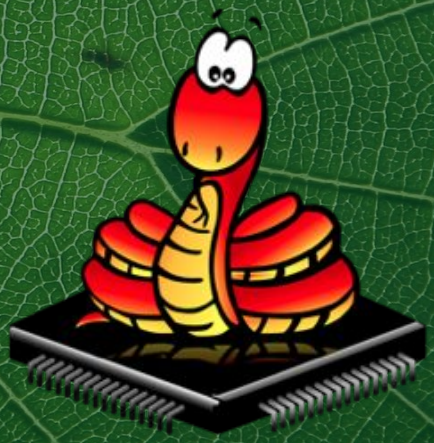
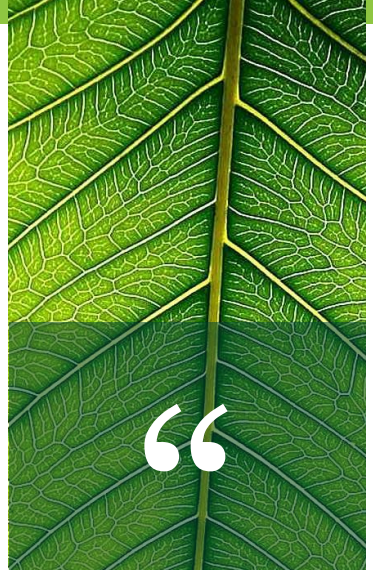
Smart Greenhouse Booklet 2019-2020



Created by Bethany Gately & Jaime Marsella



To plant a garden is to believe in tomorrow.

-Audrey Hepburn

Directions for Use:

- Please make a copy of this Google Slides before using
- This way you may remove/edit slides as you see fit

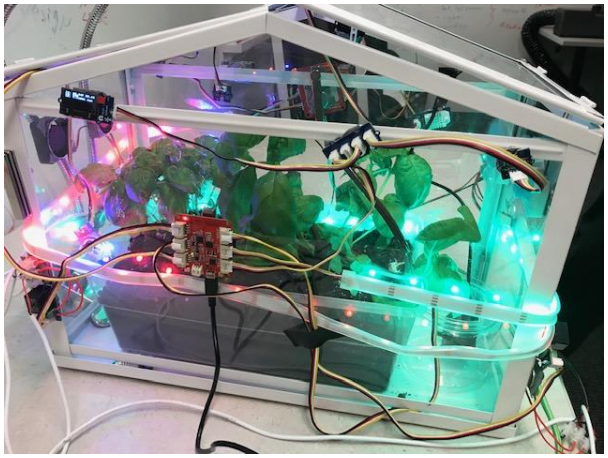
This booklet is meant to do the following:

- Provide a powerpoint for the teacher to project onto the board (to guide the lesson and record student brainstorms)
- Be accessible to students so they can access the directions, code, and reference section, and troubleshoot any problems.

Table of Contents

Table of Contents

The Mission:
Create & Maintain a "Smart" Greenhouse



Use coding to program a computer chip to take care of your plants for you, even while you're at home!

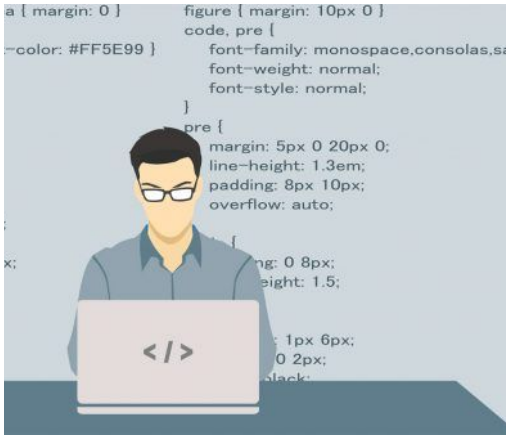


Table of Contents

Lesson 1a: Intro to Greenhouses

Table of Contents

Lesson 1a: Intro to Greenhouses

Questions of the day…

- What is a greenhouse?
- What variables affect plant growth?

Table of Contents

Note to teacher…

This first lesson (1a) is short such that you have time to set up your interactive notebook, online journal, or whatever method you choose to help students record their answers and data from the project.

You may also choose to complete the Pre-Survey (if applicable) during this time

Add link to survey here

Table of Contents

Do Now (Lesson 1a):

1. What is a greenhouse?
2. What variables affect plant growth?

Table of Contents

Class Discussion:

- What variables affect plant growth?
- What can we automate inside a greenhouse to make it “smart”?

Student responses:

Table of Contents

Methods of Growing Plants

What do you think works better?

Soil:
Growing plants in dirt

Hydroponics:
Growing plants in water

vs.

Table of Contents

Your task...
Adopt & Take Care of a Greenhouse!

- You and your team will become the “parents” of a greenhouse through *Plant Parenthood*
- As a group, give your greenhouse a name & fill out the adoption certificate!

Certificate of Adoption

On Behalf of **Plant Parenthood**

Name of Greenhouse: _____

By signing this form, I hereby agree to care for greenhouse to the best of my ability, to help the plants grow big and strong, and lead happy lives.

Adoptive Parent:

_____ _____ _____

(Printed Full Name) (Signature) (Date)

Table of Contents

Exit Ticket (Lesson 1a)

1. What does a greenhouse do?

2. What do plants need in order to survive?

Table of Contents

Lesson 1b: Intro to Coding

Table of Contents

Lesson 1b: Intro to Coding

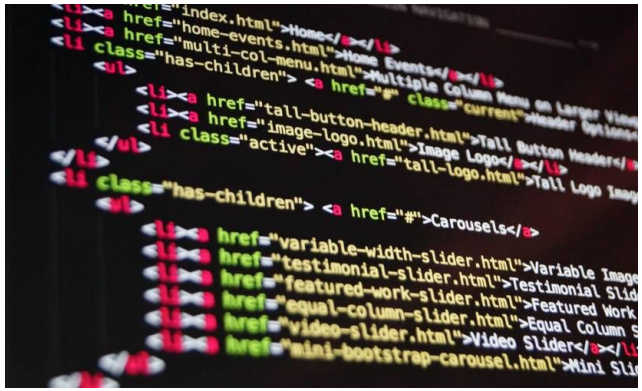
Questions of the day…

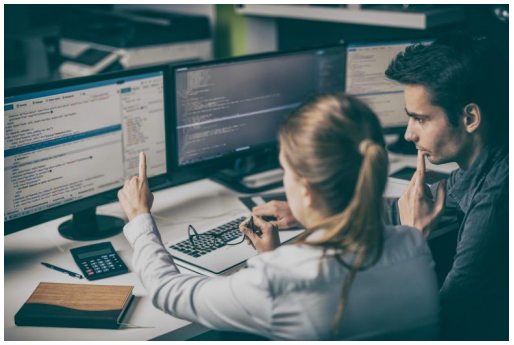
- What is “coding”?
- Why is coding important?
- What makes someone a good coder (computer programmer)?

Table of Contents

Do Now (Lesson 1b):

1. What is coding?
2. Why is coding important?
3. What makes someone a good coder (computer programmer)?





Student responses:

Table of Contents

What is Code?

Code is the language of computers!

You can program a computer so when it reads “gl.on”, it turns the lights on.This is because “gl.on” is the language the computer understands (ie. “code”).

It’s the INPUT!!

Table of Contents

What makes a good programmer (coder)?

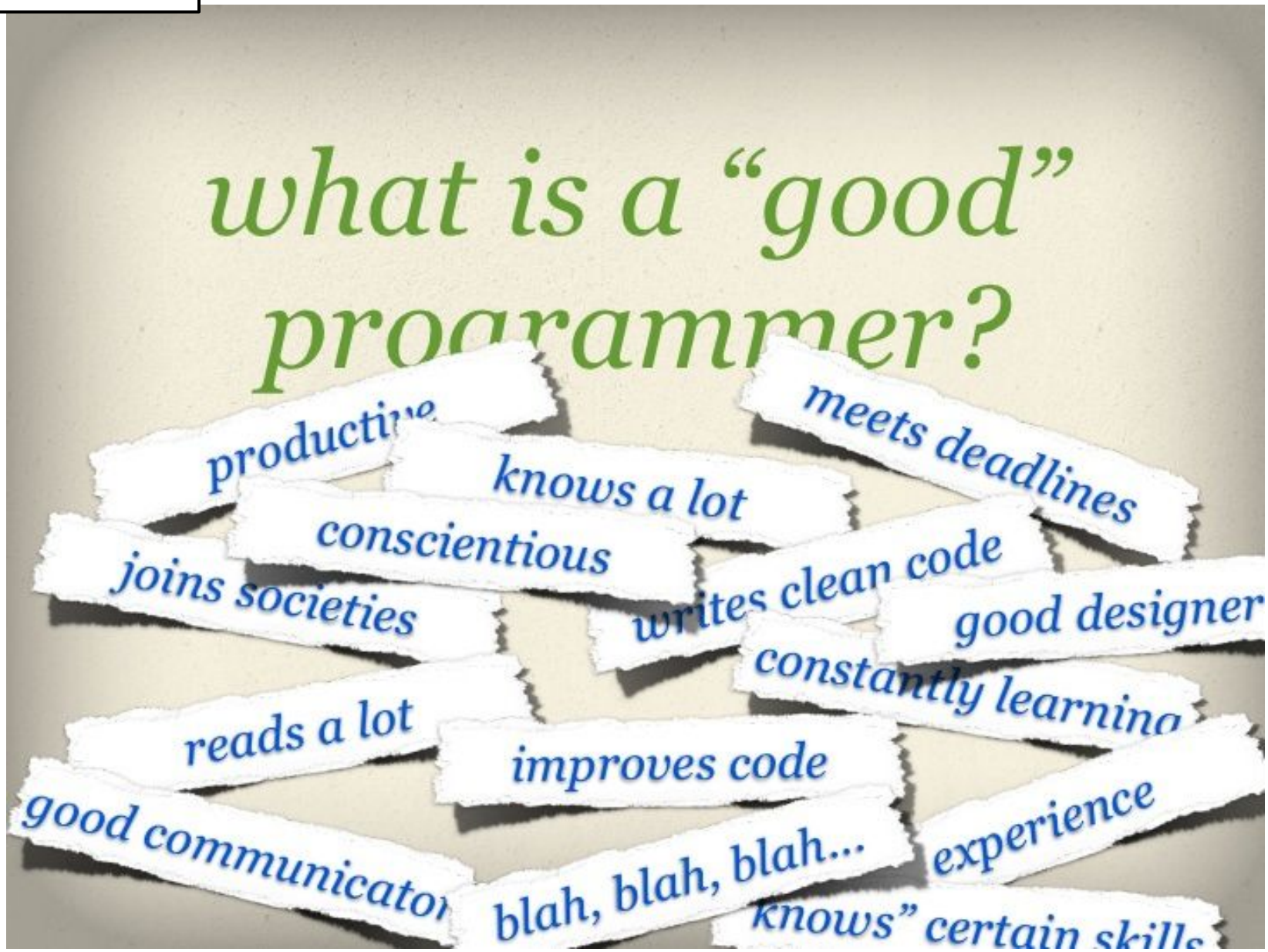

what is a "good" programmer?
productive
meets deadlines
knows a lot
conscientious
joins societies
writes clean code
good designer
constantly learning
reads a lot
improves code
good communicator
blah, blah, blah...
experience
knows" certain skills

Table of Contents

For this project, remember…

-Don’t give up
-Follow directions carefully
-Listen well
-Pay attention to EVERY detail!
    spelling,
    Capitalization,
    colons,
    Indenting (tabs)

Table of Contents

MicroPython?! Is it a snake?

MicroPython is a computer program that uses **code to control outside devices.**

Microcontrollers or “MCU” chips can be programmed to control devices (ex. lights or fans) that complete tasks while humans are away.

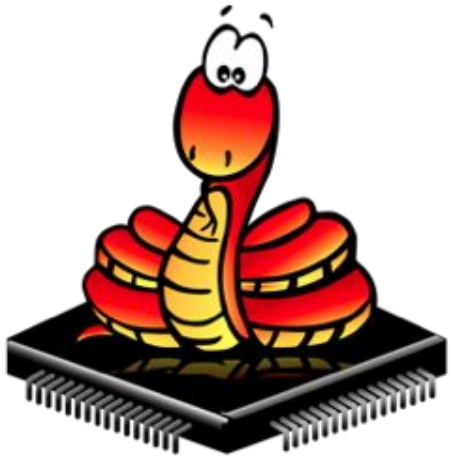


Table of Contents

Try out your coding skills!

Even ONE mistake can make the whole system fail…

The next slide shows code written by a student.

Can you identify the 7 errors in the student's code?

Remember that EVERYTHING matters!

- spelling
- Capitalization
- commas (,) and spacing
- colons (:)
- indenting (Tab)
- missing lines

Table of Contents

Ignore these line numbers & indent markers for now…

```
1   from sensors import LightSensor
2   from displays import GrowLight
3
4   gl = GrowLight(1)
5   ls = LightSensor(6)
6
7   Lux_HIGH = 1000
8
9   while True:
10      light = ls.get_lux()
11
12      if  light > Lux_HIGH:
13          print("Light level is good. Your Grow Lamp should be off", light)
14          gl.off()
15          relay.off()
16      else:
17          relay.on()
18          gl.on()
19          print("Light level is BAD. Your Grow Lamp should be ON", light)
```

← Correct Code

Directions: CIRCLE what is wrong and draw an ARROW ↗ to insert what SHOULD be there!

Student Code (7 errors):

```
from sensors import LightSensor


gl = Growlight(1)
ls = LightSensor(6)

Lux_high = 1000

while True
        light = ls.get_lux()

        if light  >  Lux_HIGH:
        print ("Light level is good. Your Grow Lamp should be off",  light)
            gl.off()
            relay.off()
        else:
            rellay.on()
            gl.on()
            print("Light level is BAD. Your Grow Lamp should be ON"  light)
```

← Student Code

Table of Contents

Exit Ticket (Lesson 1b)

1. What is coding?

2. What qualities do you already have that will make you a good coder?

3. What qualities do you need to work on to become a good coder?

Table of Contents

Lesson 1c: Intro to Devices

Table of Contents

Lesson 1c: Intro to Devices

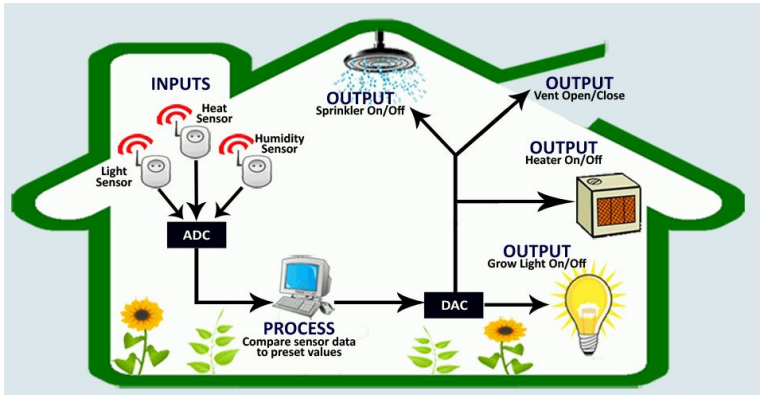
Questions of the day…

- What devices will be used in the greenhouse?
- What classes/libraries do these devices belong to, and why?

Table of Contents

Do Now (Lesson 1c):

What would you like to automate (make automatic) within your greenhouse? Name 3 things.



