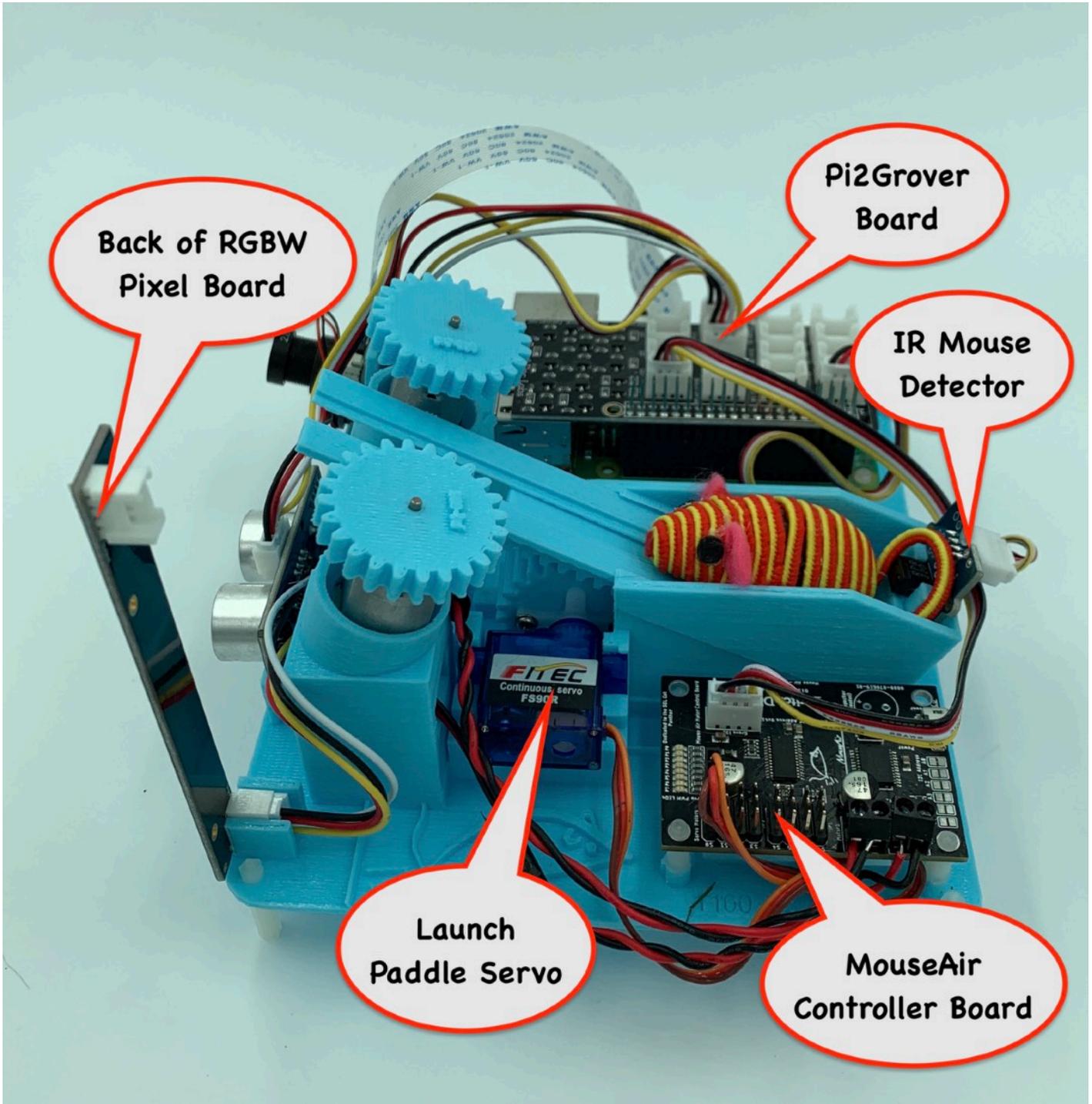
What is in the MouseAir Kit?

What Else is Required for the SolarMax LiPo Kit?

MouseAir Annotated Pictures







MouseAir Part Identification

Part A - Raspberry Pi 3B+ 0680-RP3B+-DSBT

USER SUPPLIED

Part B -SD Card 0675-RPSDCARD-DSBT



Part C - Pi 5V supply 5V 2.5A Power Supply

USER SUPPLIED

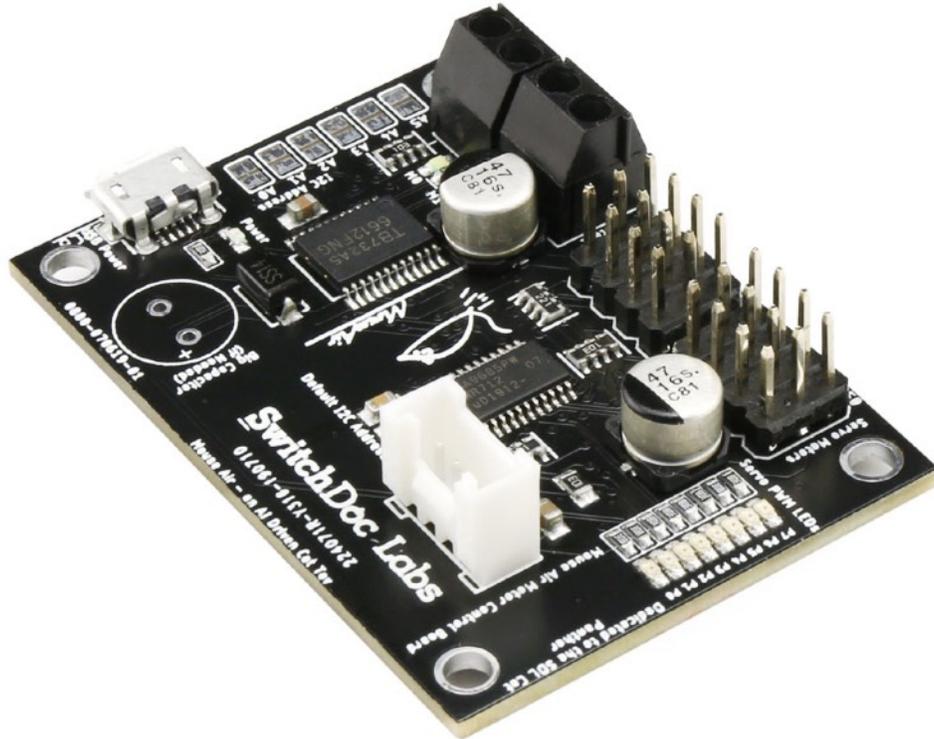
Part D - Camera 0800-CAMIRCUT-DSBT



Part E - Ultrasonic Ranger 0650-GRVULTRA-DSBT



Part G - MouseAir Controller Board 0880-MAIRV3-DSBT



Part H - RGBW Pixel Stick 0612-GRV8PIX-DSBT



Part I - Launch Motor #1 1101-5VDCMOTOR-DSBT (same as Part J)



Part J - Launch Motor #2 1101-5VDCMOTOR-DSBT (same as Part I)



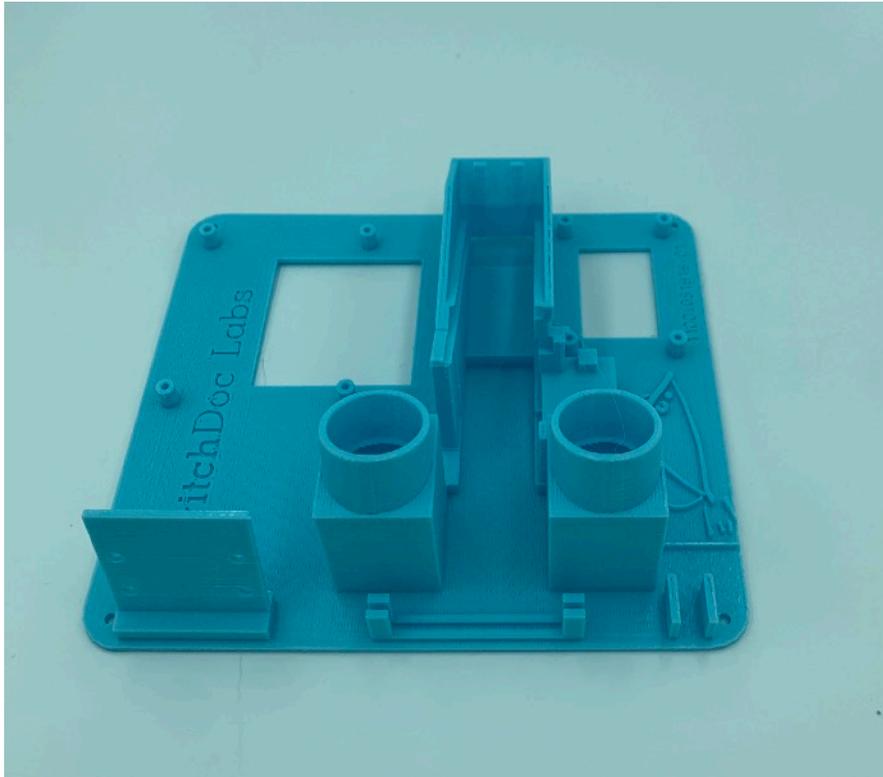
Part K - Launch Servo FS90R



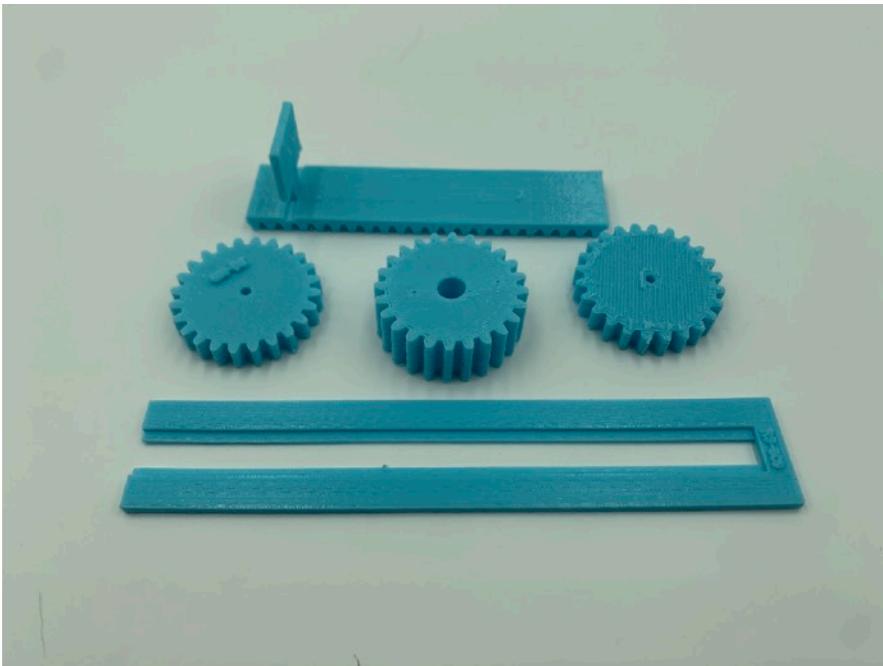
Part L - Grove IR Detector 0130-IRDETCGRV-DSBT



Part M - Base 3D Print 1100-MA3BASE-DSBT



Part N - Misc Packet 3D Prints 1102-M3DMA-DSBT



Part O - 4 20 cm Grove Cables 0075-GRV20C-DSBT



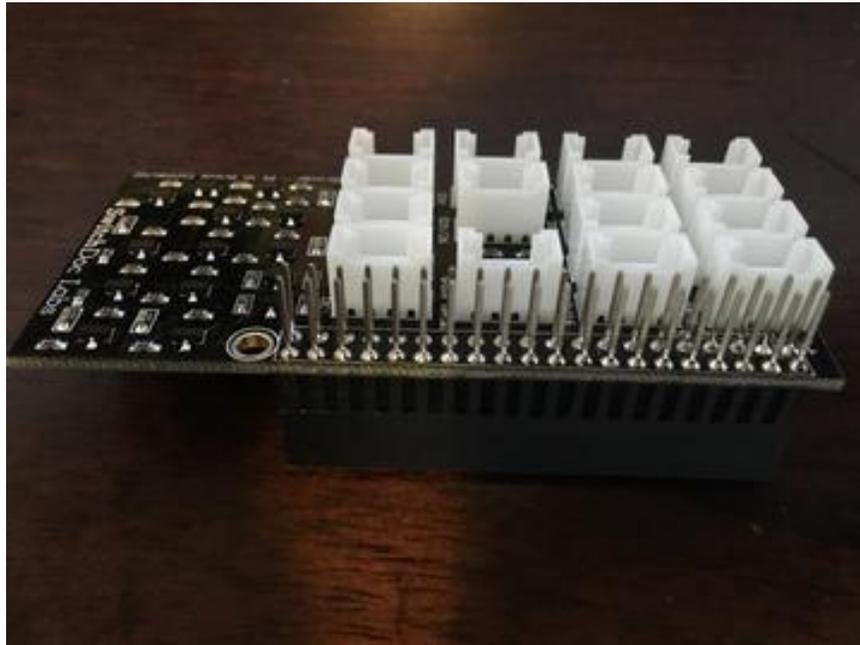
Part P -Box of M2 Nylon Parts 0671-M2NYLON-DSBT



Part Q - 120mm Raspberry Pi Camera Cable



Part R - Pi2Grover Board



Part S - Single Mouse



Preparing and Learning your Raspberry Pi

The MouseAir system requires a working Raspberry Pi. You can use virtually any not too old Raspberry Pi (3B, Zero, etc.) but you do need to set it up before starting the process of building MouseAir

Initial setting up your Raspberry Pi and connecting to it on your network is well beyond the scope of this manual. There are just too many variables in how you might set up your Raspberry Pi.