Student responses:

Table of Contents

Creating an Automated "Smart" Greenhouse

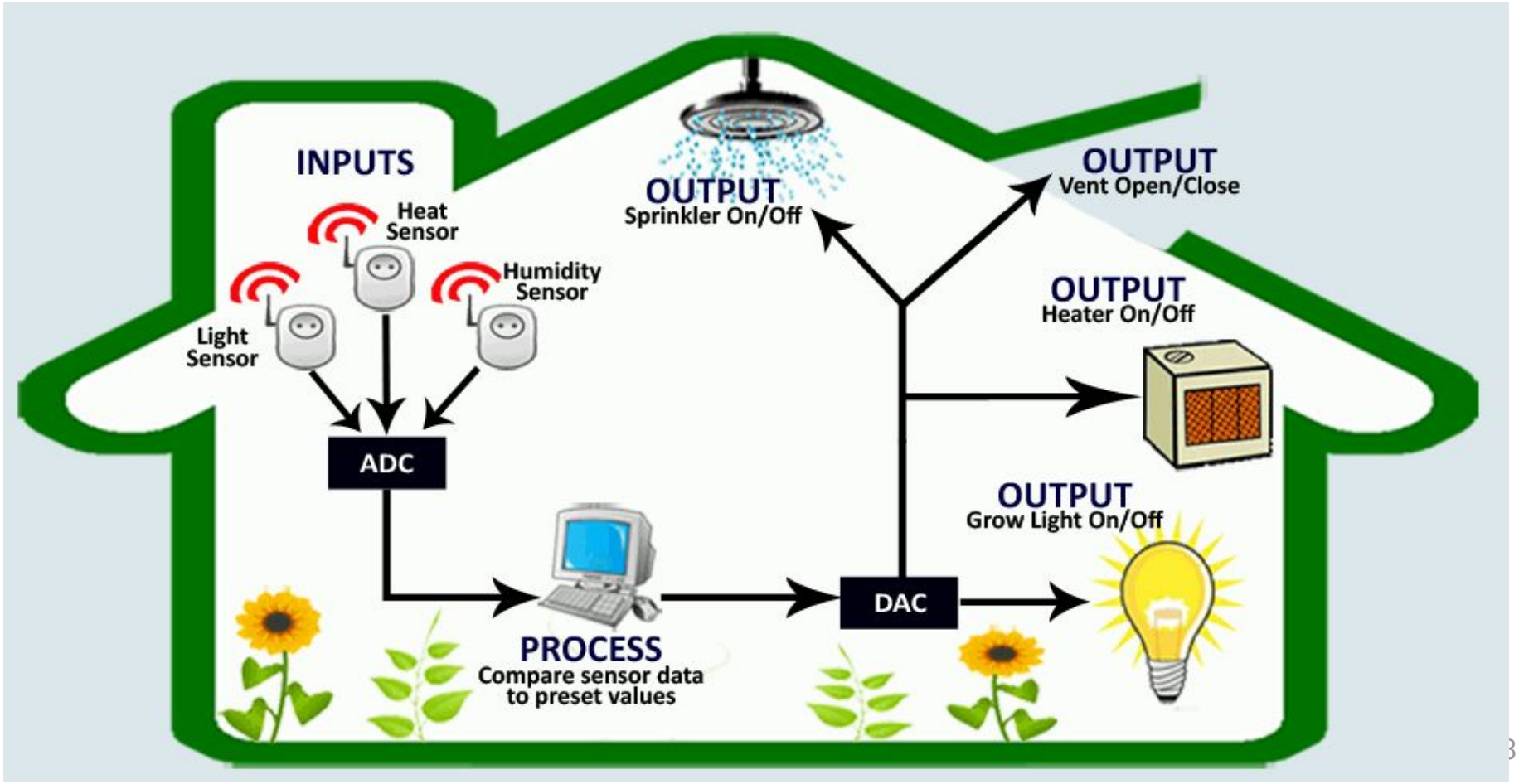


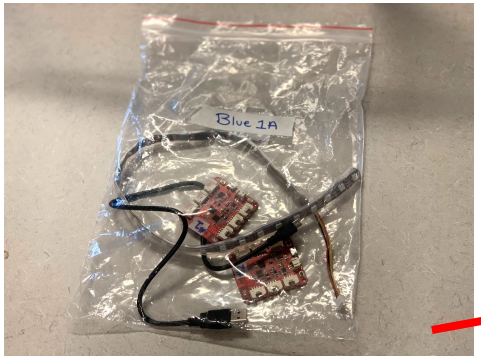
Table of Contents

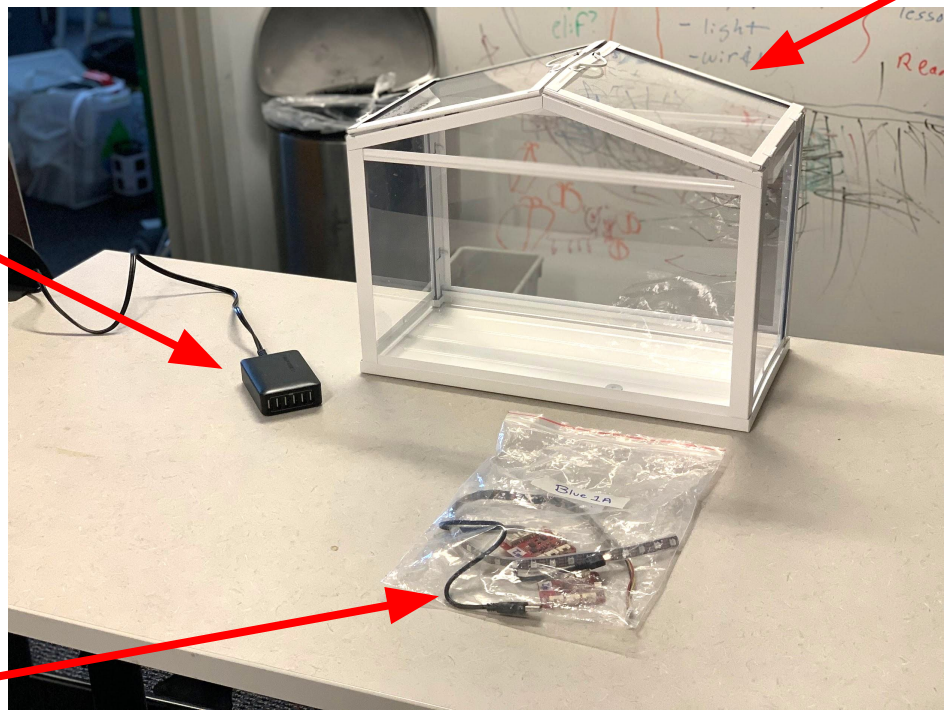
Station Setup

This is what your lab area should look like!

Greenhouse

USB Power Hub





Labeled Group Bag

Table of Contents

CODING VOCABULARY

- **Library** – a set of programs that controls an object or does something
  - 3 Libraries: Displays, Actuators, and Sensors
- **From** – tells the microcontroller (MCU) which library
- **Import** – tell the microcontroller (MCU) to get a device
- **Port –** the location on the microcontroller (MCU) where a device will be plugged in (ports 1-6)

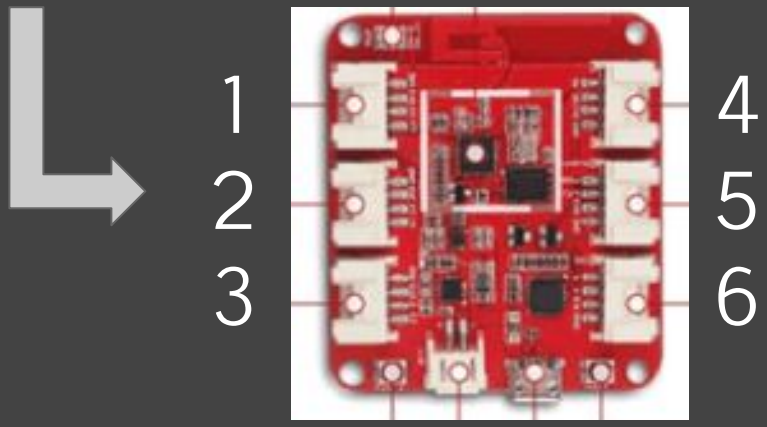
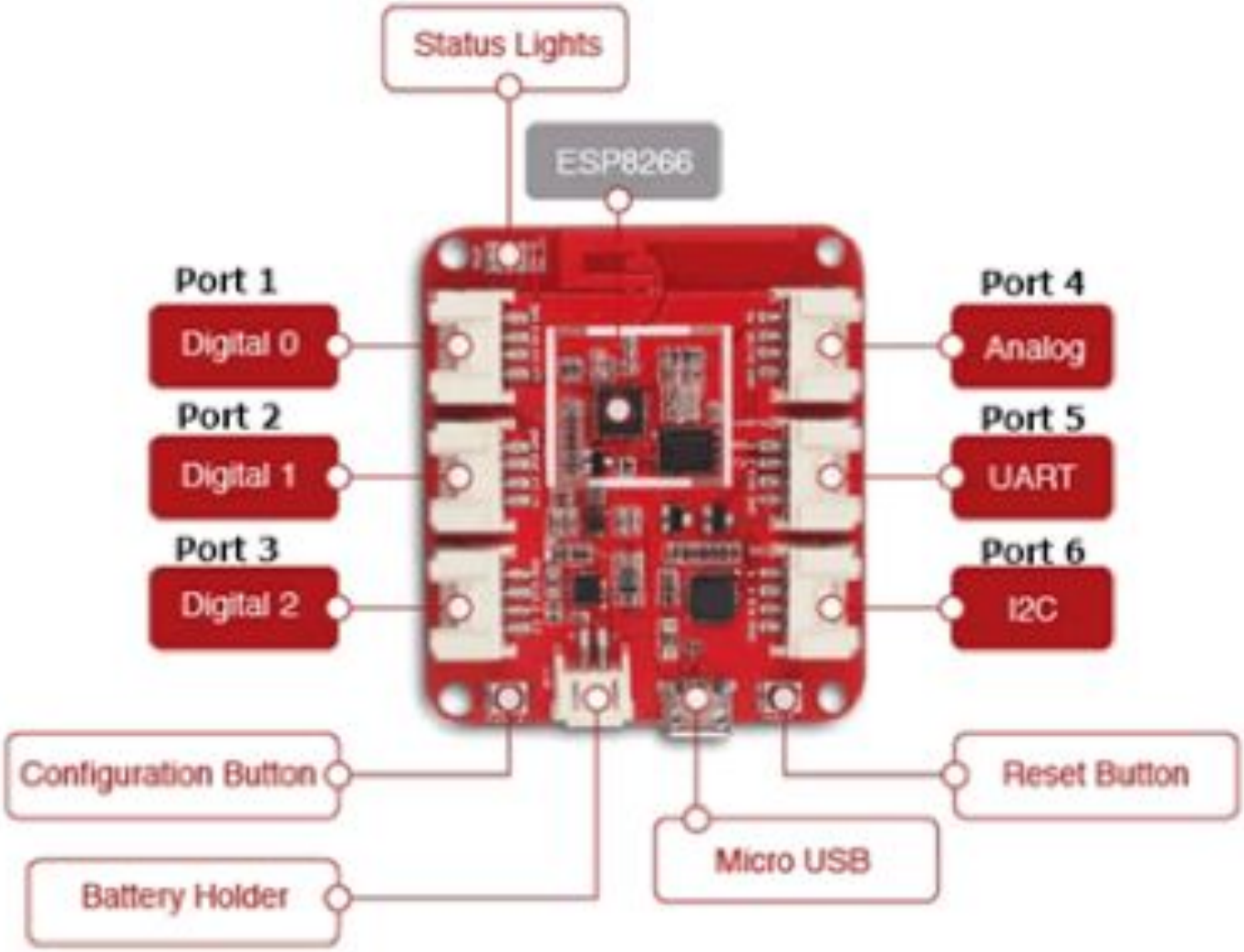


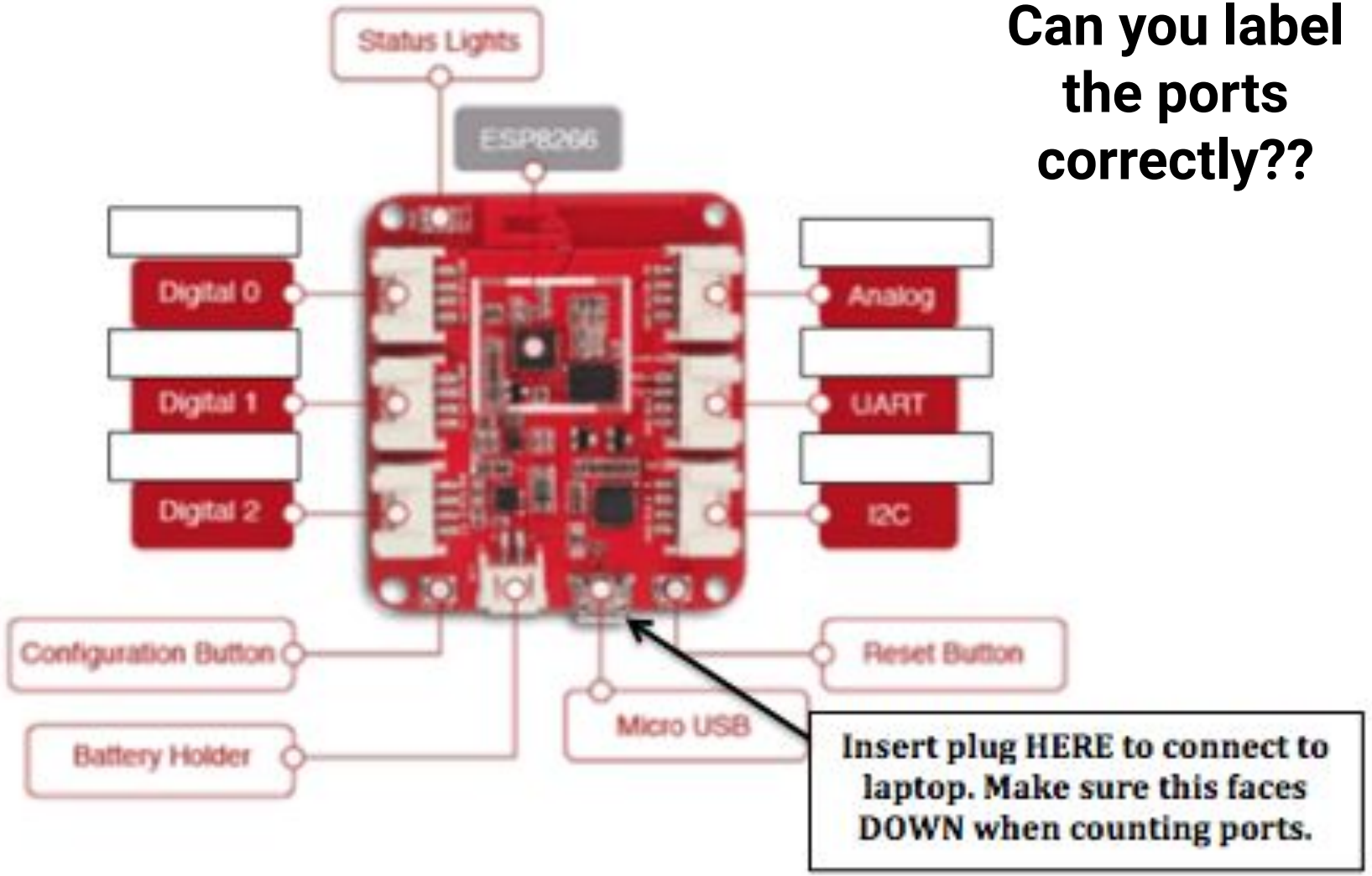
Table of Contents

Labeled MCU



"MCU" = Microcomputer

(Chip that you plug all sensors/devices in to!)



Can you label the ports correctly??

Devices for Greenhouse Project

Table of Contents

Image	Sensor/Device
	Relay
	Servo
	Temperature/Humidity
	Grow Light
	OLED Screen
	Light Sensor
	Moisture Sensor
	Button

Table of Contents

Variables you will control…

MCU 1

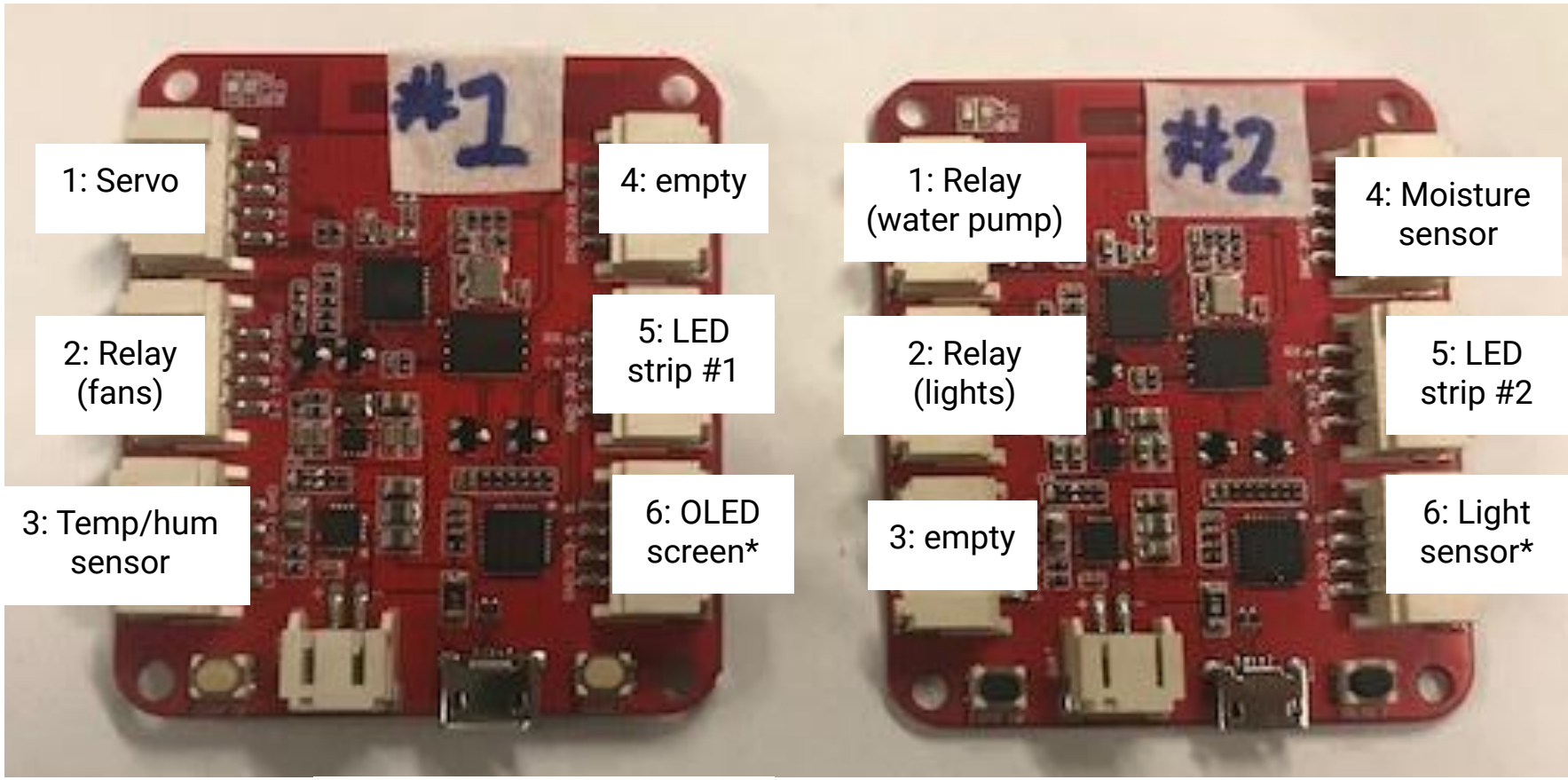
Variable	Device(s)
Air Temp.	*Temp & Humidity Sensor
Humidity	*Temp. & Humidity Sensor *Servo
Air Flow	*Relay → Propeller Fans → Exhaust Fans
Communication	*LED Lights *OLED screen

MCU 2

Variable	Device(s)
Moisture	*Moisture Sensor
Water	*Relay → Pump *Drippers/Hose
Light	*Light Sensor *Relay → Grow Lamps

Table of Contents

Devices on each MCU



* optional

Table of Contents

The Libraries!

The devices you will be using generally belong to **3 classes** (Libraries)

Actuators

↓

Do Things (move)

Devices: Relay (for lights, fans, water pump), Servo, Button

Sensors

↓

Detect/Sense Things

Devices: Light Sensor, Moisture Sensor, Temp. & Humidity Sensor

Displays

↓

Show Things

Devices: LED light, OLED screen

Table of Contents

Actuators: “Do things”

Device:	Port (s)	MCU#	What it does…
Relay 	Pump: 1 → 2 Lights: 2 → 2 Fans: 2 → 1		A switch that turns things on and off. Used for: fans, pump and lights
Servo 	1	1	Will control opening and shutting the roof of your greenhouse
Button 	1, 2, 3	N/A	Practice coding one device to influence another (button → LED light)

Table of Contents

Sensors: “Detect things”

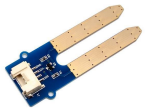
Device:	Port (s)	MCU	What it does…
Temp. & Humidity 	3	1	Detects Temperature and Humidity Levels as to keep them in a perfect range
Light Sensor 	6	2	Detects lux values (amount of light your plants are getting)
Moisture Sensor 	4	2	Detects moisture values (how dry or wet your plants soil is)

Table of Contents

Displays: "Show Things"

Device:	Port (s)	MCU	What it does...
OLED Screen	6	1, 2	Shows Temperature, Moisture and Humidity Levels within a certain time period
Grow Lamps	2	2	Shows if the lights are on or off (either giving or not giving your plants heat/sunlight)
LED Lights	5	1, 2	Shows the conditions of your greenhouse. Example: Green= Good → Plants are happy Red= Bad → Plants are NOT happy!

Activity Suggestion

- Print out blank versions of the MCU & charts with devices.
- Label the blank copies OR print out the words separately
- Have students time themselves in pairs to see how quickly they can correctly label each MCU/library/device!

Table of Contents

Exit Ticket (Lesson 1c)

1. What does an MCU do?

2. What are the 3 classes of devices?

3. Which device do you think is the MOST important to keeping your plants healthy? Why?

Table of Contents

Lesson 2a: LED Lights

Table of Contents

Lesson 2a: LED Lights

Image	Sensor/Device	Class	Ports
	Grow Light	displays	5

Table of Contents

Do Now (Lesson 2a):

1. Identify and explain 3 reasons why you might want to put LED lights on your greenhouse

Class Share-Out:

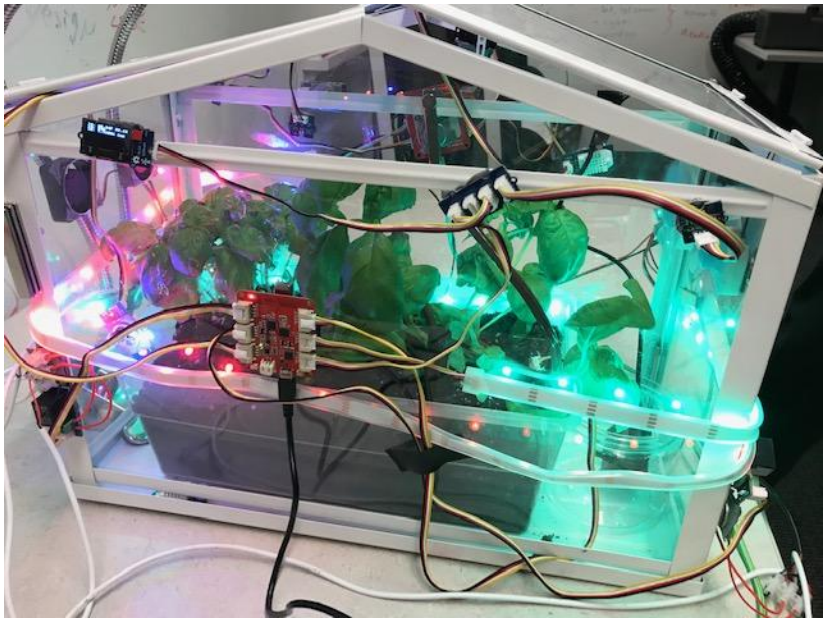


Table of Contents

Problem of the day:

How can I determine the general environmental conditions (“happiness”) of my greenhouse without plugging in devices & downloading data?

Solution: Program an LED light to display different colors depending on the condition of the greenhouse!

Table of Contents

Problem-Solving Process

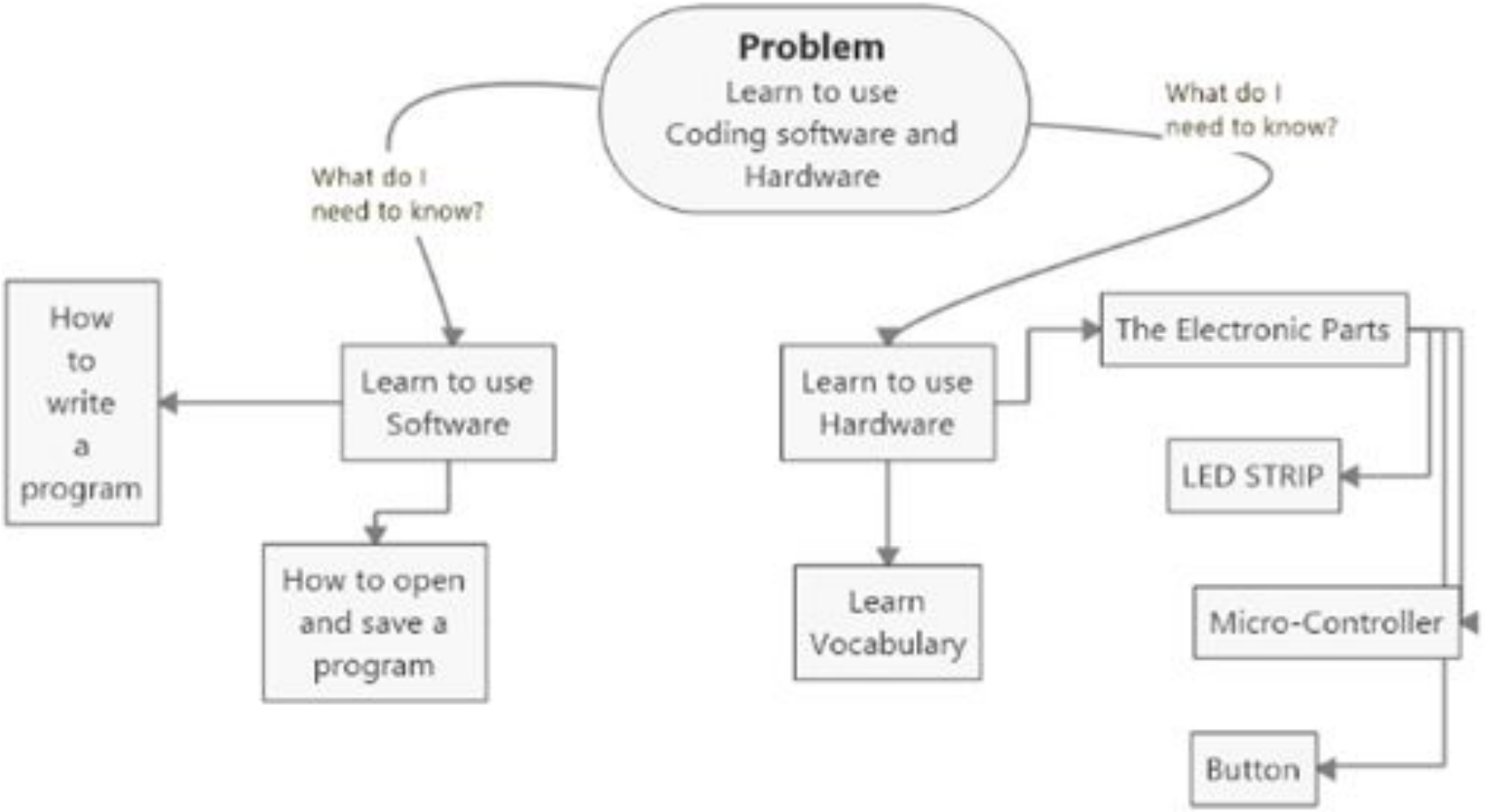


Table of Contents

Brainstorm: LED light

- What *color* might we want the LED light to blink if the greenhouse is **too hot?**
- What *color* might we want the LED light to blink if the greenhouse has a **low moisture level**?
- What could the color **orange** represent in terms of **humidity**?

Table of Contents

General format for adding new device

1) Two students work on writing the code needed to program the device
2) Two other students work on the device itself (gathering supplies, wiring, attaching to MCU, etc.)

3) Help each other as needed!
4) TODAY & TOMORROW ONLY… **each pair will code AND use the device**

Table of Contents

Receiving your group's MCUs

1. Each group will receive two blank MCU chips.
2. You are to cut a small piece of tape that will fit at the top of each MCU.
3. Label them #1 and #2.
4. Place #2 in your group's ziplock bag, and attach to your greenhouse.