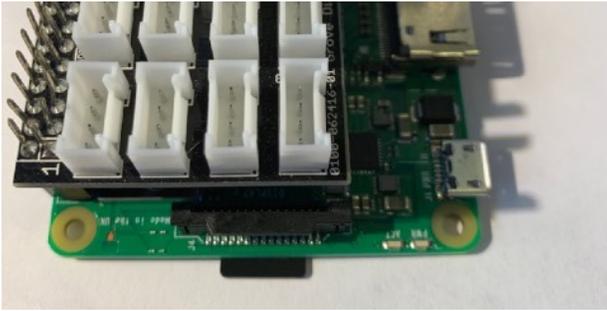
You will need an SD Card for your Raspberry Pi (Part B above).

You have included preprogrammed SD Card that has the Raspberry Pi operating system, version Buster, and all the MouseAir software installed. <https://shop.switchdoc.com/products/16gb-sd-card-with-stretch-smart-garden-system-groveweatherpi>

It contains all the software for running MouseAir and the Raspberry Pi. It is built on the Buster Raspberry Pi OS release.

Insert your SD Card (Part B) into the Raspberry Pi SD Card holder. It goes colored face down on the Raspberry Pi 3B+ and face up on the Raspberry Pi ZeroW. The picture below shows the SD Card plugged into a Raspberry Pi 3B+ (with the Pi2Grover board installed – Part A if you have the SkyWeather Kit).

(Default user: pi Default password: raspberry)



Once you have your Raspberry Pi setup up, running and can access a command line window (terminal) than you are ready to go with this manual.

This is the reason that we consider The MouseAir Kit an advanced beginners kit, rather than a beginners kit (like the OurWeather Weather Station).

Here are some resources to get you set up and running as quickly as possible.

Helpful Getting Started Videos:

<https://www.raspberrypi.org/help/videos/>

Helpful Getting Started Written Tutorial:

<https://www.raspberrypi.org/wp-content/uploads/2012/12/quick-start-guide-v1.1.pdf>

If you want to set up a headless (no monitor, keyboard, mouse) Raspberry Pi, it is more complicated. Here are some links to tutorials for that process.

<http://www.circuitbasics.com/raspberry-pi-basics-setup-without-monitor-keyboard-headless-mode/>

<http://blog.self.li/post/63281257339/raspberry-pi-part-1-basic-setup-without-cables>

Many, many more tutorials are available on the web.

Once you have it set up, take a brief tutorial about using the terminal window and the very powerful Raspberry Pi Command Line.

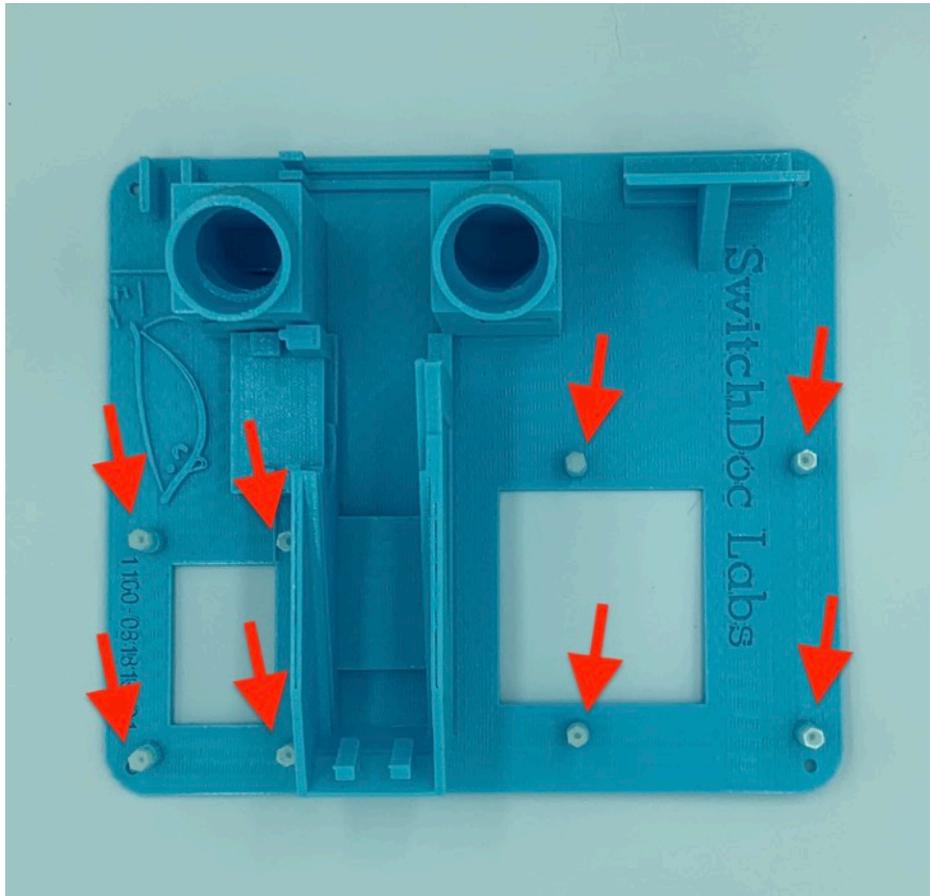
<https://www.raspberrypi.org/blog/learning-the-command-line/>

<https://www.raspberrypi.org/blog/learn-to-love-the-command-line-with-the-magpi/>

You don't need to know a lot about the command line to enjoy building and running MouseAir but you do need a bit of knowledge.

Step by Step Assembly

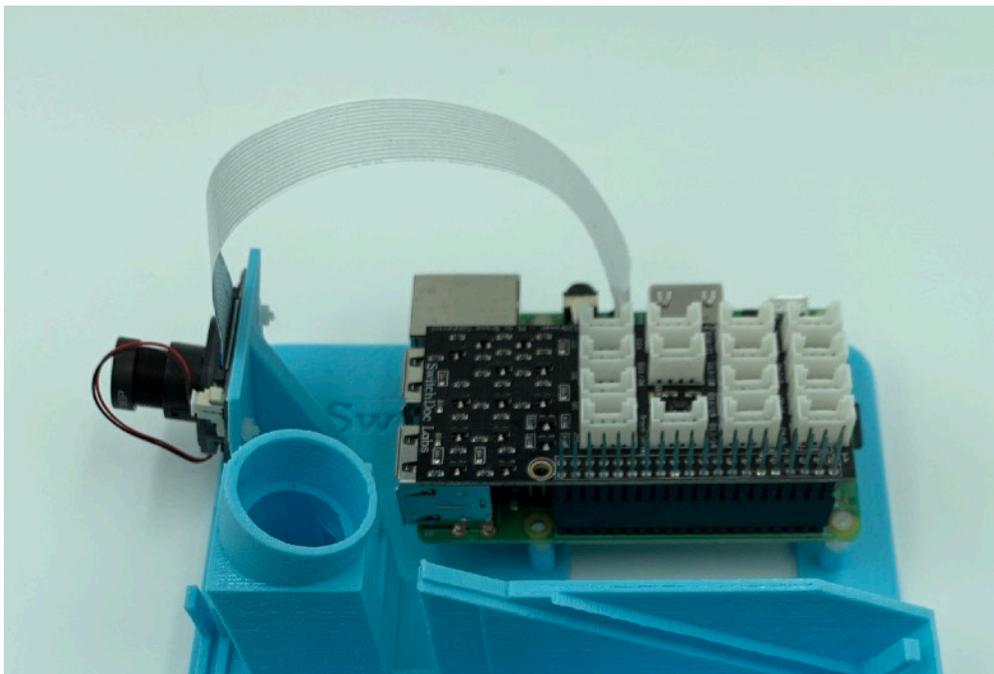
Step 1) Take MouseAir Base (Part M) and using super glue 8 M2 Posts (with the screw part of the post in the hole) in the 8 indicated post holes. Let dry.



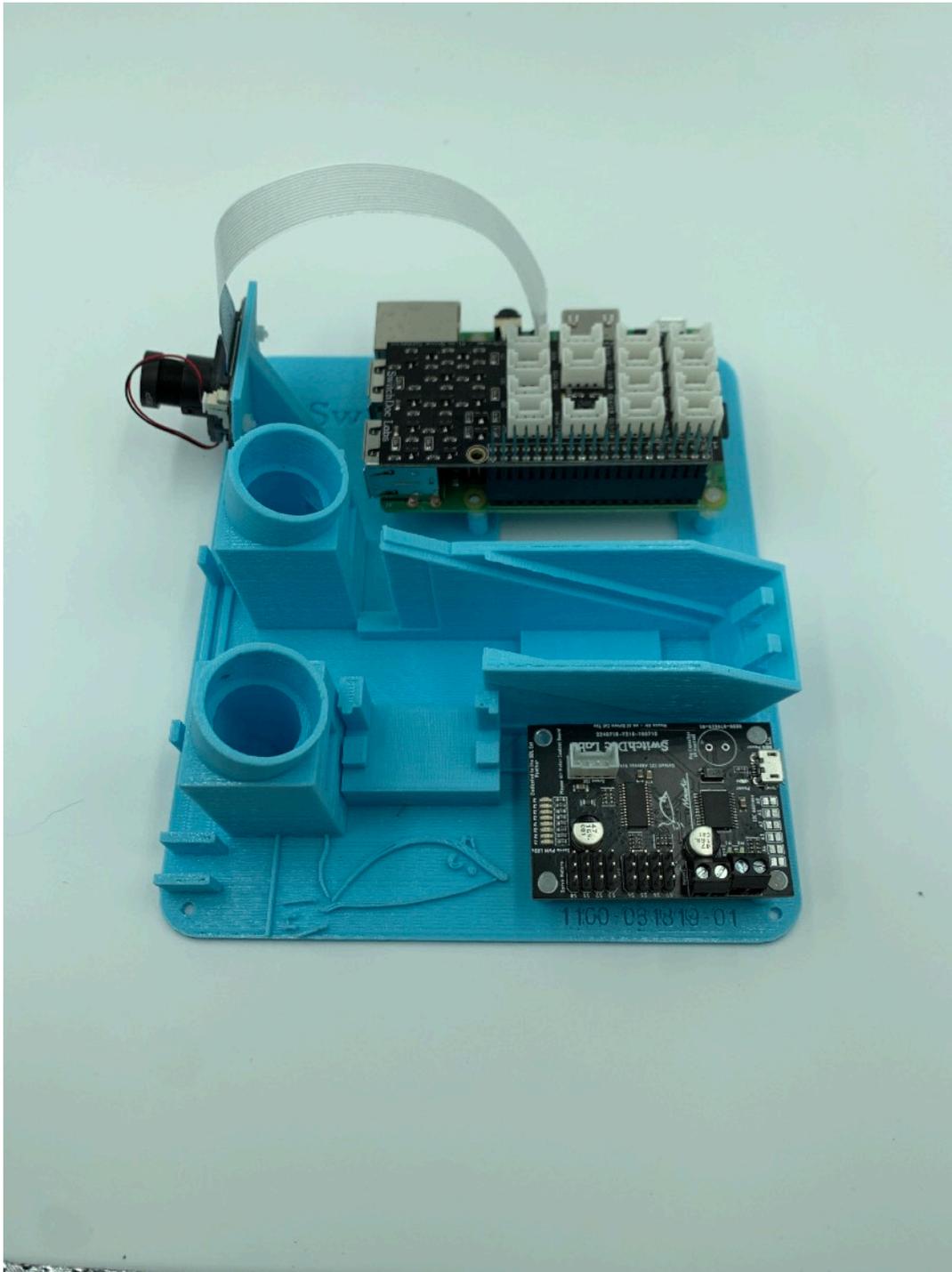
Step 2) Plug the your 120mm Pi Camera Cable (Part Q) into the Camera (note the orientation of the blue side) and the other end in your Raspberry Pi (Part A) (note the orientation of the blue side - towards the Ethernet Port)



Step 3) Using 4 M2 Screws (Part P), screw your Raspberry Pi (Part A) into the MouseAir Base (Part M) oriented as shown. Mount your Pi Camera (Part D) using the top holes of the stand and the Pi Camera using two M2 Screws and nuts.