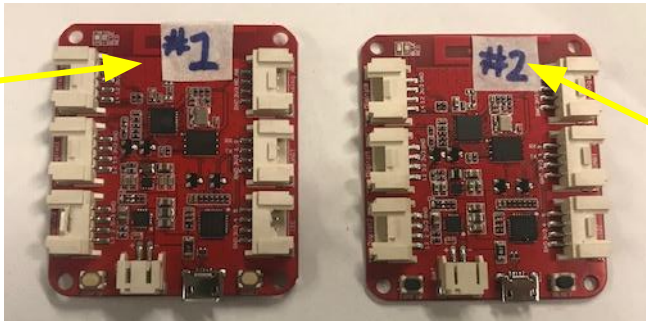


Table of Contents

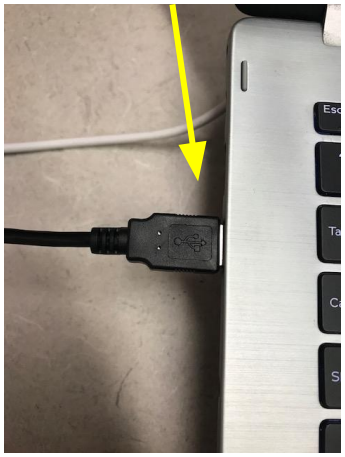
Gather Materials
for Lesson 2a: LED light strip

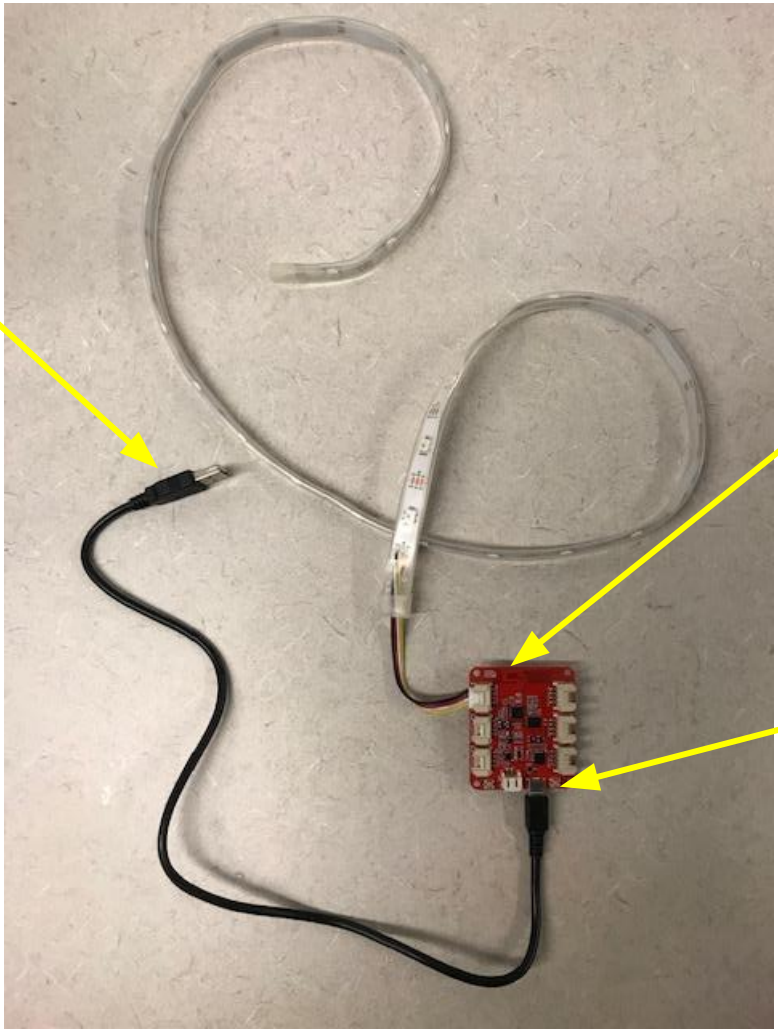
Name & # of Part	Picture of Part
1 - Wio Link Board (Micro-Controller Unit). This is known as the “MCU”	
1- Alligator Clip (To attach LED strip to greenhouse)	
1- Micro-USB Cord	
1 – LED Light strip (Sometimes called a GrowLight)	

Table of Contents

Connect as follows…

This end plugs into the computer





LED strip plugs into Port 1

micro USB plugs into MCU, facing down

Table of Contents

Once connected to the computer, the **red light** on your MCU #2 should turn on!

Now you are ready to code…

*If you see a **blue light**, reflash your MCU. If no luck, get another MCU and re-label it.*

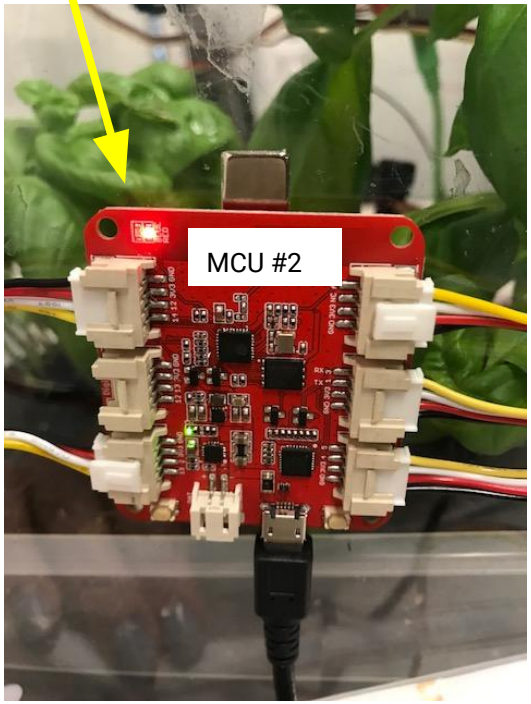


Table of Contents

How to use EsPy

(the program that runs microPython)

Step 1: Click the "EsPy- Shortcut" icon on your desktop, which will open the program.

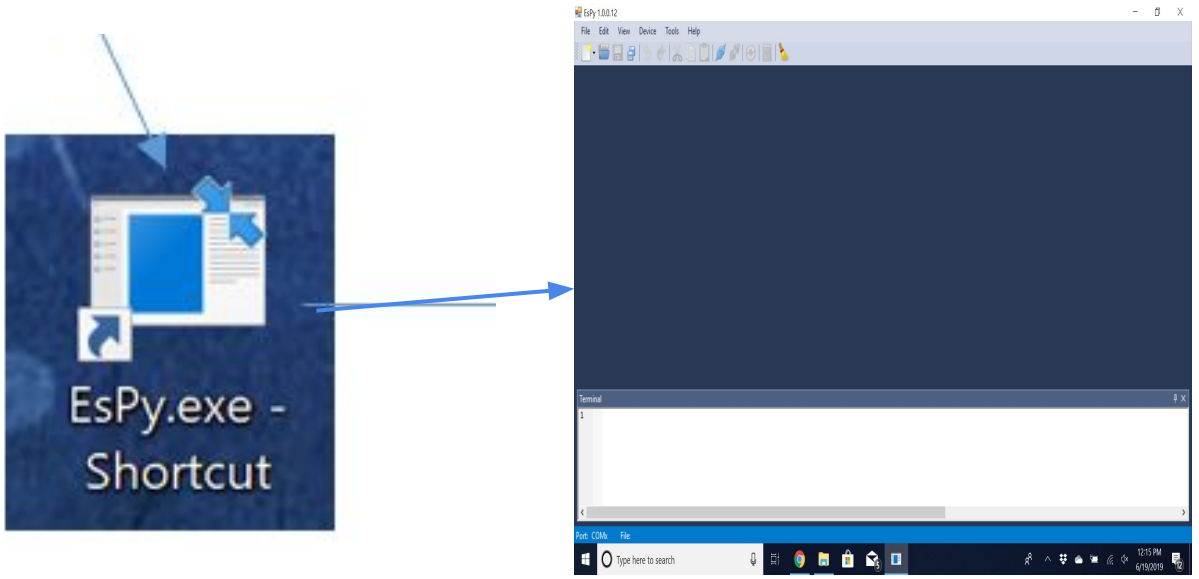


Table of Contents

How to use EsPy
(the program that runs microPython)

Step 2: Go to File → New → Python

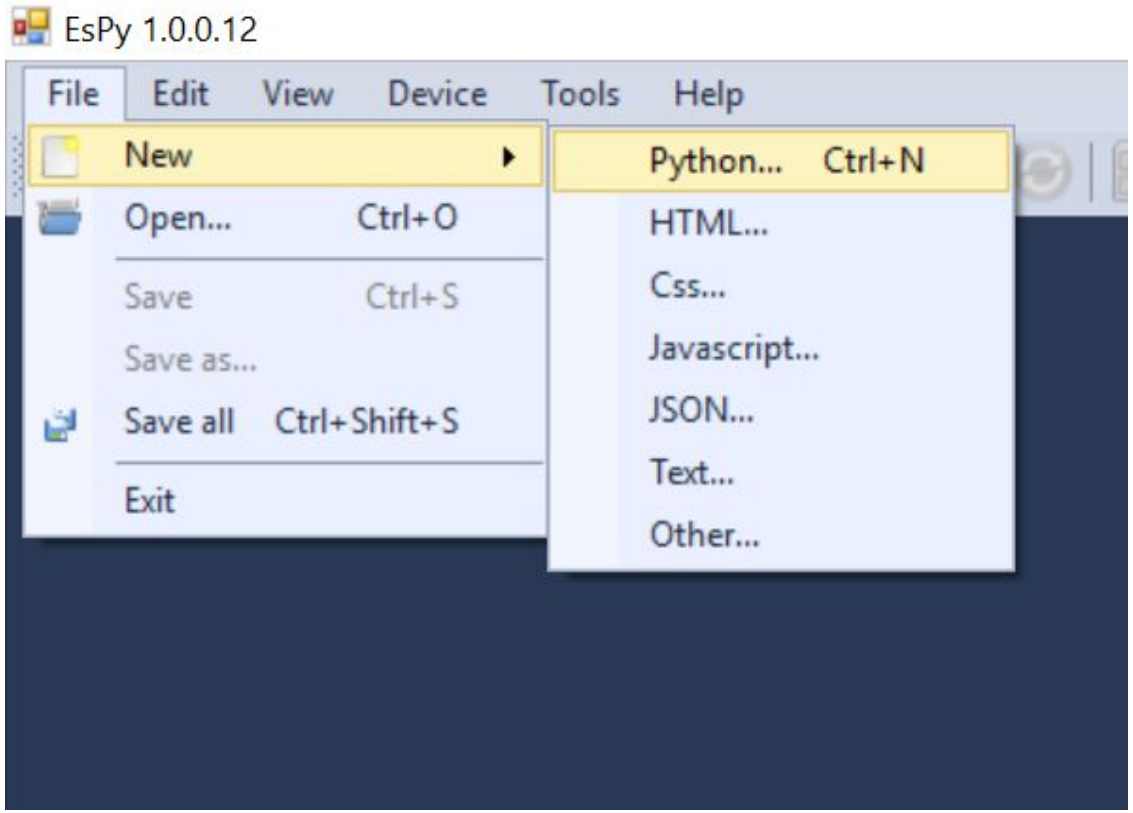


Table of Contents

How to use EsPy
(the program that runs microPython)

<u>Step 3:</u> Save your file by DOUBLE-clicking on your assigned folder (Ex. Blue 1A)

Double click to enter the folder!

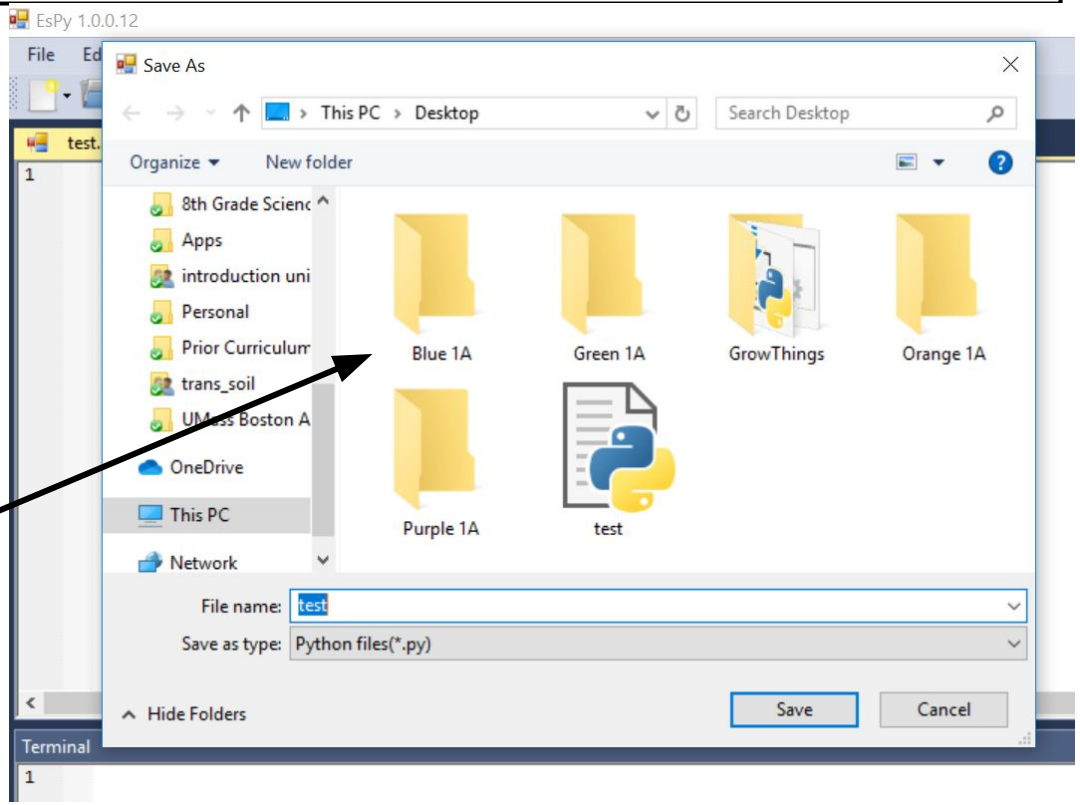
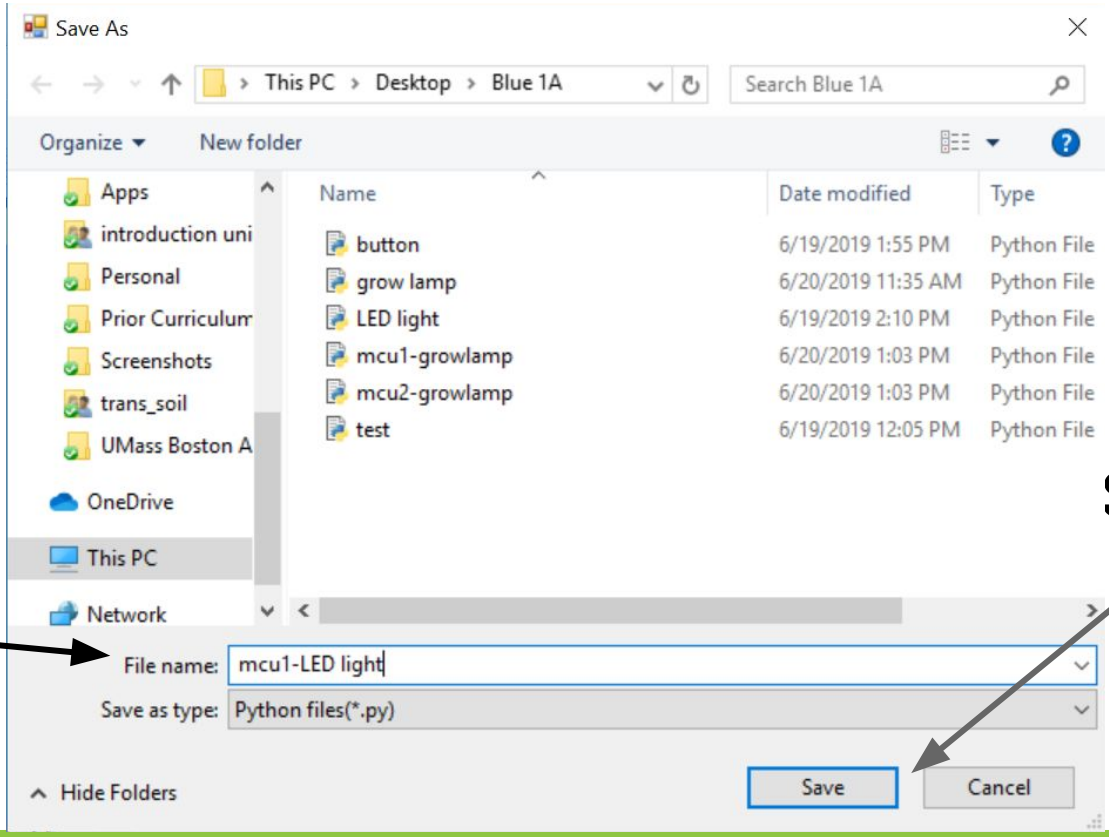


Table of Contents

How to use EsPy

(the program that runs microPython)

Step 4: Once inside the folder, name the file based on the device you are using (ex. MCU1-LED light) and press “Save”



Type the name of the device here. Ex. “MCU1- LED light”

Press Save”

Table of Contents

How to use EsPy
(the program that runs microPython)

<u>Step 5:</u> Type your code!
(You will have to open another tab or use another computer to view the code while typing)