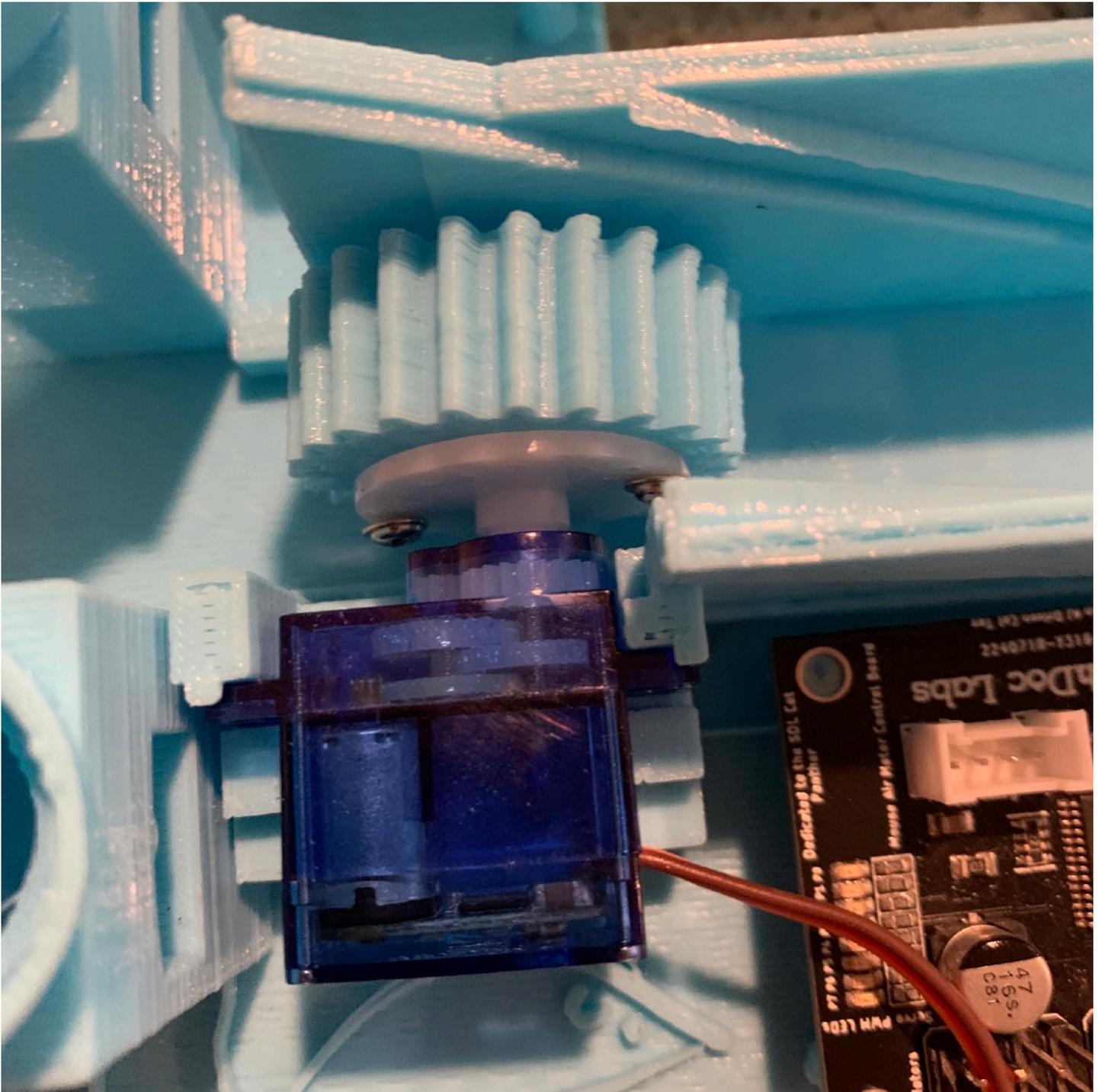
Step 4) Install the MouseAir Motor Controller Board (Part G) in the MouseAir Base in the location shown using 4 M2 Screws (Part P) on the MouseAir Base Plate (Part M). Orient the board as shown in the picture.



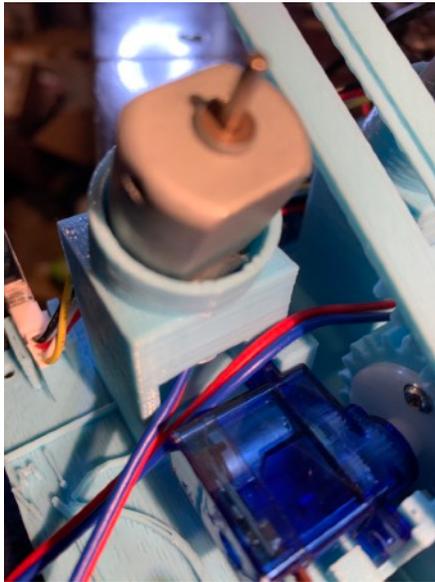
Step 5) Screw the circular servo motor plate (Part K) into the Large gear (Note: Two small holes on top) (Part N) as shown.



Step 5) Push the Gear on the FS90R Servo Motor (Part K) and gently snap the assembly as shown into the MouseAir Base Plate (Part M).



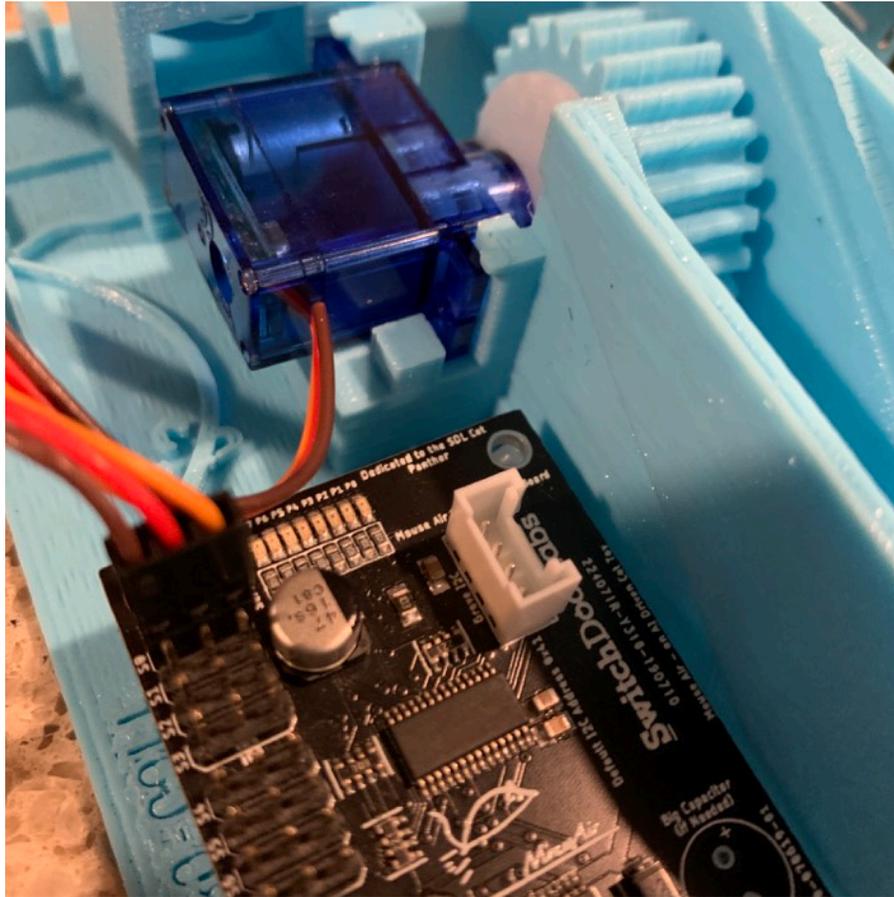
Step 6) Push the Right Motor #1 (Part I) (looking at the front the of the system) into the Right Motor Assembly on the MouseAir Base Plate (Part M) routing the wires through the hole next to the Servo Motor (Part K) and then screwing the wires into the two screw holes on the MouseAir Motor Controller (Part G) closest to the front the system with the blue wire closest to the front. All four wires for the motors are shown in this picture with the two for Motor #1 towards the top of the second picture below.



Step 8) Push the Left Motor #2 (Part I) (looking at the front the of the system) into the Left Motor Assembly on the MouseAir Base Plate (Part M) routing the wires through the hole next to the Servo Motor (Part K) and then screwing the wires into the two screw holes on the MouseAir Motor Controller (Part G) farthest from front of the system with the blue wire closest to the front. All four wires for the motors are shown in this picture with the two for Motor #2 towards the bottom of the second picture below.



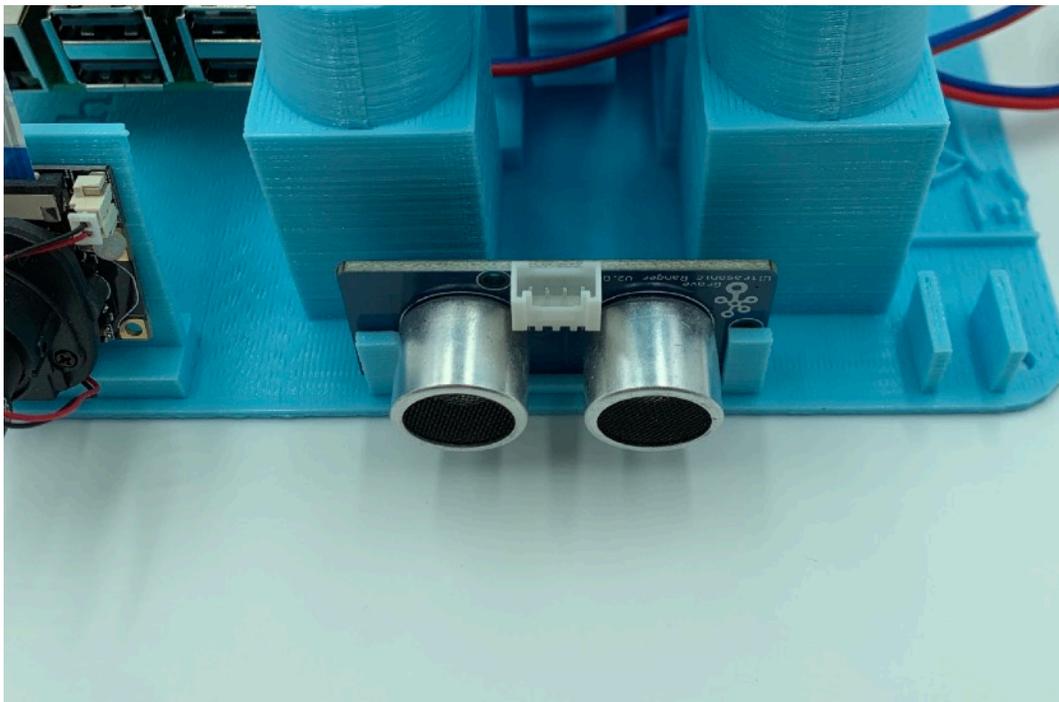
Step 9) Plug the 3 Wire plug coming from the Servo motor (Part K) into the three pins parallel to the front of the MouseAir Base Plant (Part M) on the MouseAir Controller Board (Part G). Orange Wire towards the left as shown below.