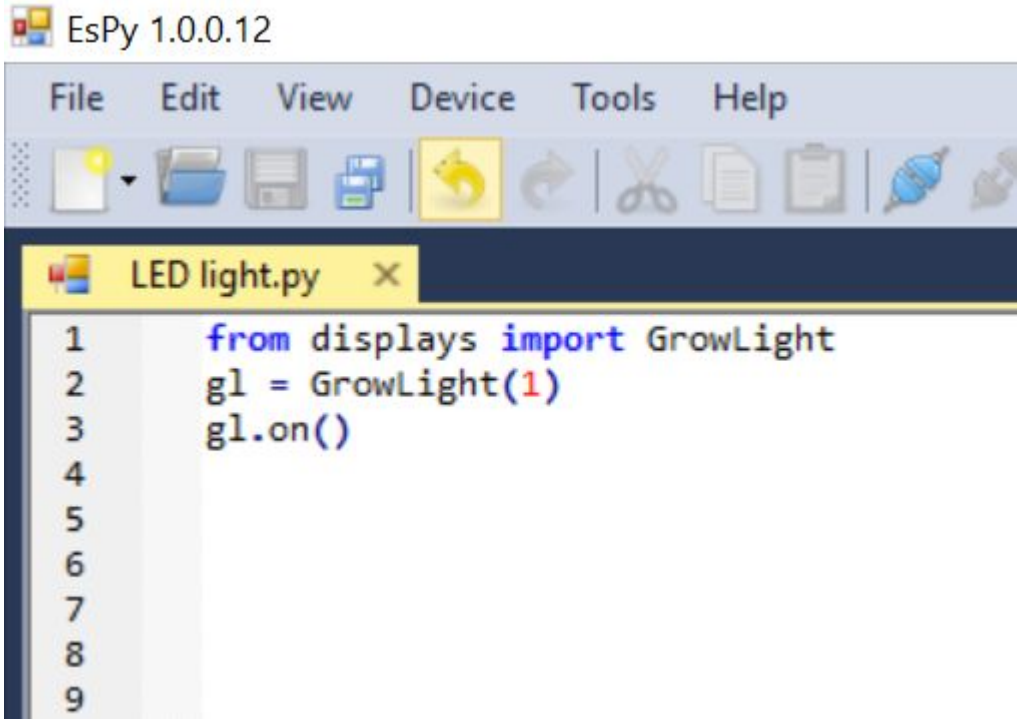


Table of Contents

Code for LED Light Strip

EsPy 1.0.0.12

File Edit View Device Tools Help

LED light.py

```
1  from displays import GrowLight
2  gl = GrowLight(1)
3  gl.on()
4
5
```

Table of Contents

How to use EsPy
(the program that runs microPython)

<u>Step 6:</u> Press "Connect" then "Run" (play) to run your code!

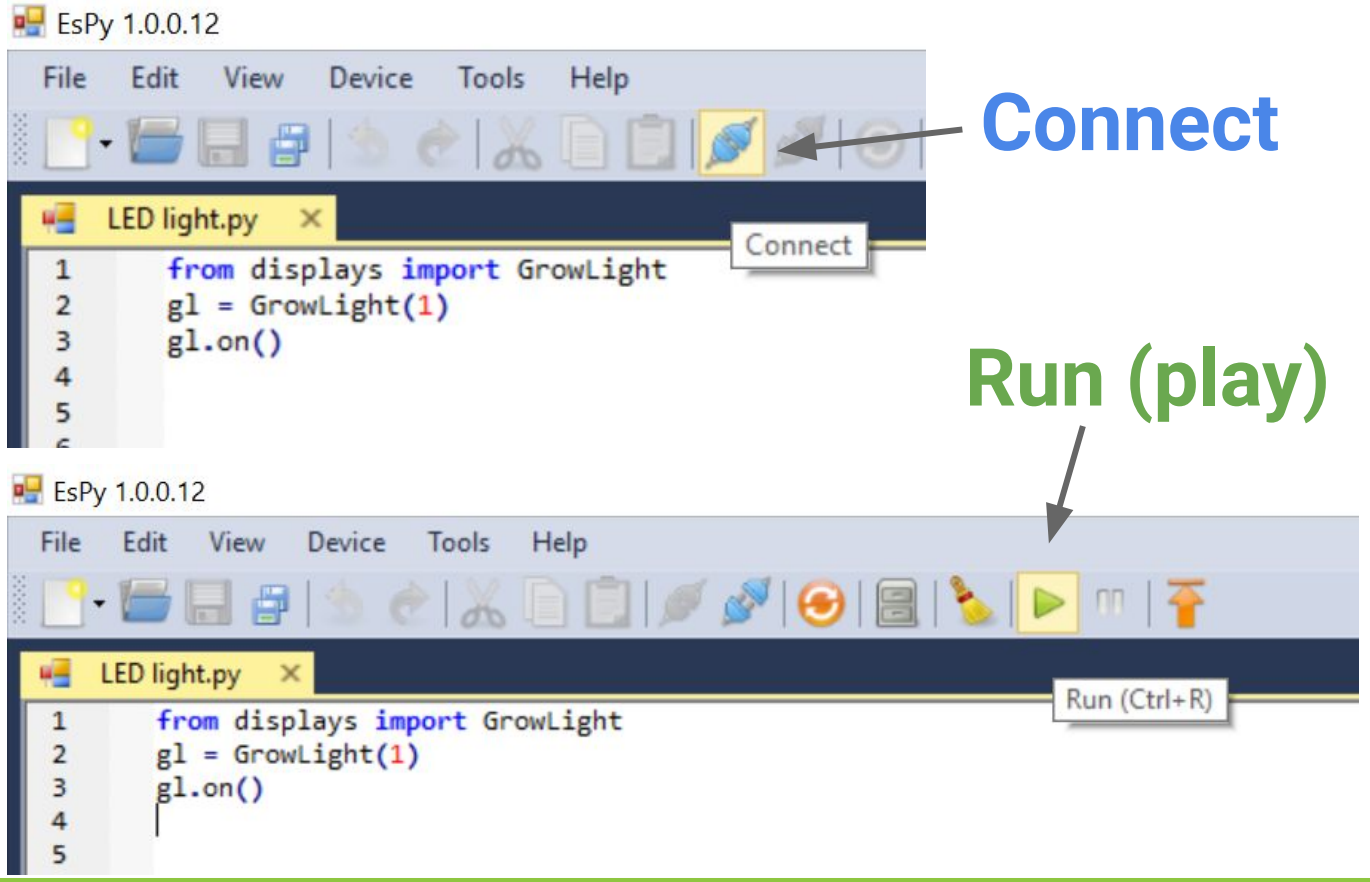


Table of Contents

Not working? Check your terminal for any errors..

The bottom of the screen provides information. Notice here, the terminal is stating there is an error is in **line 1**. The person forgot to capitalize the “L” in “GrowLight”

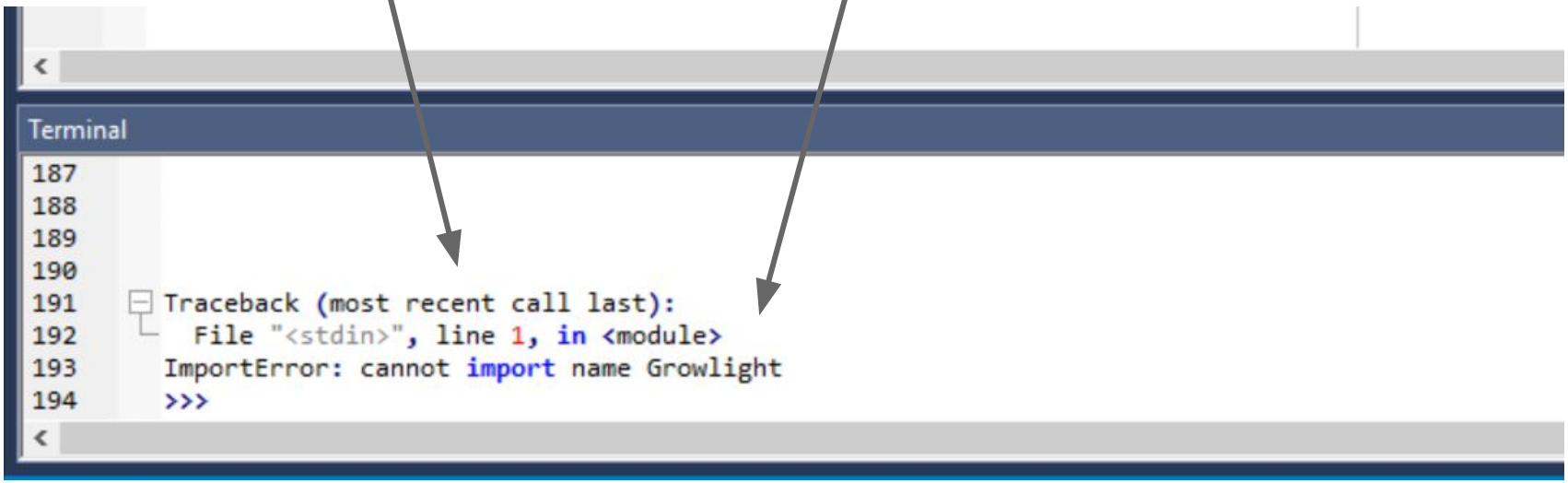


Table of Contents

Fix errors & test again

Click “Disconnect”, then “Connect” and “Run” again.

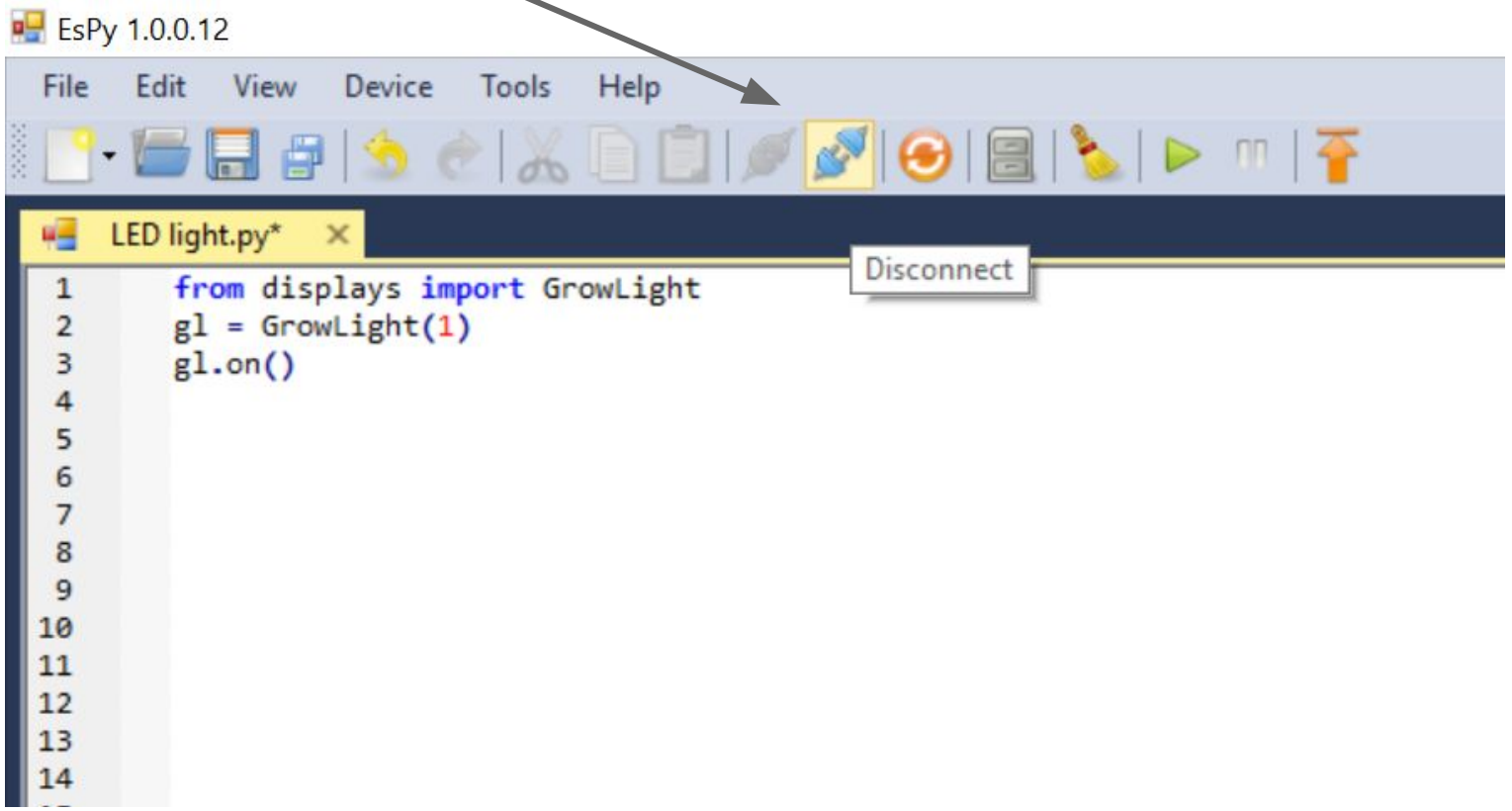
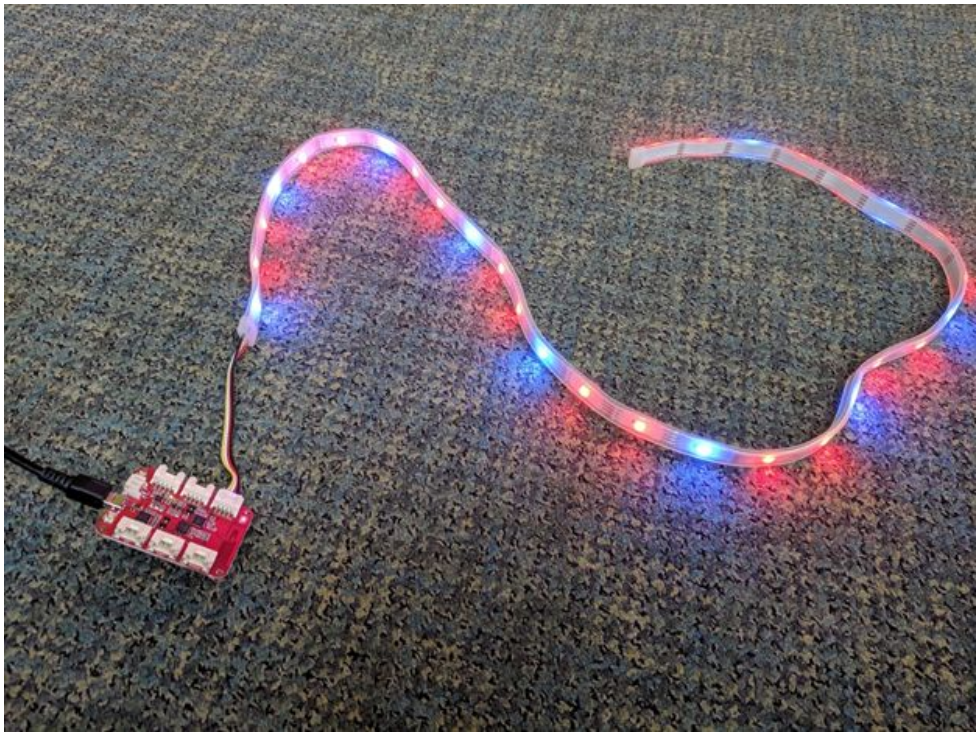


Table of Contents

Is it working??

If so, your LED strip should look like this!



If it's working, congrats!
Try these LED Light Challenges…

Table of Contents

Challenges: LED Light

Challenge #1: Replace **gl.on()** with **gl.blink().** What happened?

Your code should look like this:

LED light.py*

```
1    from displays import GrowLight
2    gl = GrowLight(1)
3    gl.blink()
4
```

Table of Contents

Challenges: LED Light

Challenge #2: Insert a color next to gl.blink.