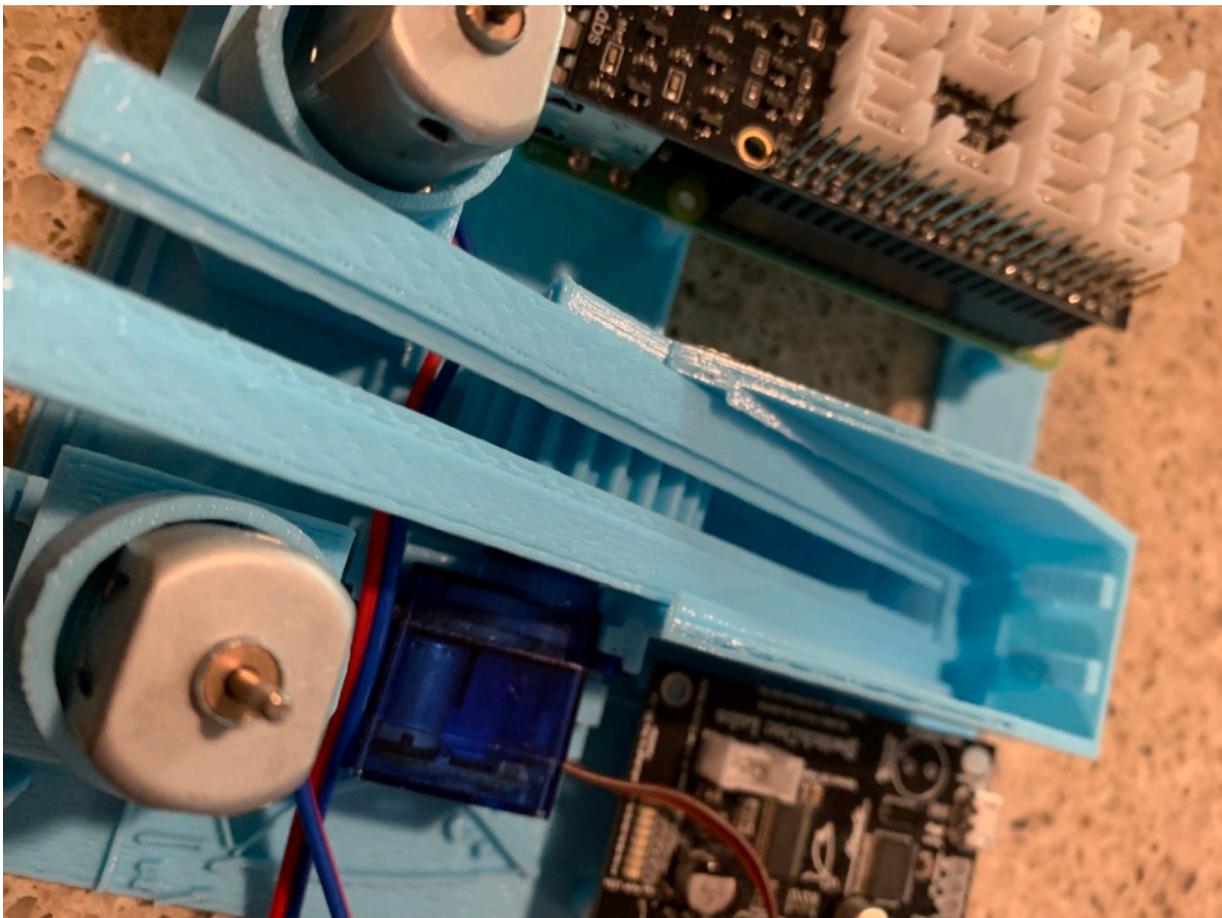
Step 10) Now take the Pi2Grover Board (Part R) and gently press it on the pins on the Raspberry Pi (Part A) as shown. Make sure that none of the pins on the top are bent and make sure you place it correctly on the Raspberry Pi as shown. Make sure the pins are aligned!



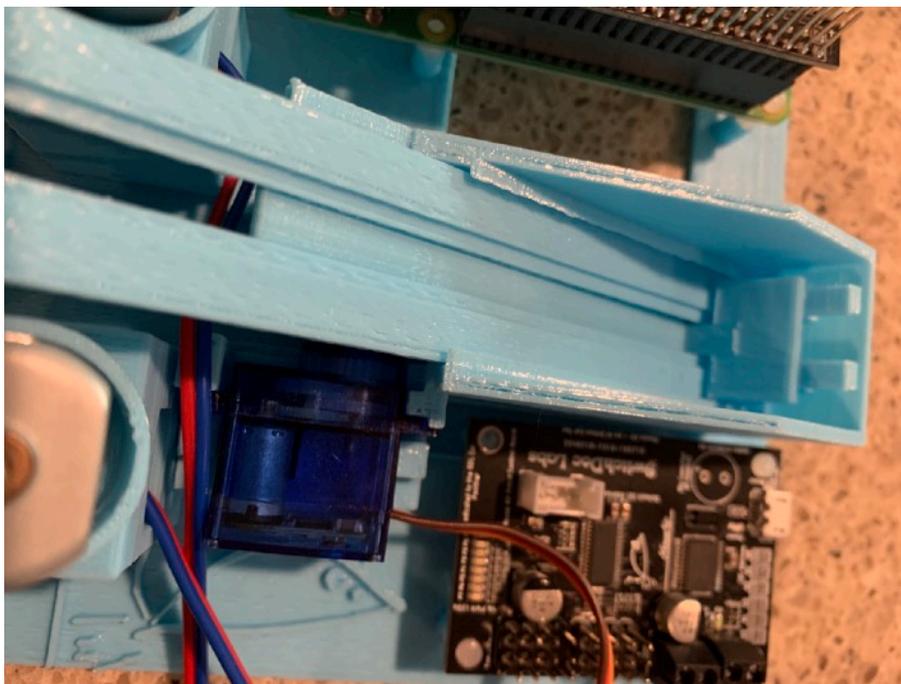
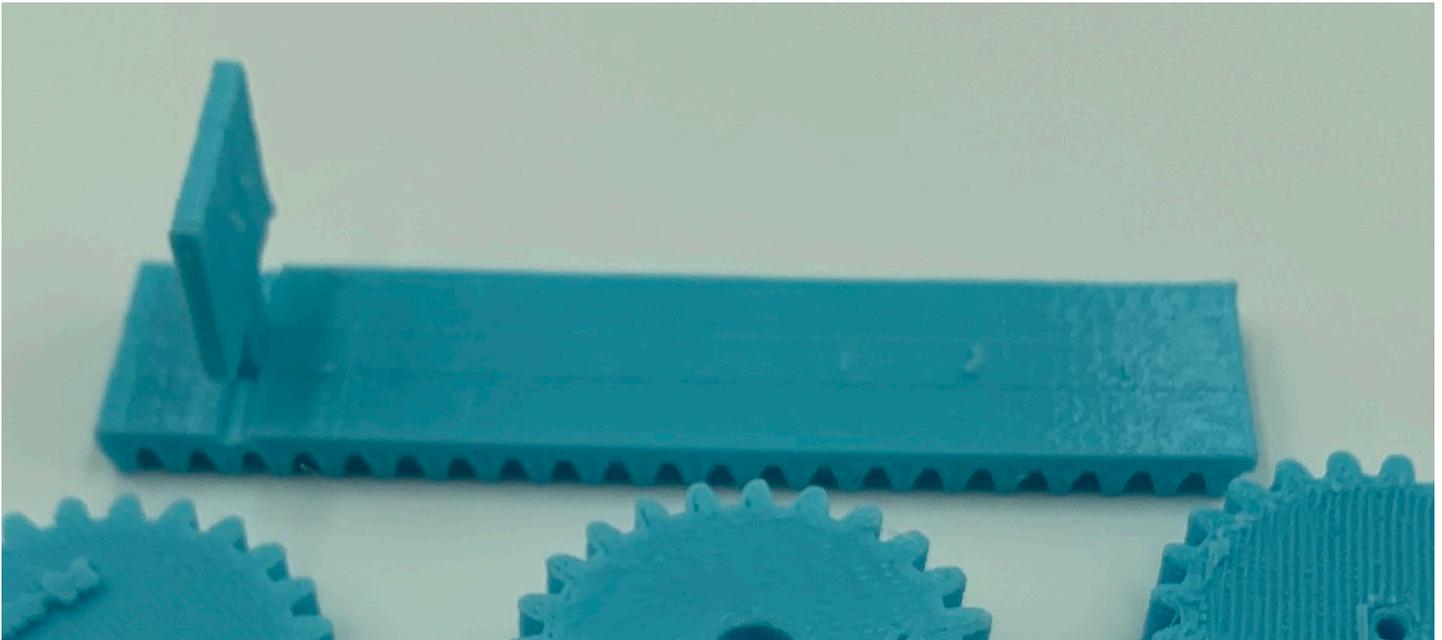
Step 11) Take the Ultrasonic Ranger (Part H) and push it into the front bracket (with the white Grove connector facing up) of the MouseAir Base (Part M).



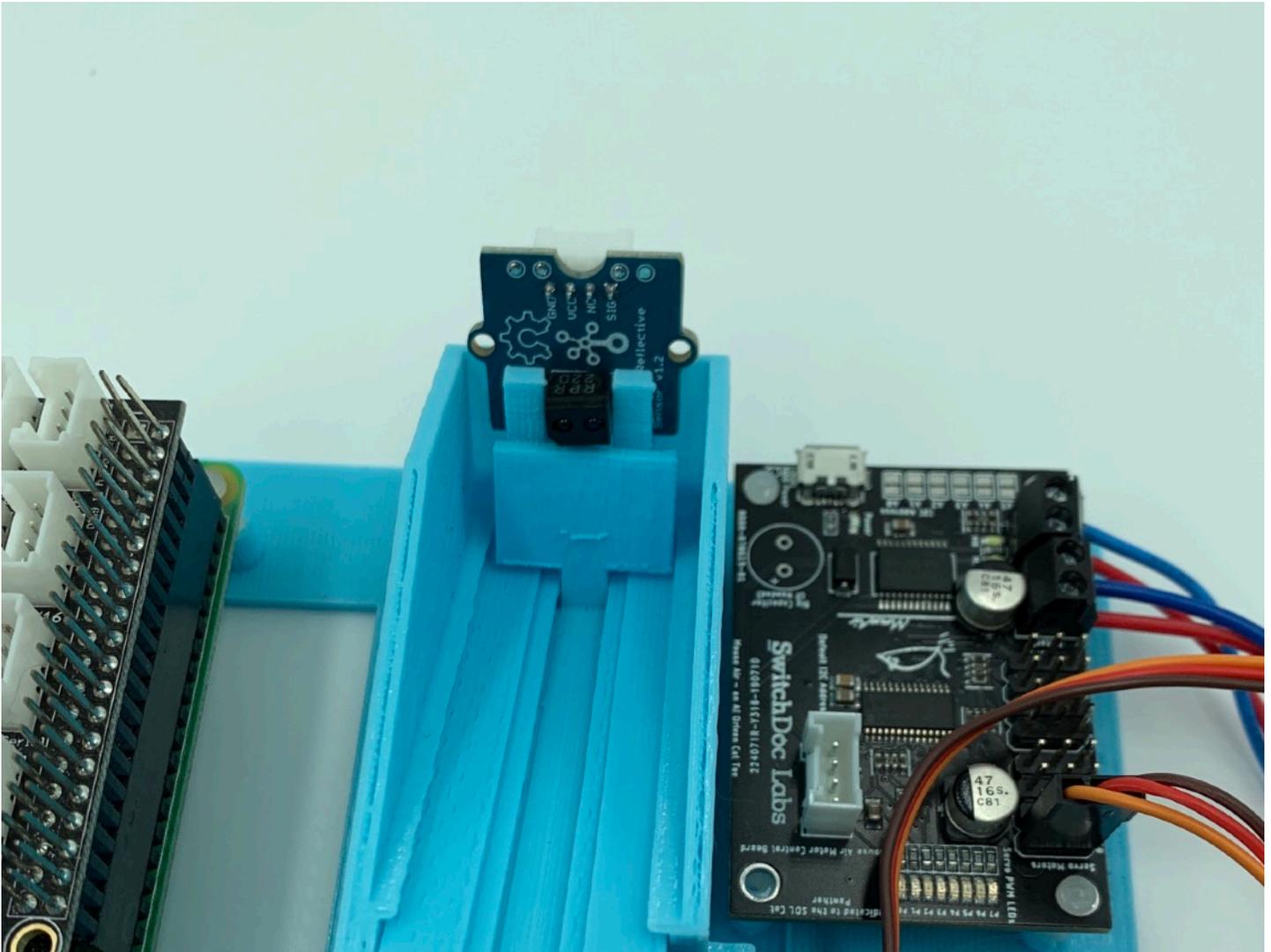
Step 12) Take the Launcher Base plate (Part N) and slide it into the launcher slot on the MouseAir Base Plate (Part M) as shown below.



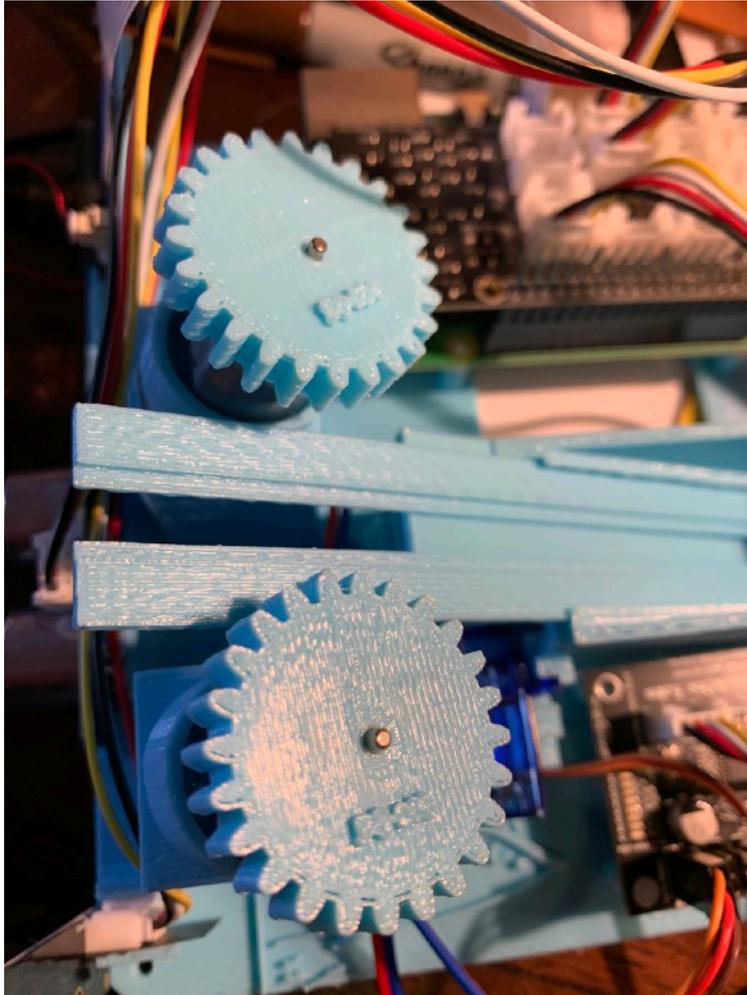
Step 13) Now take the Pusher Ratchet Gear (Part N) and with the Paddle up and in the slot gently push it into the launcher on the MouseAir Base Plate (Part M). You will engage the gear with the servo motor (Part K) but you will be able to turn it as you push it all the way in. The assembled launcher will look as below.



Step 14) Take the Grove IR Detector (Part L) and place it in the slot at the back of the Launcher Slot on the MouseAir base as shown (with the Grove connector facing the back). If it is too loose, place a piece of tape on the back of the Grove IR Detector (Part L) to wedge it better into the slot.



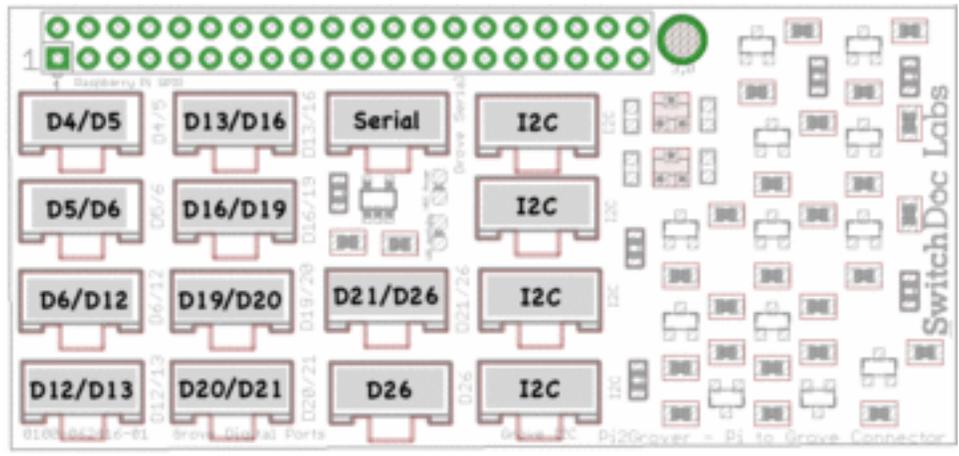
Step 15) Take the two remaining smaller gears remaining (Part N) and place one on each of the Motor #1 and Motor #2 Shafts as shown.



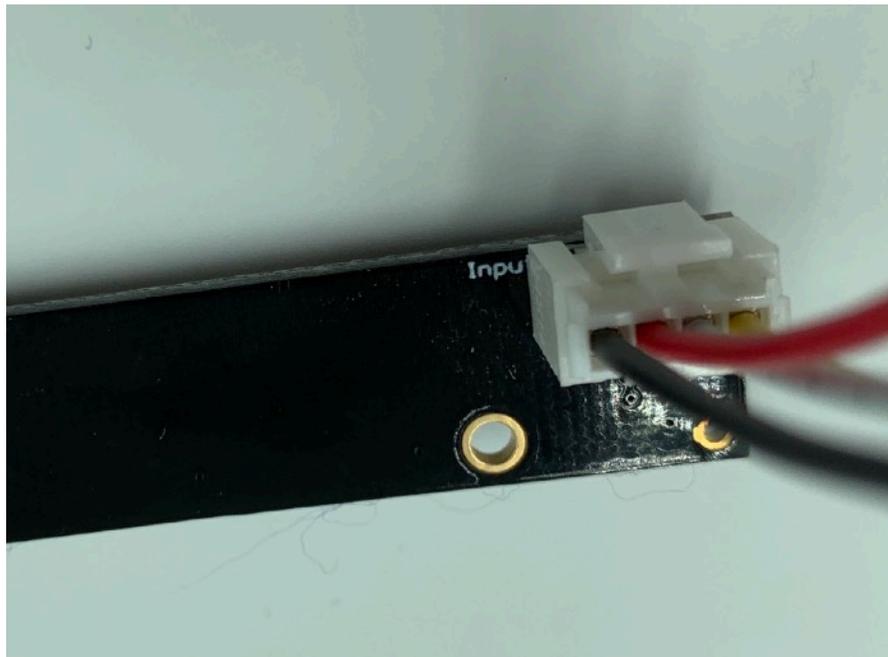
This completes the physical assembly of MouseAir. Now we proceed to the wiring.

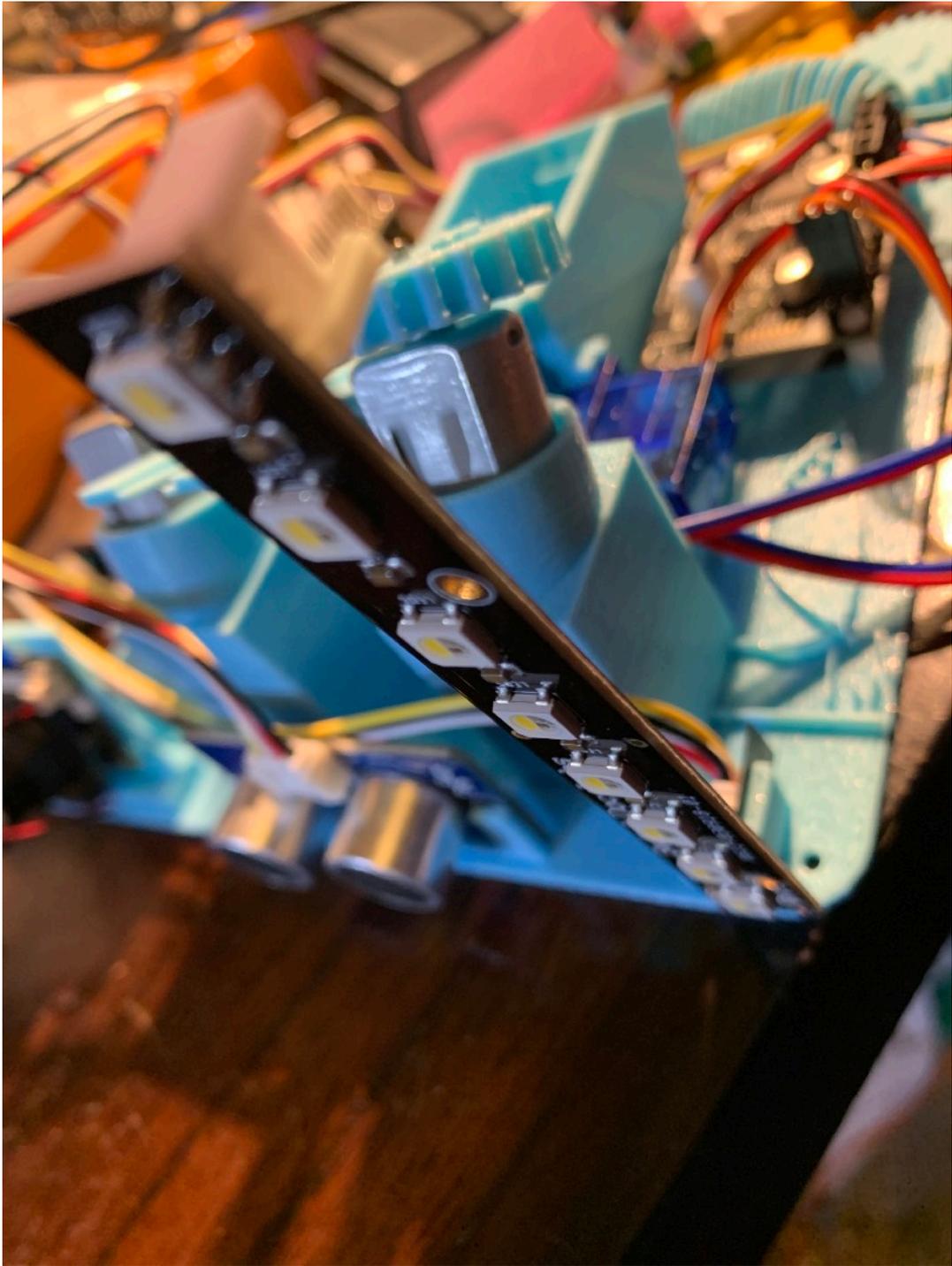
MouseAir Grove Wiring

The pi2Grover board (Part R) is the board used to complete all the device wiring to the Raspberry Pi (Part A). Note the location of all the Grove connectors in the diagram below:



Step 1) Take a 20cm Grove Cable (Part O) and plug it into the Grove Connector marked INPUT on the 8 Pixel Stick (Part H). Place the Pixel Stick in the front right slot of the MouseAir Base Plate (Part M) as shown. If it is a little loose, place some tape on the side to wedge it in.





Step 2) Complete the Grove Cable Wiring in the table below. All these are Grove 20cm Cables (Part O)

Pi2Grover	Device
D21/26	8 Pixel -Part H -(already Connected in Part 1) INPUT
I2C	I2C Grove Connector on MouseAir Motor Controller Board (Part G)
D26	Grove Ultrasonic Ranger (Part E)
D4/5	Grove IR

Congratulations! You have now completed assembly. Now on to testing.

Testing Your MouseAir System

Now that you have your Raspberry Pi running and can log into the device, it is time to test your hardware. Open a command line terminal window on your Raspberry Pi and do:

```
cd /home/pi/SDL_Pi_MouseAir
```

If any of the tests fail, check your wiring very carefully. It is unlikely you have a bad part. It is far more likely that you just haven't connected them correctly.

If you need to adjust the wiring, always shutdown your Raspberry Pi with "sudo halt" and unplug the Pi from power.

Test #1 - Mouse Detector

This test will show if your IR Mouse Detector is wired correctly and will work.

Type:

```
sudo python3 testMouseDetector.py
```

After the test starts, drop your mouse in the launching slot in front of the detector and you will see the mouse detected. When you are done, hit control-c. You should see something similar to following. After the test, remove the mouse!

```
pi@SwitchDocLabs:~/SDL_Pi_MouseAir $ sudo python3 testMouseDetector.py
```

```
Test Mouse Presence Detector Version 1.0 - SwitchDoc Labs
```

```
Program Started at:2019-11-03 13:30:57
```

```
Mouse NOT Detected
Mouse Detected
```

```
^CTraceback (most recent call last):
  File "testMouseDetector.py", line 37, in <module>
    time.sleep(0.3)
```

Page 37

KeyboardInterrupt

Test #2 - Ultrasonic Detector

This test will show if your Ultrasonic Ranger is connected correctly and functional.

Type:

```
sudo python3 testUltra.py
```

After the test starts, put your hand in front of the Ultrasonic Ranger and you should see the distance change. You should see something to this:

```
pi@SwitchDocLabs:~/SDL_Pi_MouseAir $ sudo python3 testUltra.py
Grove Ultrasonic get level and print
14.569cm
14.614cm
14.295cm
14.491cm
14.164cm
5.275cm
6.415cm
14.618cm
14.487cm
14.642cm
pi@SwitchDocLabs:~/SDL_Pi_MouseAir $
```

Test #3 - Launch Motors

This test will check to see if your motors and motor controller board are hooked up correctly. If the motors don't turn on check all your wiring. You will see lights on the Motor Controller board when the motors turn on.