Your code should look like this:

LED light.py

```
1    from displays import GrowLight
2    gl = GrowLight(1)
3    gl.blink(color=[255,0,0])
4
```

Table of Contents

Challenges: LED Light

Challenge #3: Change the numbers in the color interval so that it reflects the code below.
What color does this represent?

```
1    from displays import GrowLight
2    gl = GrowLight(1)
3    gl.blink(color=[0,255,0])
4
```

What about this one?

```
1    from displays import GrowLight
2    gl = GrowLight(1)
3    gl.blink(color=[0,0,255])
4
```

Try a combination of numbers 0-255, what colors can you make??

Table of Contents

Challenges: LED Light

Challenge #4: Add a **times** and **interval** for the blinking. Start with the code below, then change it to play with different options.

Your initial code should look like this:

```
1    from displays import GrowLight
2    gl = GrowLight(1)
3    gl.blink(color=[0,0,255], times=2, interval=.5)
4
```

Challenge #5: Try to have the LED light blink **blue** 5 times, then **red** 5 times, then **green** 5 times.
(*Hint*: Copy and paste the last line (gl.blink…) 3 times and change the numbers in each!)

Table of Contents

Exit Ticket

1. Did you get your LED lights working? If not how come?
2. Did you change the LED light colors?

Table of Contents

Lesson 2b: Button

Table of Contents

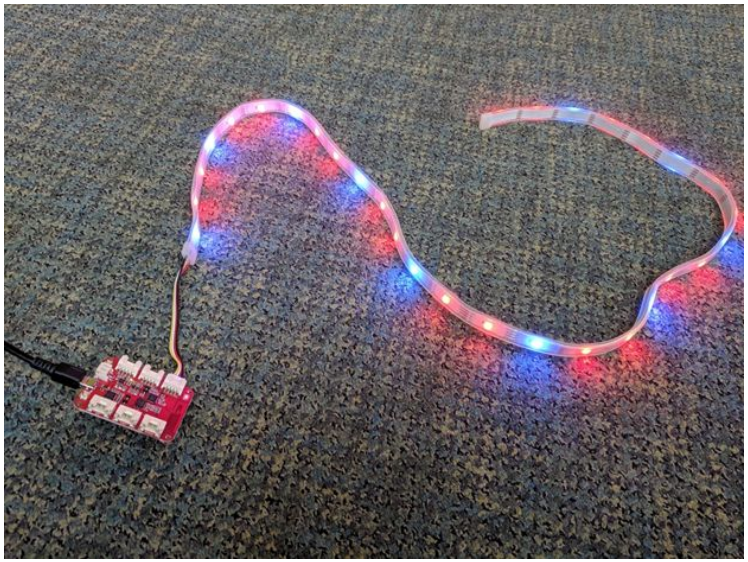
Lesson 2b: Button

Image	Sensor/Device	Class	Ports
	Button	Actuators	1,2,3

Table of Contents

Do Now (Lesson 2b):

1. Write the three lines of code from memory that are required to turn on the LED strip.
2. What do each of those lines of code mean?



Class Share-Out:

Table of Contents

Reviewing Yesterday's Code

```
1  from displays import GrowLight
2  gl = GrowLight(1)
3  gl.on()
```

1) Which library is used?
2) Which device is used?
3) What is the code/nickname for GrowLight (LED strip)?
4) What port is the GrowLight (LED strip) plugged in to?

Table of Contents

Problem of the day:

How do you turn on your LED light strip without pressing “connect” and “disconnect” on the computer?

Solution: Program a button to do it for you!!!

Table of Contents

Problem-Solving Process

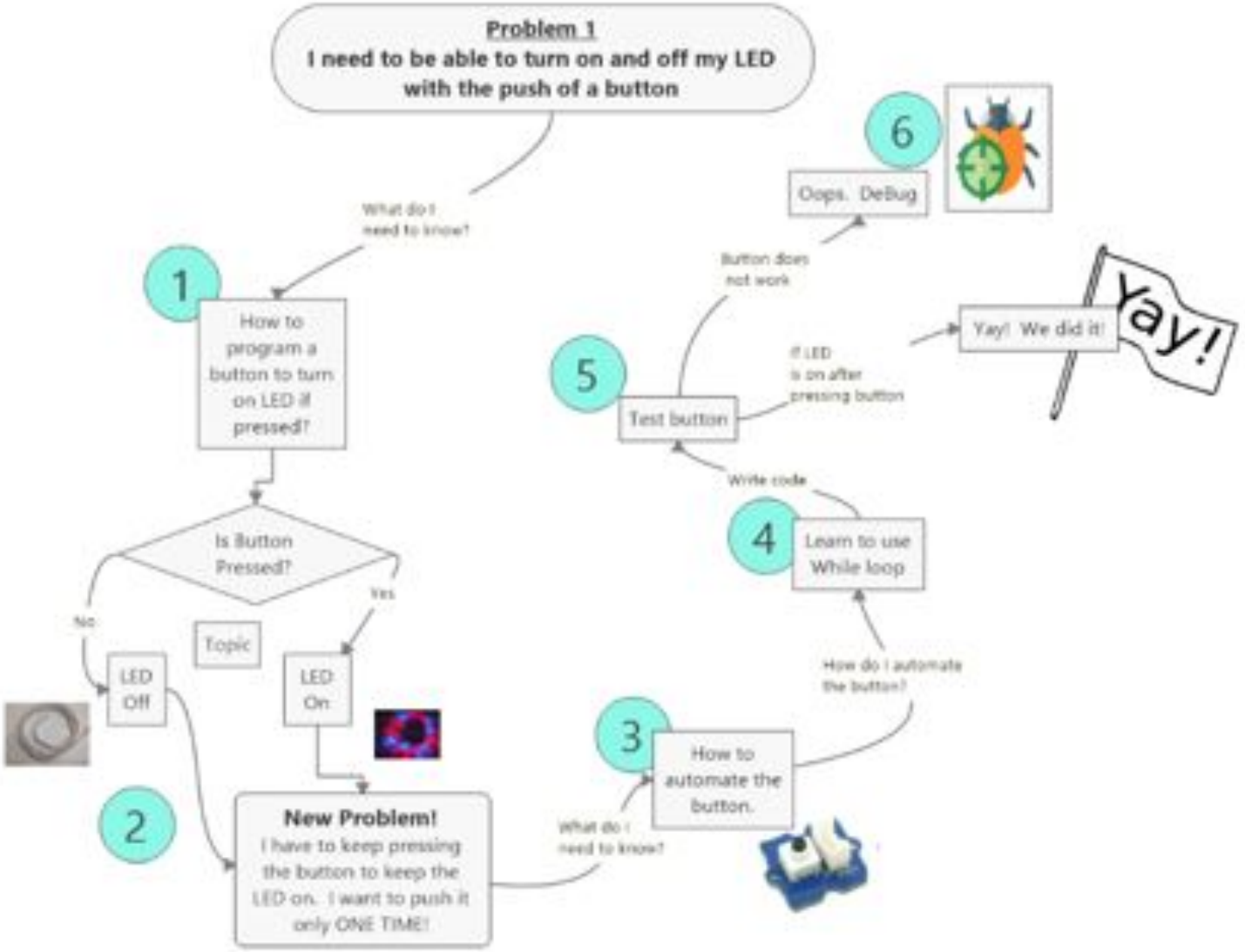


Table of Contents

Brainstorm: Button

- How do you **connect the LED lights** to the button?
- What do we need to **add/change to the code** so it includes the button?
- What if we want to press the button to turn it **off**? How might we **reverse** the action?

Table of Contents

Group Format Suggestion

1.) The button activity can be completed in pairs instead of as a full group.

2.) One partner can gather the materials and connect the button and LED light to the MCU and computer.

3.) The other partner can open to the “Code for Button” slide from the Table of Contents and type the code into a new EsPy file.

4.) Test out button & then attempt challenges!

Table of Contents

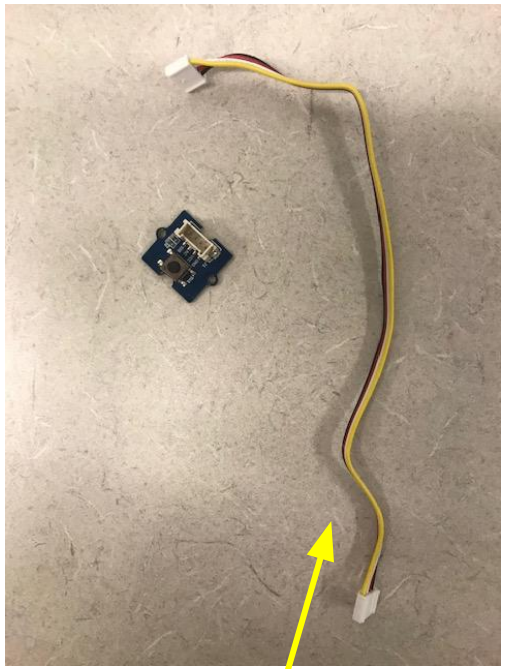
Gather Materials
for Lesson 2b: Button

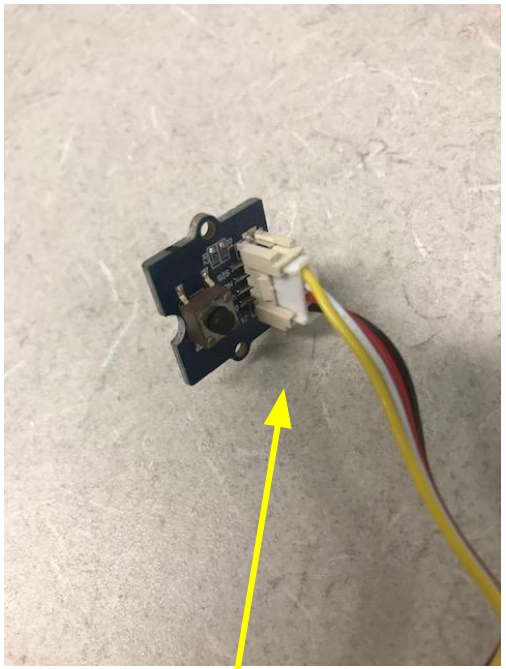
Name & # of Part	Picture of Part
1 - Wio Link Board (MCU #1)	
1- Micro-USB Cord	
1 – LED Light strip	
1- Button	
1- 4 pin connector wire	

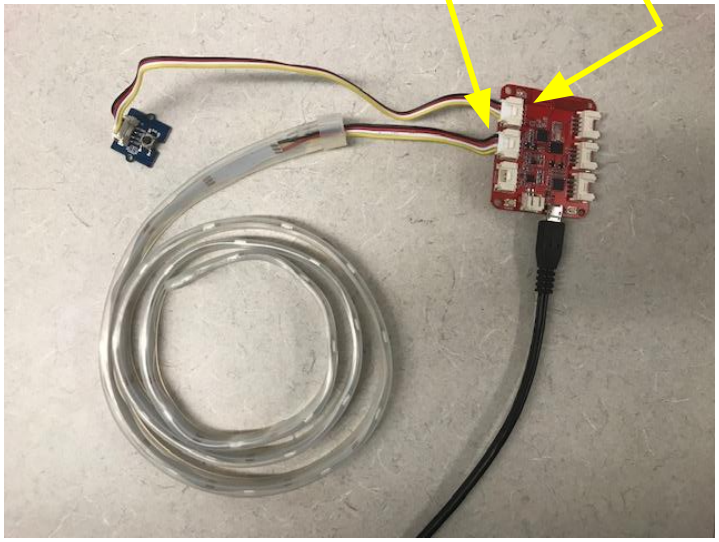
Table of Contents

Connect as follows…

Plug the button into Port 1
and the LED strip into Port 2







Grab a connector wire and attach one end to the button

Plug the other end of the USB into the computer

Table of Contents

Once connected to the computer, the **red light** on your MCU #2 should turn on!

Now you are ready to code…

*If you see a **blue light**, reflash your MCU. If no luck, get another MCU and re-label it.*

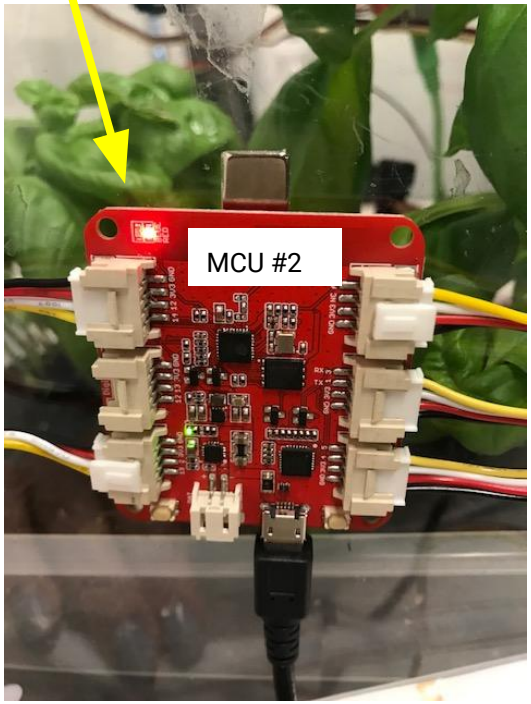
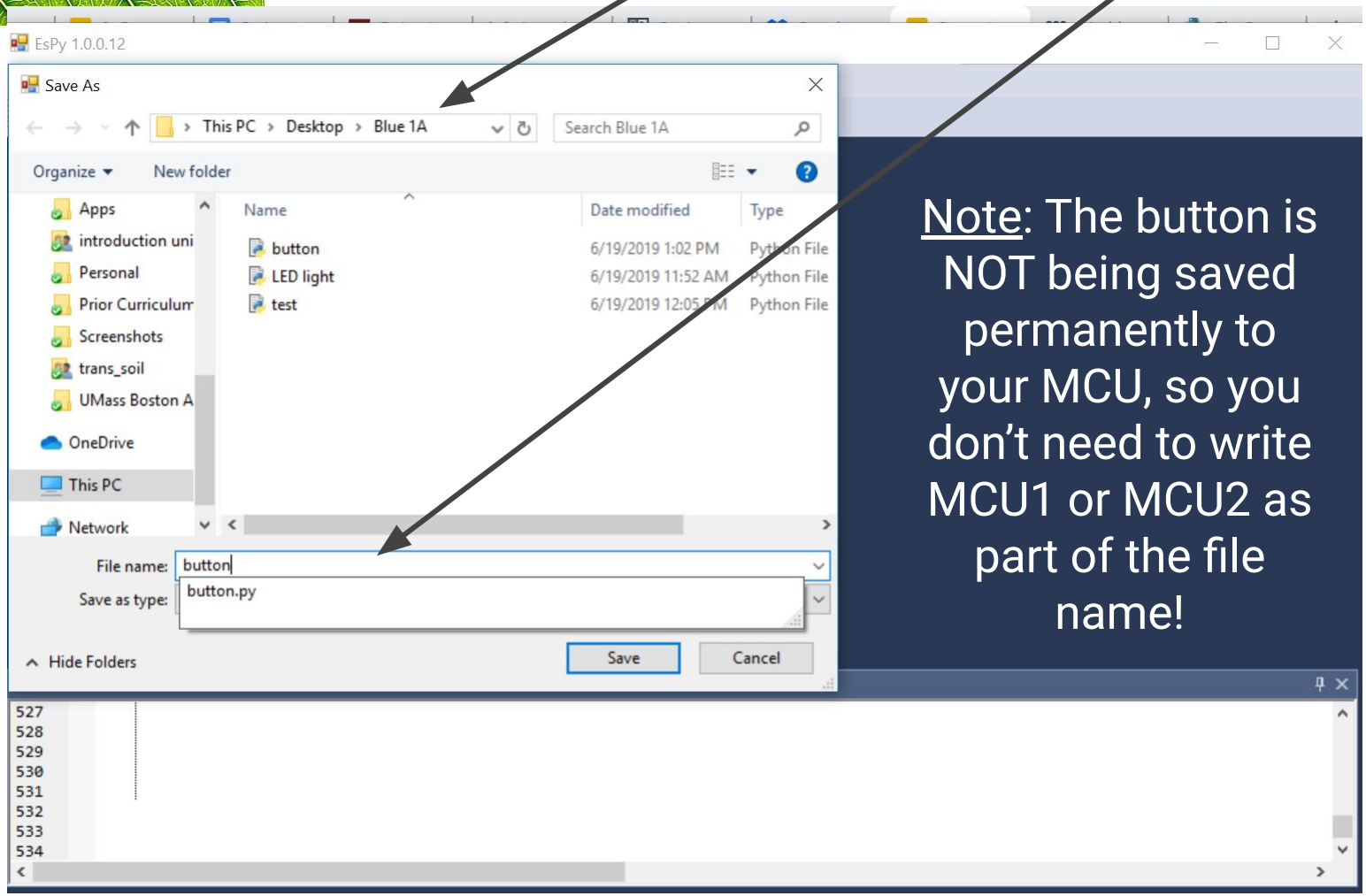


Table of Contents

Open a new EsPy file and save in your class folder as “Button”



Note: The button is NOT being saved permanently to your MCU, so you don’t need to write MCU1 or MCU2 as part of the file name!