Type:

```
sudo python3 testLaunchMotors.py
```

Both motors should turn on. Motor #1 (Right from Front) should spin CCW (Counter Clock Wise) and Motor #2 (Left from Front) should spin CW (Clock Wise). You will also see two LEDs turn on and off on the MouseAir Motor Control Board. Motor #2 always turns on first. You should see something like the following. Terminate the test with a control-c.

```
pi@SwitchDocLabs:~/SDL_Pi_MouseAir $ sudo python3 testLaunchMotors.py
```

```
Test Launch Motors Version 1.0 - SwitchDoc Labs
```

```
Program Started at:2019-11-03 13:54:39
```

```
moving motor
```

```
LSpeed, Rspeed 3854 3854
```

```
LSpeed, Rspeed 3854 3854
```

```
Page 38
```

Version 1.1 October 2019

```
LSpeed, Rspeed 3854 3854
^CTraceback (most recent call last):
  File "testLaunchMotors.py", line 28, in <module>
    Motors.launchMotorsOff()
  File "/home/pi/SDL_Pi_MouseAir/Motors.py", line 79, in launchMotorsOff
    time.sleep(0.5)
KeyboardInterrupt
```

shutting down motors

DEBUGGING INFO:

If neither motor turns on, see if the white LEDs are turning on on the MouseAir Motor Controller Board. If they are not, then check your wiring from the MouseAir Motor Controller Board to the Pi2Grover Board. It should be in an I2C slot on the Pi2Grover Board.

If the white LEDs turn on, then check your motor wiring. Make sure the wires are secure in the screw down terminals on the MouseAir Motor Controller board. If they are then make sure you haven't broken off the wires on the motors in the motor mount assemblies on the the MosueAir Base Unit.

If the motors don't turn the right directions (as shown above) then you have reversed the wires from the motors to the MouseAir Controller Board.

Test #4 - Launch Servo

First make sure there is no mouse in the launching slot.

Type:

```
sudo python testLaunchServo.py
```

You will see the servo move the paddle up towards the motors and then retract. This is how we launch the mouse. You will see something similar to the following. It runs the servo test twice and quits.

```
pi@SwitchDocLabs:~/SDL_Pi_MouseAir $ sudo python testLaunchServo.py
Moving servo on channel 0 0, press Ctrl-C to quit...
pi@SwitchDocLabs:~/SDL_Pi_MouseAir $
```

DEBUGGING INFO:

If the Servo doesn't work, first check the MouseAir Controller board wiring. Does the green LED turn on (to the left part of the board nearest the front of MouseAir) during the program execution? If not, check your MouseAir Motor Controller wiring. If it does not not, then check

your wiring from the MouseAir Motor Controller Board to the Pi2Grover Board. It should be in an I2C slot on the Pi2Grover Board.

Make sure that the servo is plugged into the first slot near the front of MouseAir and that the orange wire is to the left from the front and the brown wire is to the right from the front.

Test #5 - Shoot Sequence

Now the fun starts. We are doing our complete mechanical system test. First drop a mouse into the launch slot back against the paddle. You should see the red light (visible from the back) on the Mouse Detector light up. If you don't want to install the Blynk App, you can use this to shoot manually.

Type:

```
sudo python3 testShootSequence.py
```

MouseAir will start the motors, push the mouse up into the spinning wheels and launch the mouse! If it doesn't, go back through tests #1- #4 again.

DEBUGGING INFO:

If your mouse doesn't quite get to the motors, first run it again and make sure this is the problem.

If the mouse gets to the wheels and doesn't launch, the adjust the motors and wheels. Make sure the motors are pushed all the way down and then tilt the wheels a bit together. Put tape around the motors to hold them in place where you want them. All these mechanical and 3D printed parts vary slightly, so you may have to adjust things.

If after as few tests, the mouse still isn't reaching the wheels, then you can adjust these values in state.py. These numbers are in seconds are used to time how long you go forward, backwards and the delay between the movements. It won't take much.

```
LaunchTimeForward = 1.6  
LaunchTimeBackward = 0.8  
LaunchTimeDelay = 0.3
```

Test #6 - Machine Learning AI Test

This test will make sure that the Artificial Intelligence software is installed and works correctly.

Note: When you run this software, you will see many lines of warnings, which you can ignore. At the end you will see something like the following:

```
import complete  
Model: "sequential"
```

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 150, 150, 32)	896
max_pooling2d (MaxPooling2D)	(None, 75, 75, 32)	0
conv2d_1 (Conv2D)	(None, 75, 75, 32)	9248
max_pooling2d_1 (MaxPooling2D)	(None, 37, 37, 32)	0
conv2d_2 (Conv2D)	(None, 37, 37, 64)	18496
max_pooling2d_2 (MaxPooling2D)	(None, 18, 18, 64)	0
dropout (Dropout)	(None, 18, 18, 64)	0
flatten (Flatten)	(None, 20736)	0
dense (Dense)	(None, 64)	1327168
dropout_1 (Dropout)	(None, 64)	0
dense_1 (Dense)	(None, 2)	130
=====		
Total params: 1,355,938		
Trainable params: 1,355,938		
Non-trainable params: 0		

```

None
1
1.0
[[0. 1.]]
Our Network has concluded that the file 'AITraining/ukbench09941.jpg' is a NotCat
100% Confidence Level
0
1.0
[[1. 0.]]
Our Network has concluded that the file 'AITraining/cat.1332.jpg' is a Cat
100% Confidence Level
1.0 0.0
pi@SwitchDocLabs:~/SDL_Pi_MouseAir $

```

One interesting thing about this printout above is it shows you just how big this machine learning program is! 1,355,938 variables that have been trained.

Now on to installing the Blynk Application and controlling MouseAir.

Installing Your Blynk App

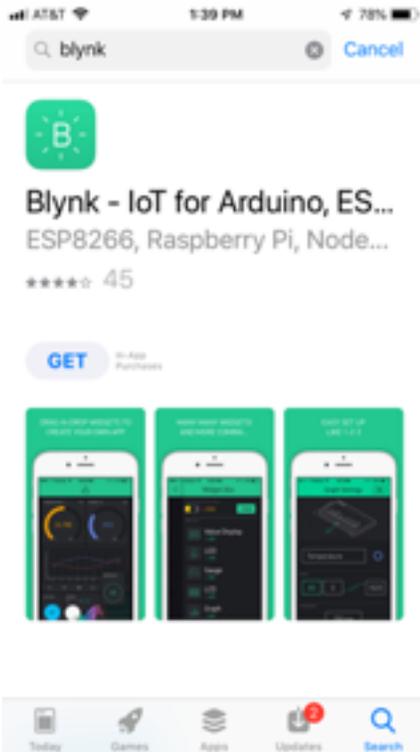
What is Blynk

Blynk is a digital dashboard for your iOS or Android device that allows you to easily build graphical interfaces by dragging and dropping widgets. Blynk runs on iOS and Android apps to control Arduino, Raspberry Pi and the likes over the Internet.

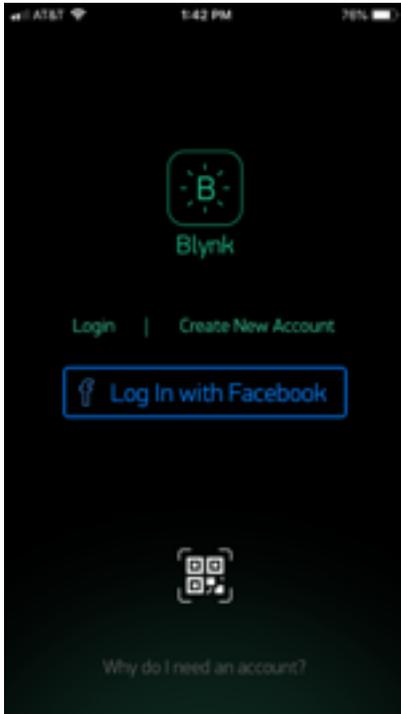
It's a digital dashboard where you can build a graphic interface for your project by simply dragging and dropping widgets.

Step by Step Install (for iPhones – Android is very similar)

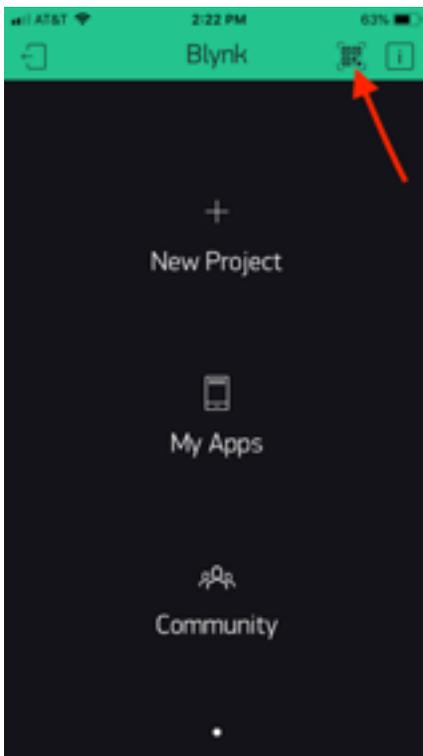
Step 1) Install the Blynk app on your mobile phone



Step 2) Open the Blynk app and create an account

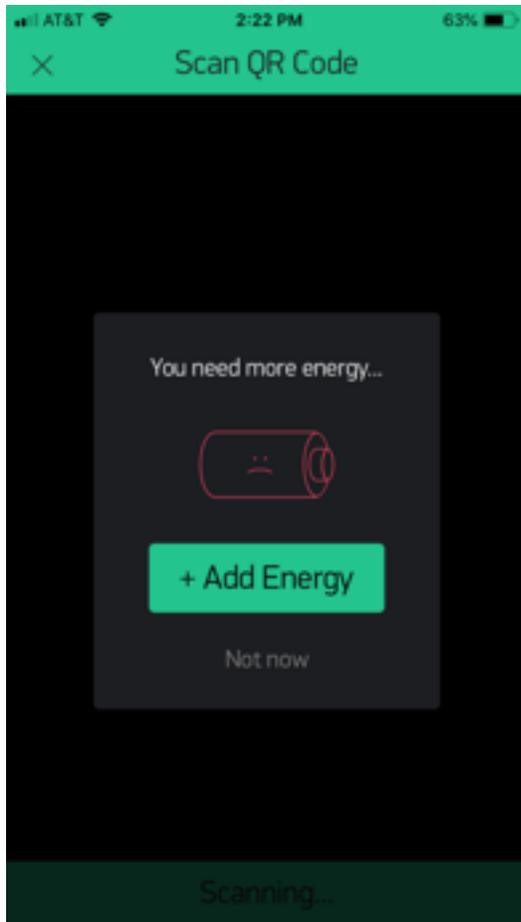


Step 3) Now click the button to scan a QR (see arrow)

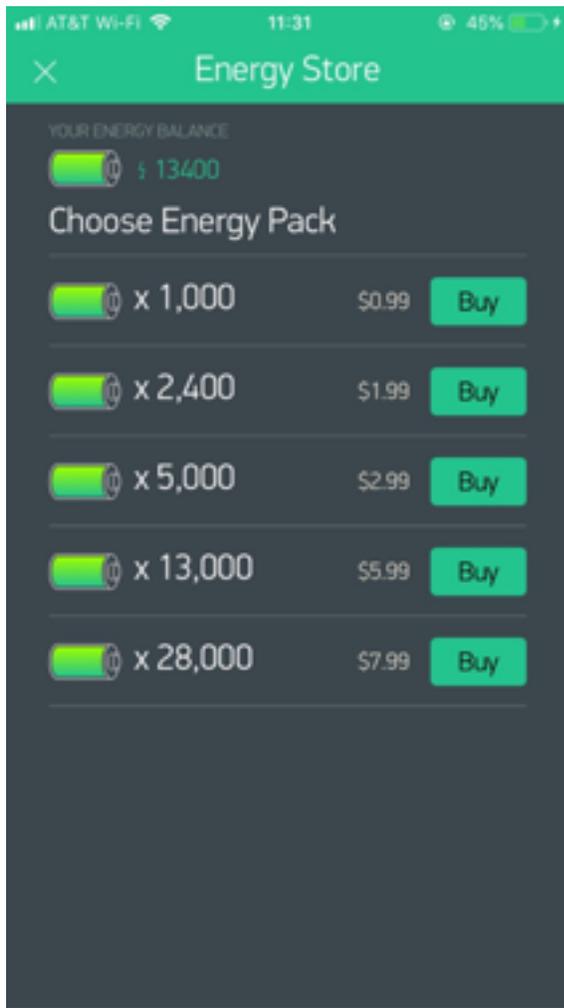


4) The current QR code for your Blynk App is located on the MouseAir Product Page.
<https://shop.switchdoc.com/products/mouseair-w-fullset-of-3d-prints-no-pi>

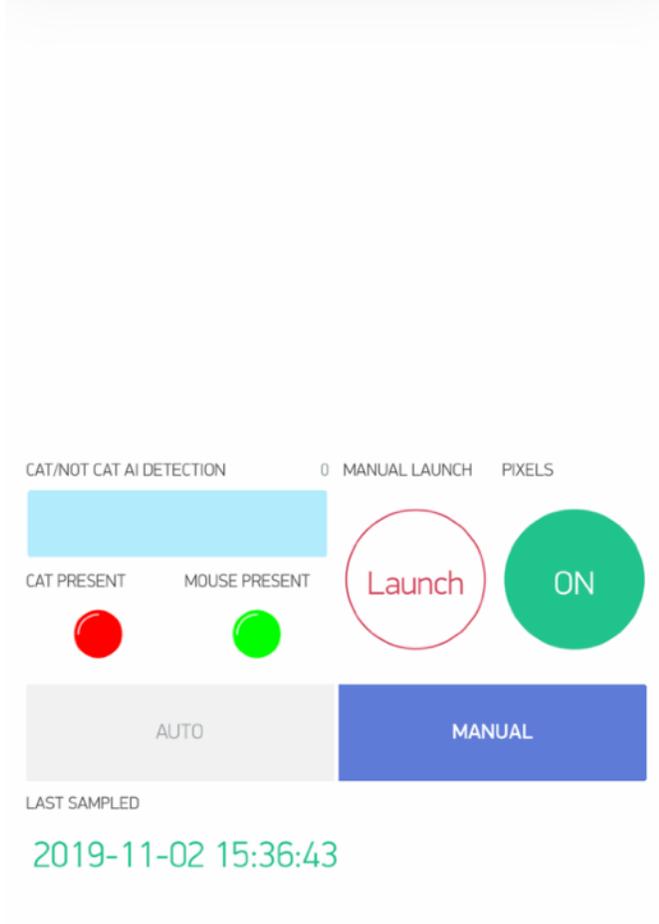
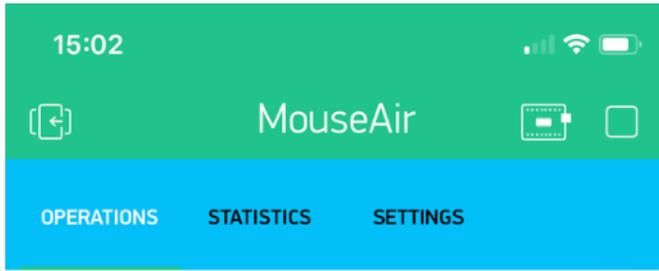
Scan the QR code on this page you will then see the screen below the QR code



Step 5) Add Energy. MouseAir requires a total of 5900 Blynk energy. You start your account with 2000, so you need to purchase 3900 Blynk Energy. As of this writing, it will cost \$5.99. It varies.

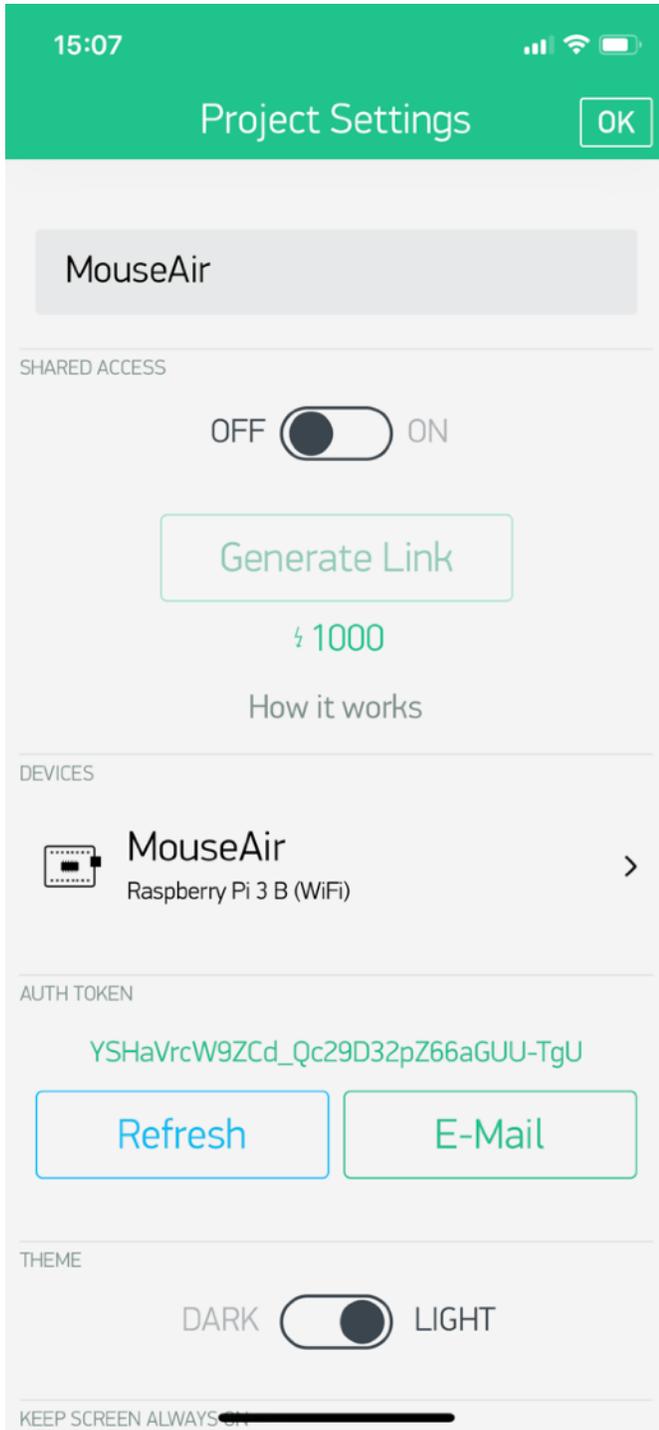


Step 6) Scan the QR code again. You will now see the MouseAir App on your screen.



Step 7) Click in middle of the project to select the project. Then click the indicated button to go to project settings. Now copy and paste or email yourself the authentication token (AUTH TOKEN) as we will be putting this in the MouseAir configuration file in the next section. The one below WILL NOT WORK. You need your own.

Your screens will look somewhat different than the one below.



You have completed the Blynk MouseAir App installation.

Installing the Blynk Authentication Codes into MouseAir

The final step to connecting MouseAir to the MouseAir App is to copy your Authentication Token (AUTH TOKEN above) to the MouseAir configuration file.

Step 1) using a terminal window, go to the SDL_Pi_MouseAir directory

```
cd ~/SDL_Pi_MouseAir
```

Step 2) Copy the config.py file to conflocal.py, if you haven't already. (this keeps you from copying over your changes when you update SkyWeather)

```
cp config.py conflocal.py
```

Step 3) Open the conflocal.py file with your favorite editor

Step 4) Change the following lines:

```
#####
```

```
# Blynk configuration
```

```
#####
```

```
USEBLYNK = False
```

```
BLYNK_AUTH = 'xxxxx'
```

```
BLYNK_URL = 'http://blynk-cloud.com/'
```

Change False to True

Replace xxxxx with your authentication token from above. Then it should look like this (but use YOUR authentication token – this one won't work):

```
#####
```

```
# Blynk configuration
```

```
#####
```

```
USEBLYNK = True
```

```
BLYNK_AUTH = 'daff75da224b447b69ce2e5b0abe5312'
```

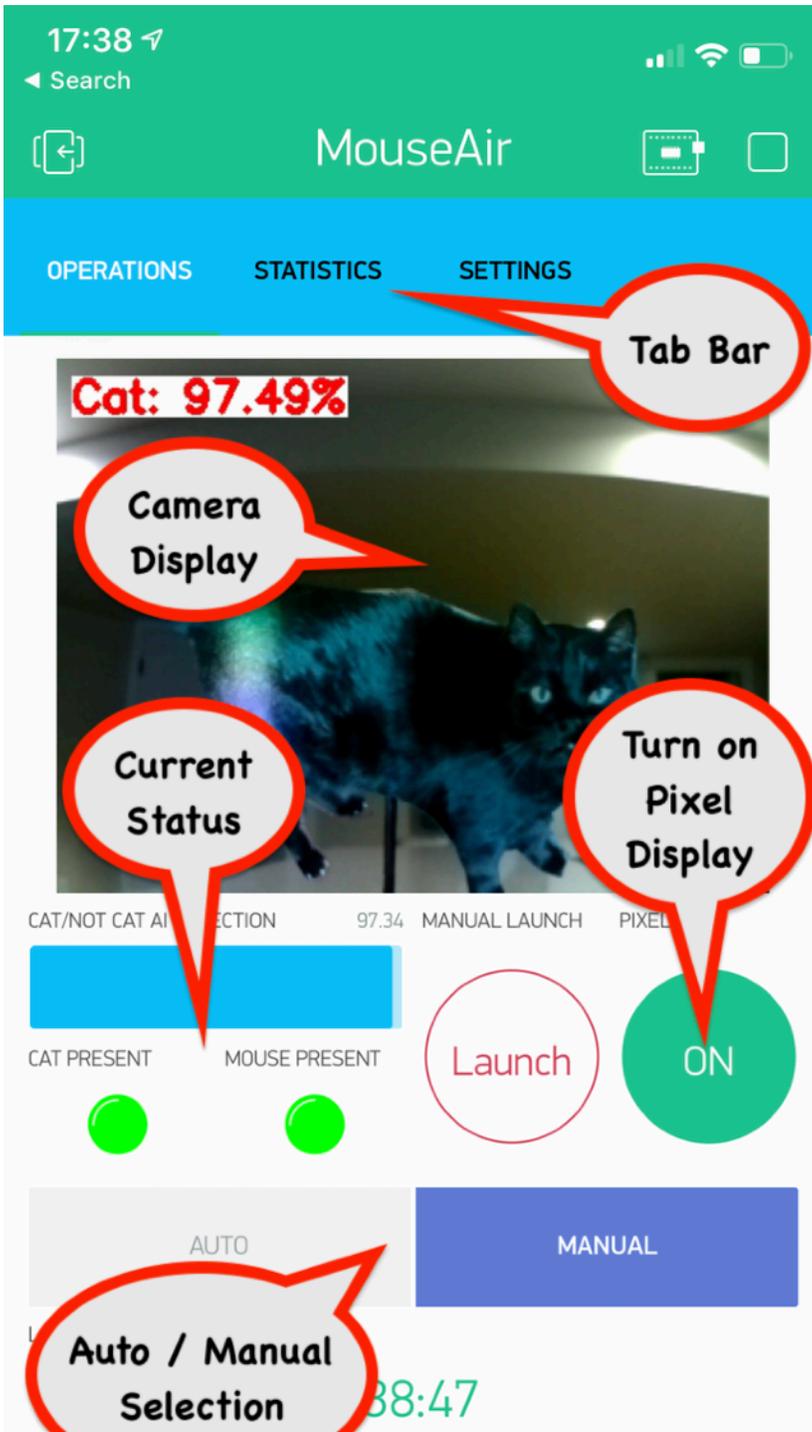
```
BLYNK_URL = 'http://blynk-cloud.com/'
```

Step 5) You are now complete. When we start up the MouseAir software your app will be ready to work.

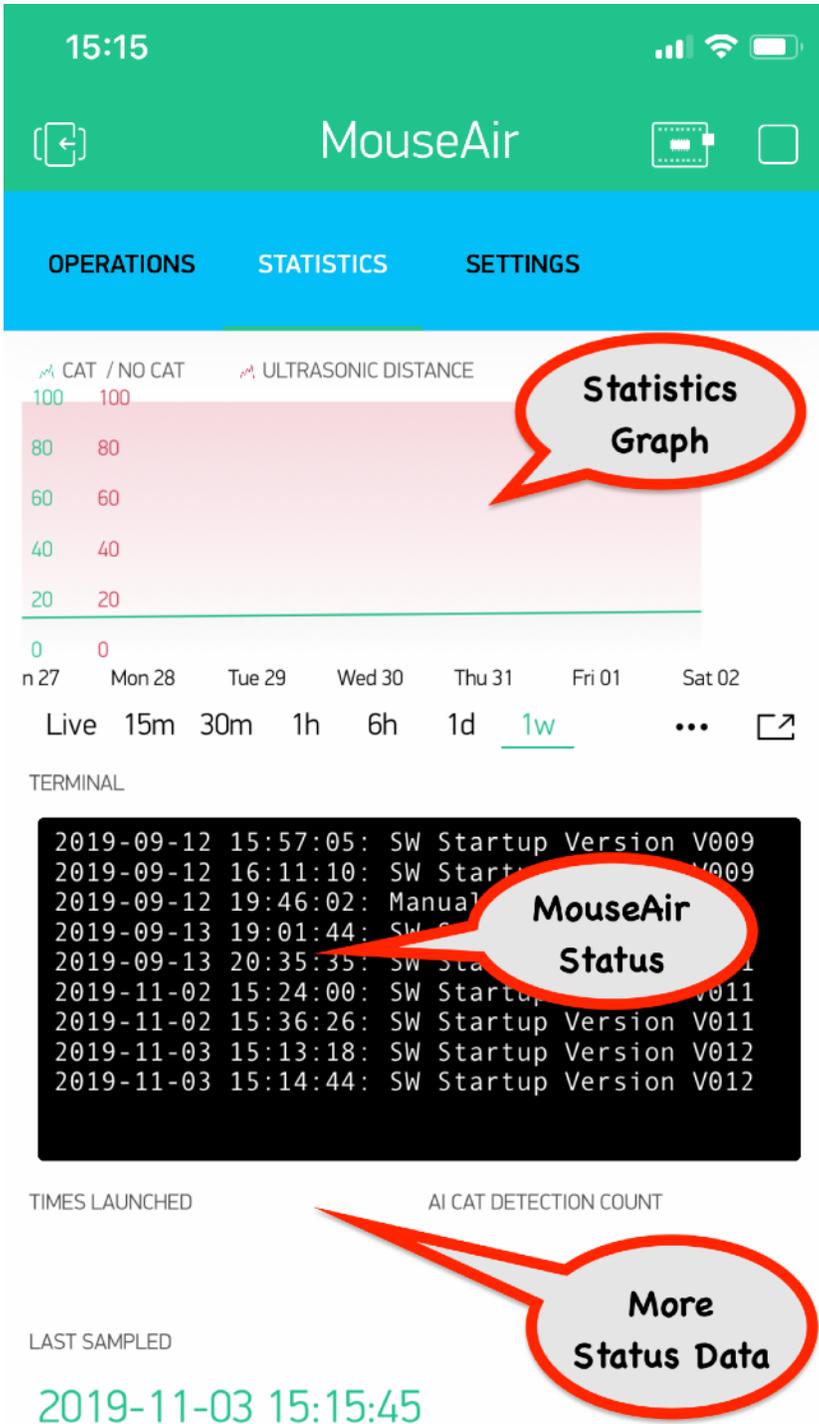
What is in the SkyWeather Blynk App?