Table of Contents

Code for Button

These lines will show up automatically when you press “Tab”

Make sure that you press “Tab” to indent the line, and DO NOT use the spacebar!!

EsPy 1.0.0.12

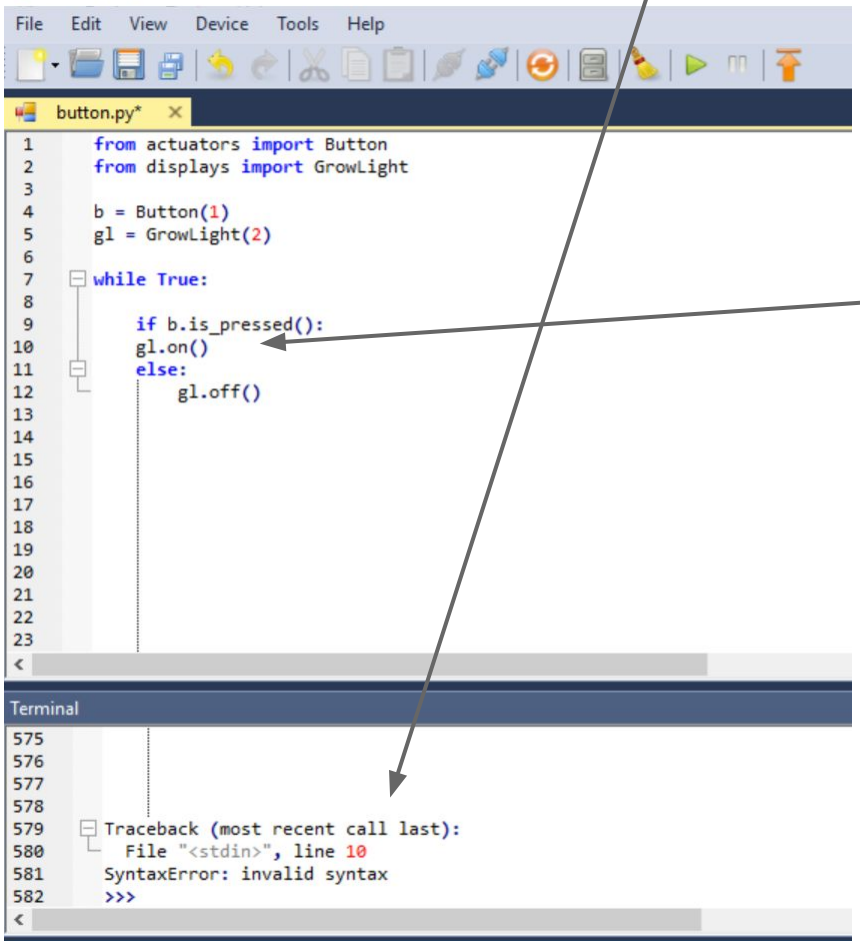
File Edit View Device Tools Help

button.py*

```
1   from actuators import Button
2   from displays import GrowLight
3
4   b = Button(1)
5   gl = GrowLight(2)
6
7   while True:
8
9       if b.is_pressed():
10          gl.on()
11      else:
12          gl.off()
13
14
15
16
```

Table of Contents

Not working? Check your terminal for any errors..



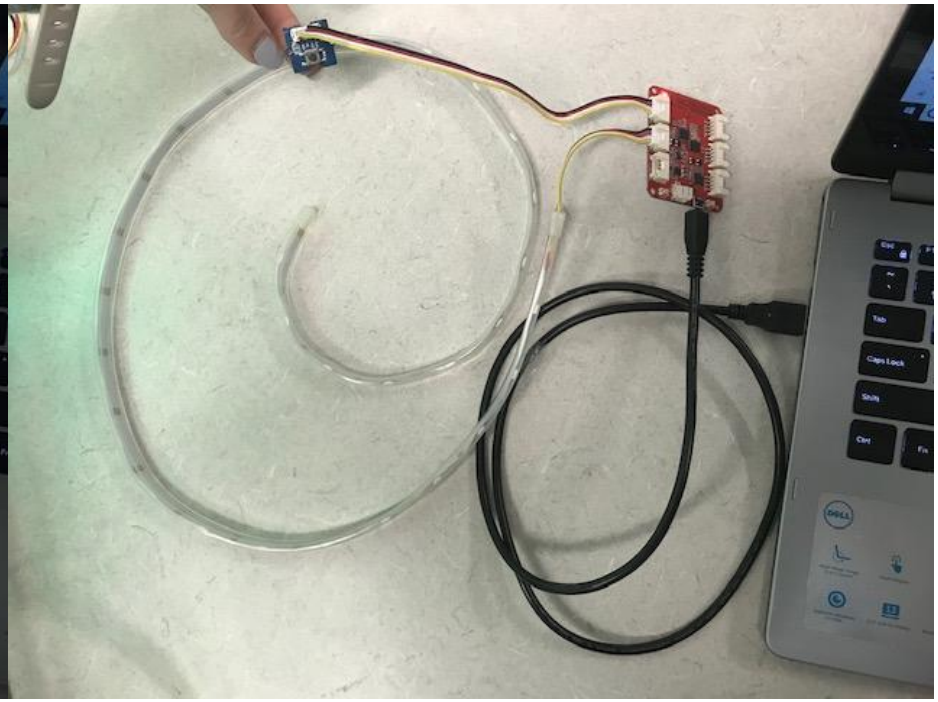
In this example, the indenting is wrong. The student forgot to press “Tab” before **gl.on()**

Table of Contents

Is it working??

If so, move on to the challenges!

Button pressed



Button released

Table of Contents

Challenges: Button

Challenge #1: Reverse it! Change your code so the LED light is ON when the button is NOT pressed and OFF when the button IS pressed.

Challenge #2: Change your code so the LED light blinks blue when the button is pressed and blinks red when the button is NOT pressed. *(HINT: Remember what you learned in the last lesson! You can go back if you need.*

Challenge #3: Change your code so when you press your button, your light blinks red/blue quickly like a police car for a total of 20 continuous seconds.
(Hint: You will have to copy and paste!)

Table of Contents

Exit Ticket (Lesson 2b)

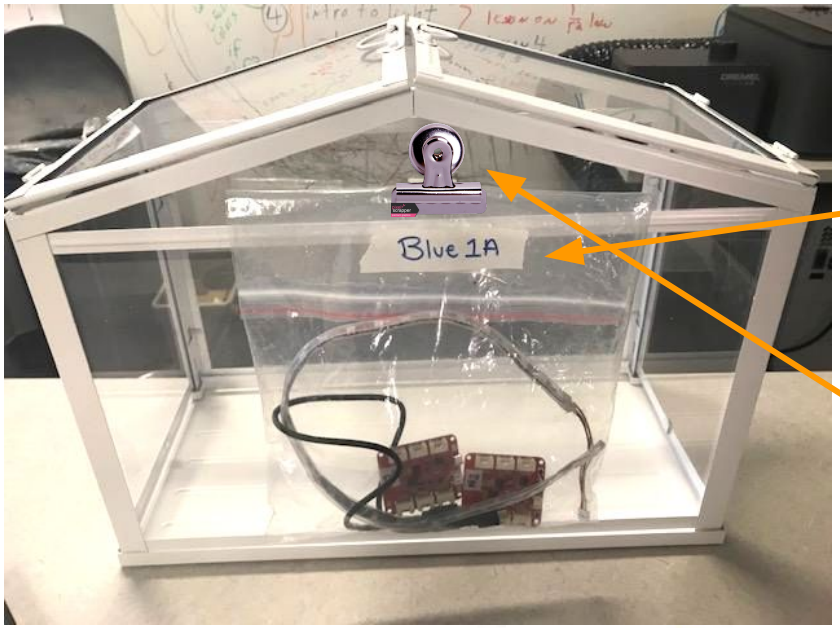
1. What colors would you like to use to represent different conditions in your greenhouse?

2. Why are we NOT keeping the button as part of our greenhouse?

Table of Contents

Clean Up!

1. Unplug the 3 pieces and place them in your group's ziplock bag (along with MCU #2)



2. Make sure your ziplock is labeled with your group name/number!

3. Attach your bag to your greenhouse using a magnetic alligator clip.

Table of Contents

Lesson 3a: Grow Lamps

Table of Contents

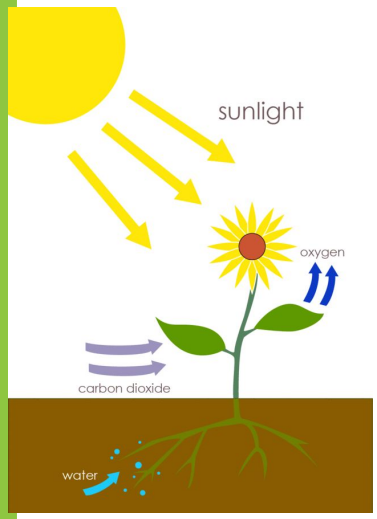
Lesson 3a: Grow Lamps

Images	Sensor/Device	Class	Ports
 	Relay Grow Lamps	Actuators Displays	1, 2

Table of Contents

Do Now (Lesson 3a):

1. Why do plants need sunlight?
2. What can we add to our greenhouse to represent sunlight?
3. When should they go on or off?



Class Share-out

Table of Contents

Problem of the day:

How do we connect and turn on/off our Grow Lamps?

Solution: We can use a Relay (essentially an on/off switch)

Table of Contents

Problem-Solving Process

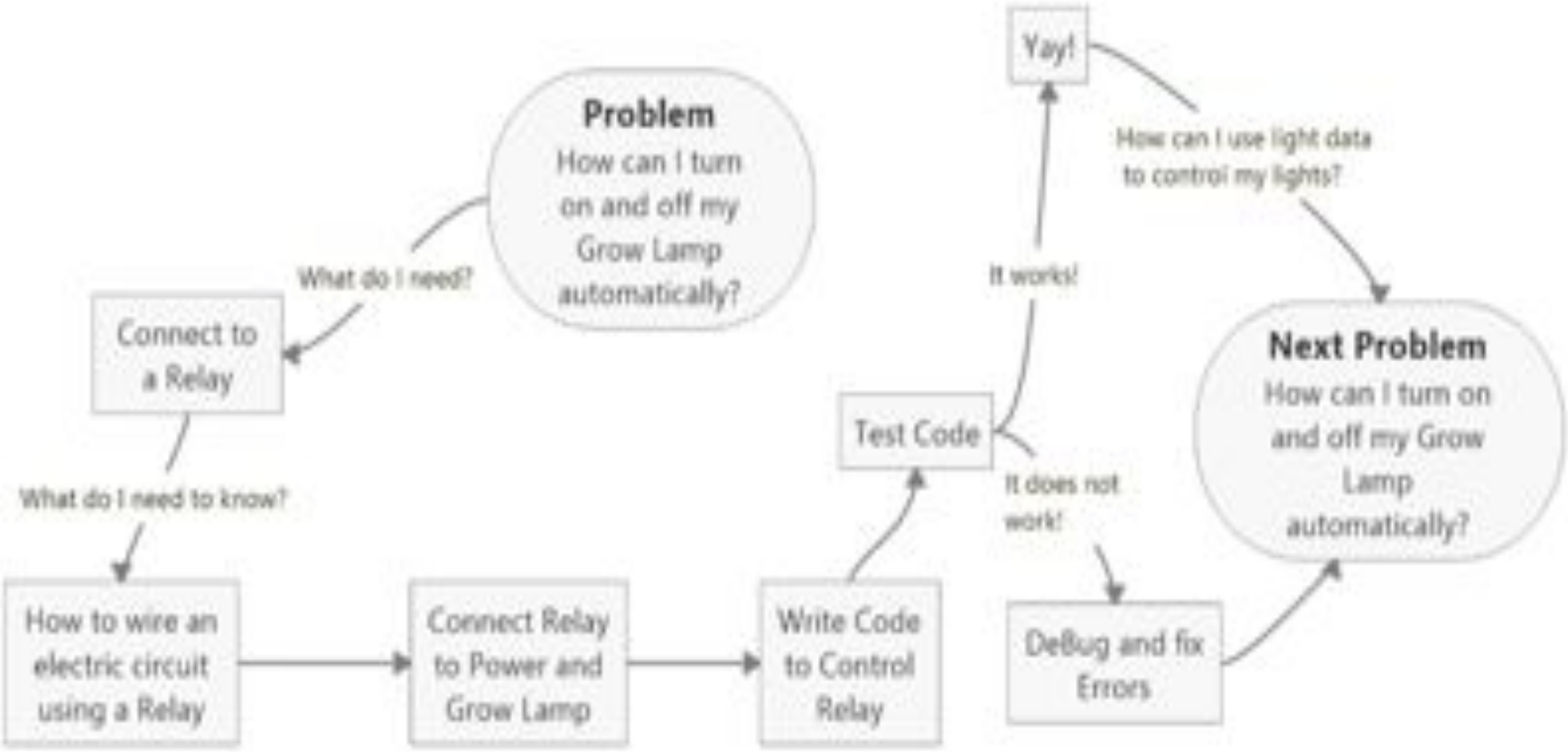


Table of Contents

Brainstorm:

- How do we connect the lamps to to our greenhouse?
- How do we turn them on?
- What will the Relay do?
- How can we give our plants light when we are away? timer?

Table of Contents

Group Format Suggestion

1.) The Grow Lamp lesson can be completed as a group of 4.

2.) 2 group members will be in charge of gathering materials and putting the devices together.

3.) 2 group members will be in charge of coding

Reminder: The pair coding also must open one computer to these slides so the coder can see the directions/code itself!