There are three screens in the MouseAir app. You select the screen at the top of the page.

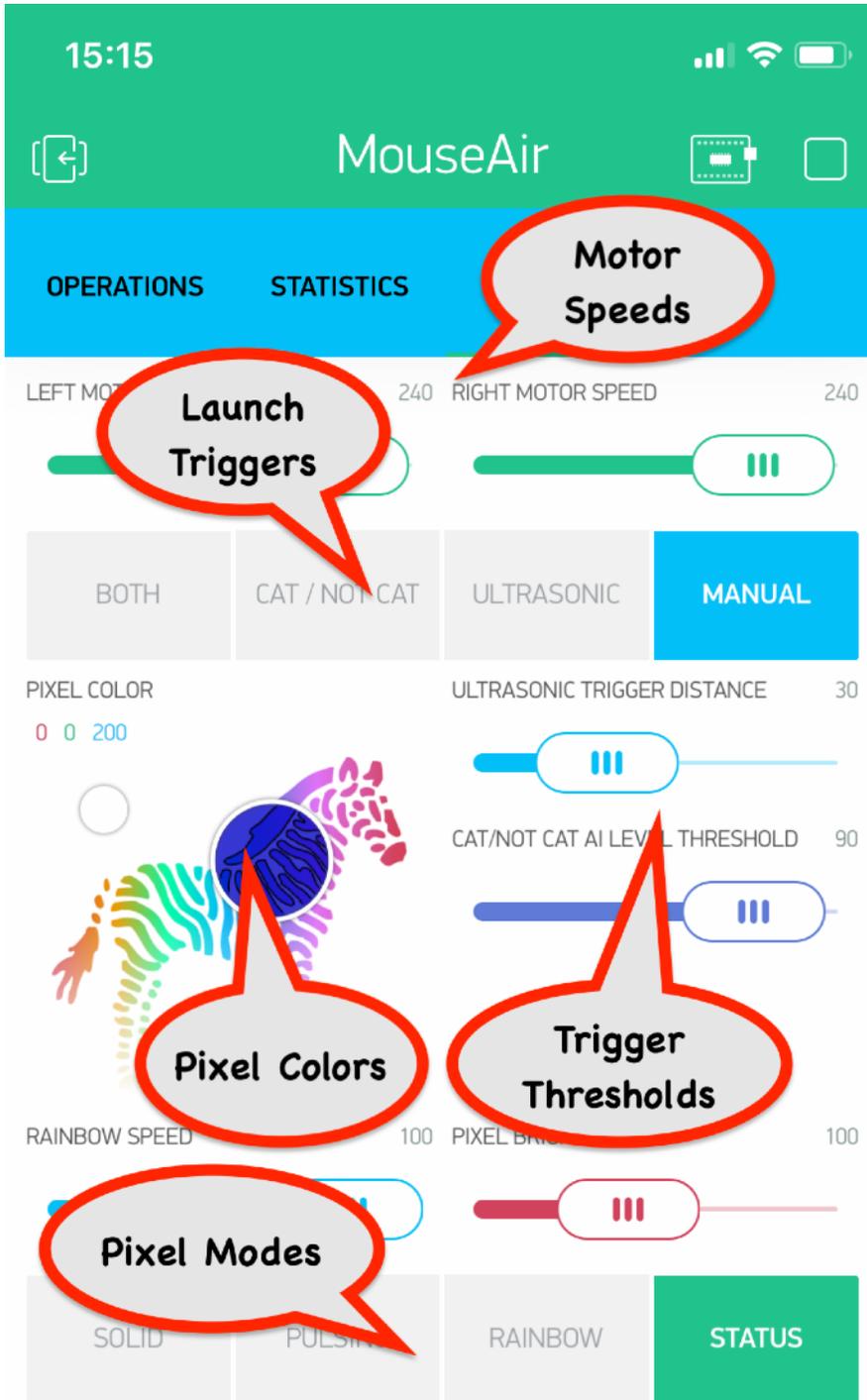
Main Operations Screen



Statistics Screen



Settings Screen



MouseAir Software Status is on the MouseAir Product Page.

<https://shop.switchdoc.com/products/mouseair-w-fullset-of-3d-prints-no-pi>

General MouseAir Operation

Now that we have tested our hardware and installed the Blynk app, it is time to run MouseAir and see it work.

Preparing your Pi

To run MouseAir, you will need to do either:

- Hook up your Raspberry Pi to an HDMI monitor, Mouse and Keyboard. Turn on the system and login to the Pi and open up a terminal window.

Or

- (Preferred) Set up VNC on your main computer and login into the Raspberry Pi via VNC. There are many tutorials on the web how to do this and it depends on the computer you are using. The SDCard is already set up to run VNC. You would VNC into <your ip address>:5900 . Example: 192.168.1.120:5900

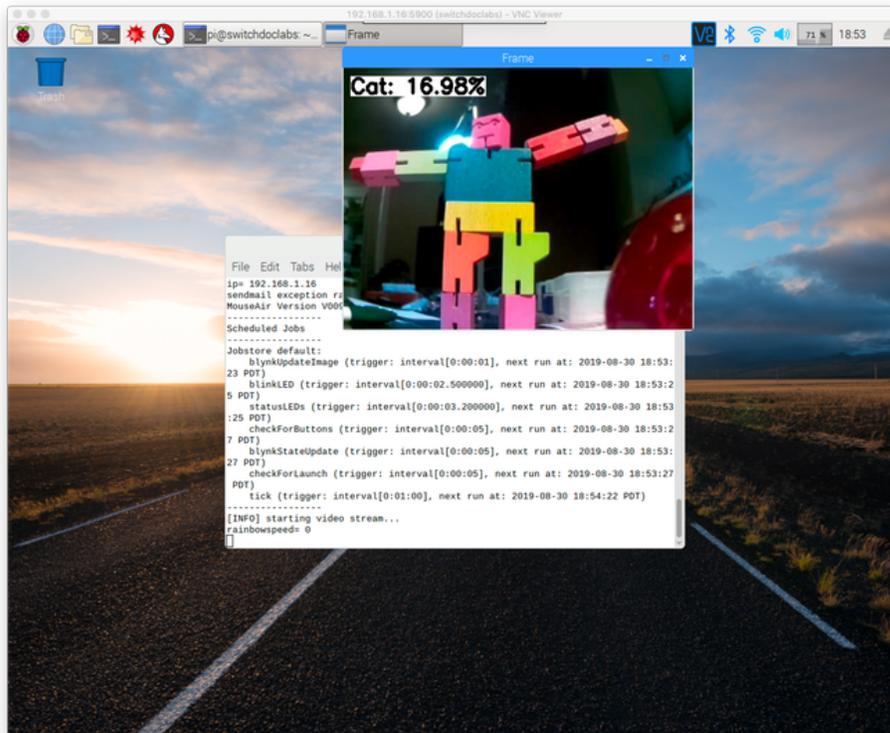
Now that you have a terminal window on your screen, you are ready to start MouseAir.

Starting MouseAir

Type into the Terminal Window:

```
cd SDL_Pi_MouseAir
sudo python3 MouseAir.py
```

You will be greeted (after a lot of lines of printouts and about 30 seconds) with a screen similar to this:



Now go over to your MouseAir app and hit the Launch button.

Experiment with the AI running on the camera. Remember that lighting is important to this AI. Too much or too little or a lot of clutter on the screen will lead to false Cat detections.

For any questions or comments (and remember, this AI needs more training!) please visit the tech support and discussion forums at forum.switchdoc.com and look for MouseAir.

Training the AI in Mouse Air

MouseAir is designed for the user and experimenter to improve the AI in the Cat/NotCat machine learning neural network. We trained the software to do a pretty good job under some lighting conditions to identify a cat moving into the picture. It has its quirks (like sensitivity to lighting conditions) that could be trained out of the AI.

Understanding Machine Learning in Python

Our CTO, recently completed a book called “Python All-in-One for Dummies” that covers AI on the Raspberry Pi and Machine Learning in Python. This book is really the source behind the technology used in MouseAir.

You can find the book here: <https://amzn.to/2NH5wbf>

Reading through the AI sections in this book are the best way to launch yourself into how MouseAir uses AI.

Doing your Own Training

In order to do effective training on the MouseAir AI, we suggest you get a Raspberry Pi 4B with 4GB. It works well. Not as fast as buying an NVidia card, but a lot cheaper.

You can do all the installs of TensorFlow, Panda, Numby and others into a NOOBs distribution, but we recommend starting with our SDCard for Buster that has all of this installed on the card. You can find it here:

<https://shop.switchdoc.com/products/16gb-sd-card-with-stretch-smart-garden-system-groveweatherpi>

If you want to try training, then download this software from github. The training package has all of the pictures that were used in the general training.

https://github.com/switchdoclabs/SDL_Pi_MouseAirTraining

Look at and run the programs in the archive to generate your own training data.

The Science and Education Goals Behind MouseAir

Everything we build for the Maker market is designed for education and learning. Making is education. Making is learning. Building your own projects allows you to innovate around a framework and do wonderful things that of which we have never thought.

The educational goals for MouseAir are:

- Understand interaction between hardware and the Raspberry Pi

- Explore the use of Artificial Intelligence in a practical application
- Understand the use of sensors and vision analysis

The MouseAir AI Machine Learning Program

The term Artificial Intelligence is being thrown around a lot these days to apply to a bunch of things that aren't really AI. It's a marketing term for these folks, rather than a technical term. The AI in MouseAir is an example of a neural network based Machine Learning program, which really is an AI technique.

If you want an introduction about AI and neural networks that are based on the Raspberry Pi, then please check out Book 4 of Dr. Shovic's book, "[Python All-in-One for Dummies](#)". We call our machine learning program Cat/NotCat because we are only interested in categorizing pictures as either being a Cat or Not being a Cat. It is either a Cat or NotCat. Sounds strange, but there we go!

Before you can use a new machine learning program, you have to train it. We train the MouseAir program by using 1000's of picture of Cats and the 5,000 - 10,000 pictures of NotCats. The neural network we are using has over 1,500,000 variables and about 160 neurons. This is a multi-level neural network model. The one we are using for early training is below:

```
model = tf.keras.Sequential()

model.add(Conv2D(32, (3, 3), input_shape=(150, 150, 3), padding='same', activation='relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Conv2D(32, (3, 3), padding='same', activation='relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Conv2D(64, (3, 3), activation='relu', padding='same'))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Dropout(0.25))
model.add(Flatten())
model.add(Dense(64, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(2, activation='softmax'))

...
```

You don't have to understand any of this to build or use the MouseAir project, but if you want to tinker with software this is a great thing to play with (like change it for dogs!) and experiment with. AI techniques are complicated to develop, but often they aren't that hard to use. Like Cat/NotCat.

Support

As with all SwitchDoc Labs products, technical support is given through the forums on Forum.switchdoc.com

If you have issues that can be solved by our fabulous customer service department, please go to www.switchdoc.com and send your issues through our Contact page on the top menu.

Disclaimer

SwitchDoc Labs, LLC takes no responsibility for any physical injuries and possession loss caused by those reasons which are not related to product quality, such as operating without following the operating manual and cautions, natural disasters or force majeure.

SwitchDoc Labs, LLC has compiled and published this manual which covers the latest product description and specification. The contents of this manual are subject to change without notice.