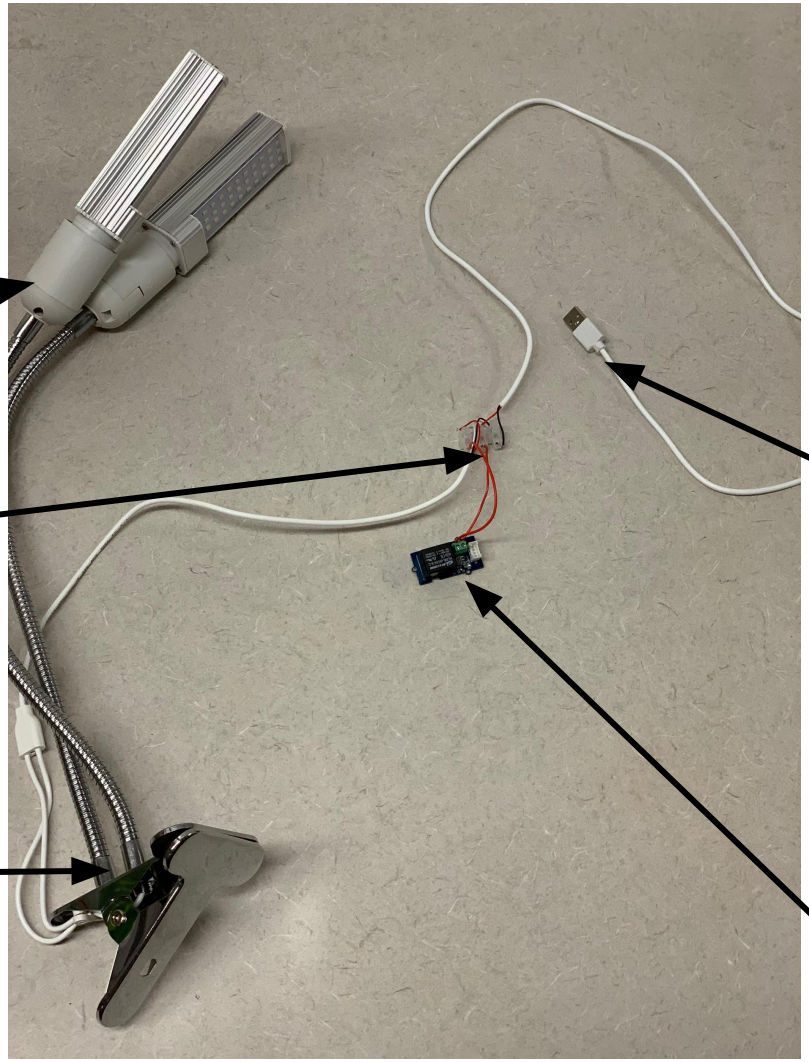
Table of Contents

Gather Materials
for Lesson 3a: Relay & Grow Lamp

Number/Name of Part	Picture of Part	Number/Name of Part	Picture of Part
1- Wio Link Board (MCU).	MCU #2 	1- Grow Lamp	
1- Micro-USB Cord (to plug in the MCU)		1- Micro-USB Cord Stripped (wires exposed)	
1- Relay with 2 red wires		1- Screwdriver	
1- terminal block		1- 4 Pin Wire Connector	
May need a pair of wire strippers (unlikely)			

Setup 1

This is your Lamp Setup!



Grow Lamps

Terminal Block: Where all the wires get connected

Clip to clamp onto base of greenhouse

Power Source Will get plugged into your USB power hub!

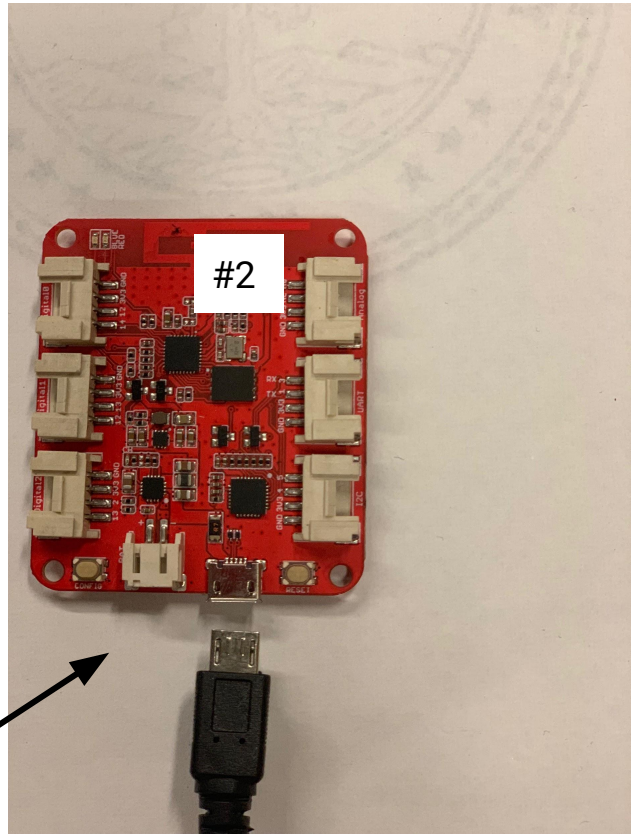
Relay!

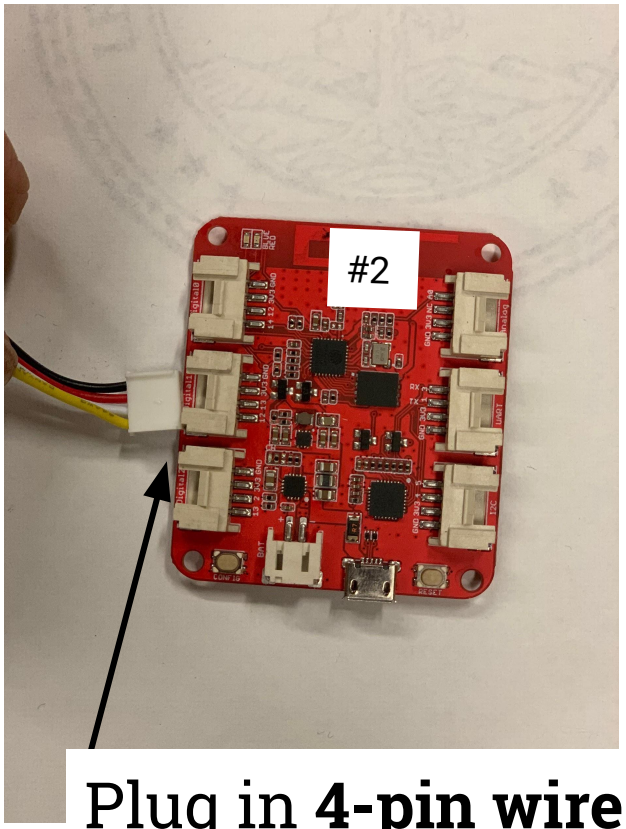
Table of Contents

Setup 2

Assemble your MCU #2 and related wires (USB & 4-pin cable)

Plug in **black USB cable**



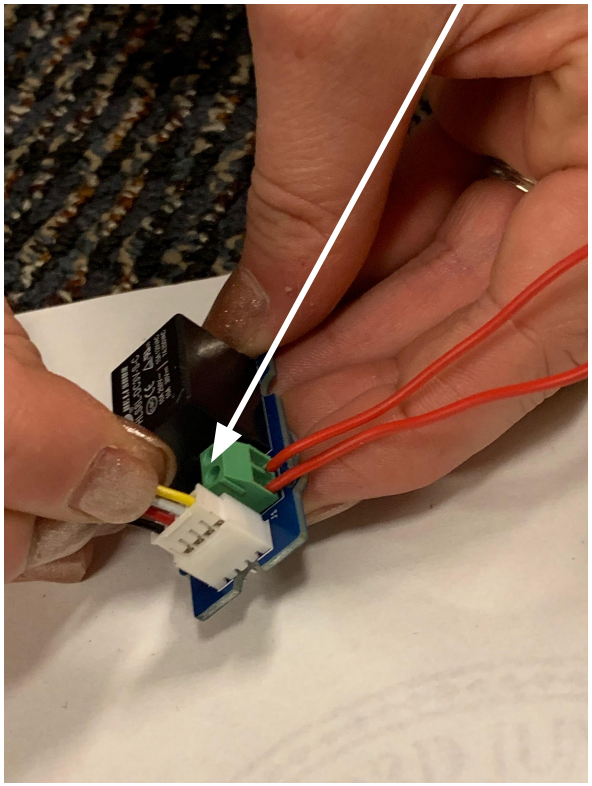


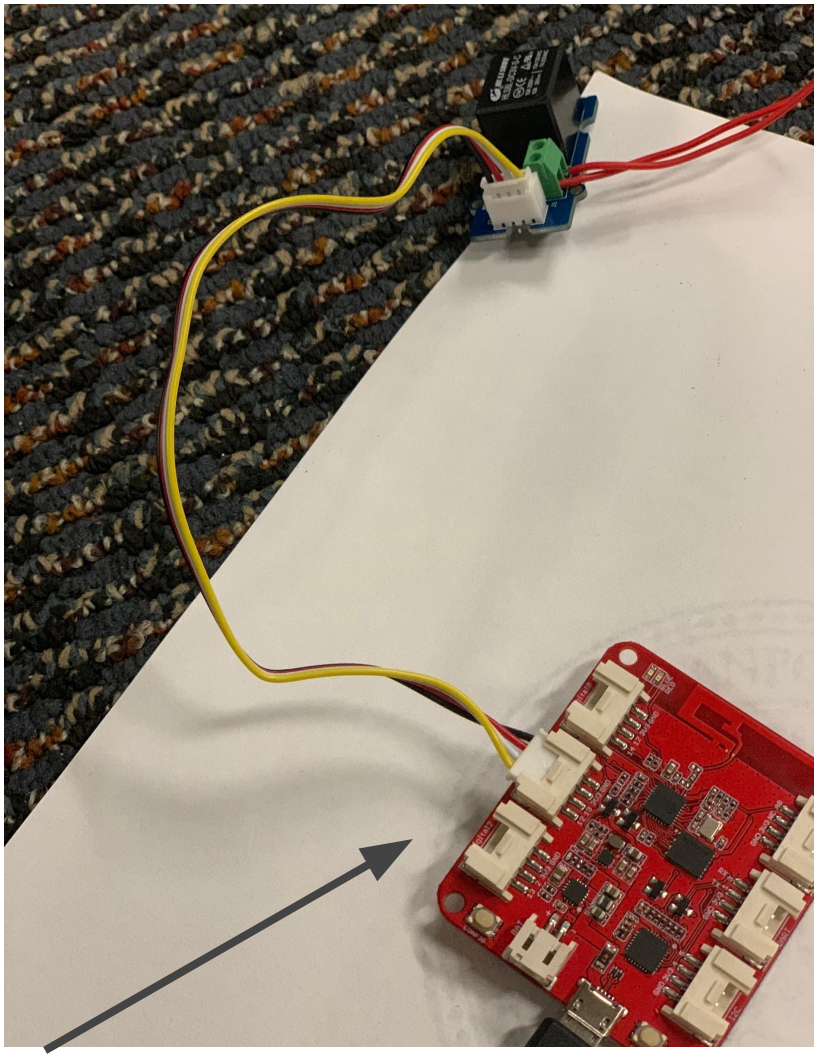
Plug in **4-pin wire connector** to Port 2

Table of Contents

Connecting Setup 1 & 2

Put the 4-pin connector into the Relay

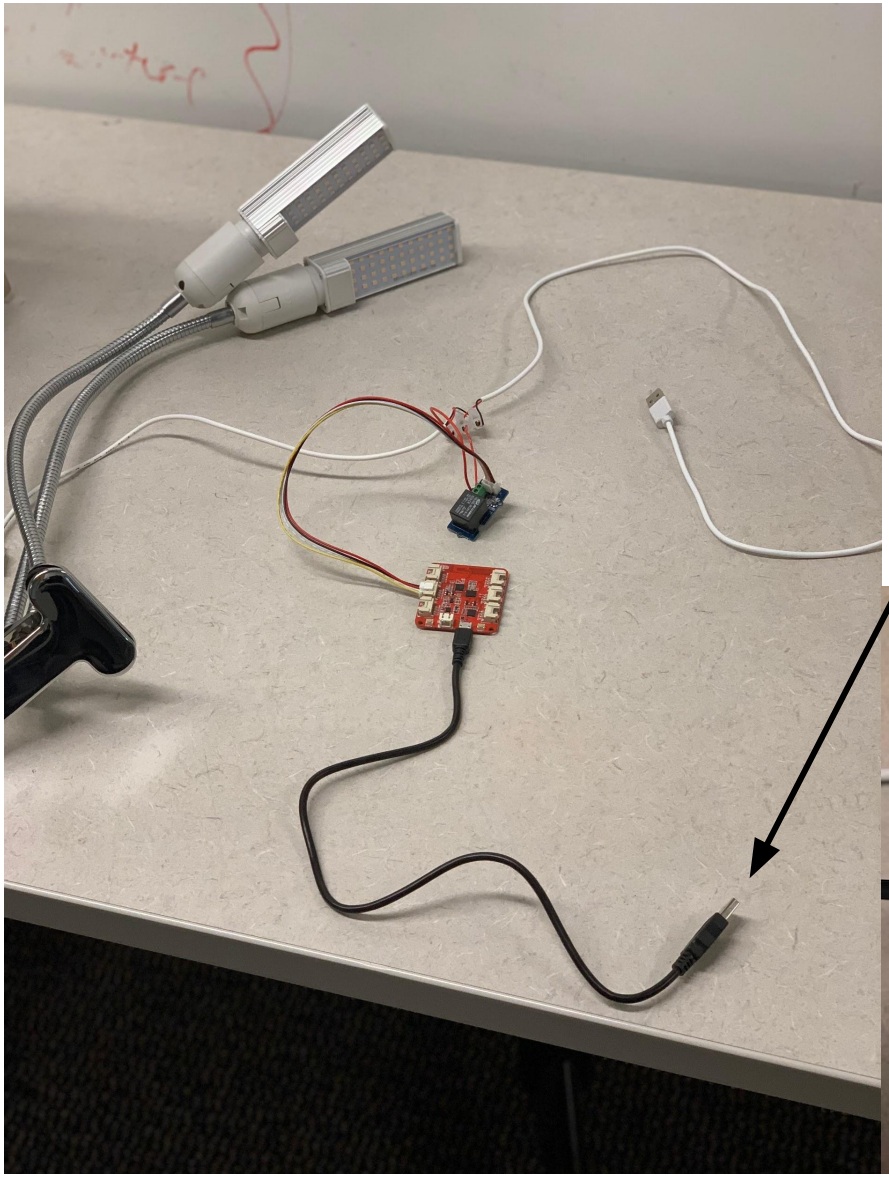




You have now connected your Grow Lamp setup to your MCU set up!

Table of Contents

Final Product!



This end of USB cord plugs into the computer

Table of Contents

Once connected to the computer, the **red light** on your MCU #2 should turn on!

Now you are ready to code…

*If you see a **blue light**, reflash your MCU. If no luck, get another MCU and re-label it.*

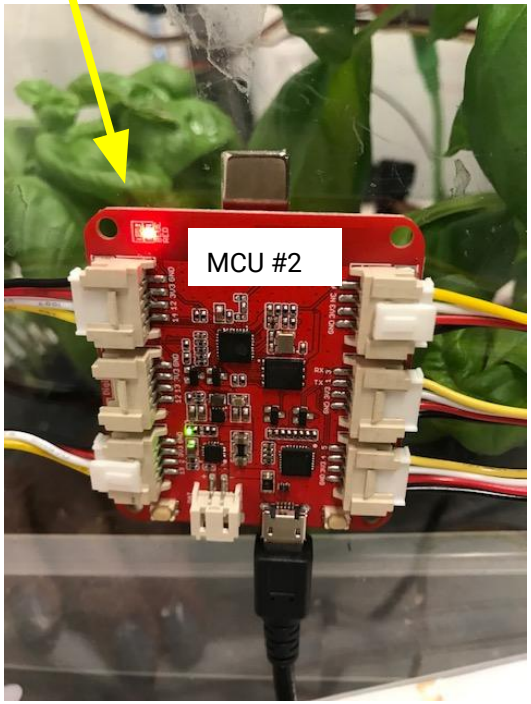


Table of Contents

Open EsPy from the desktop

For reminders on how to complete each step of the saving process, click **HERE.**

Click this from desktop

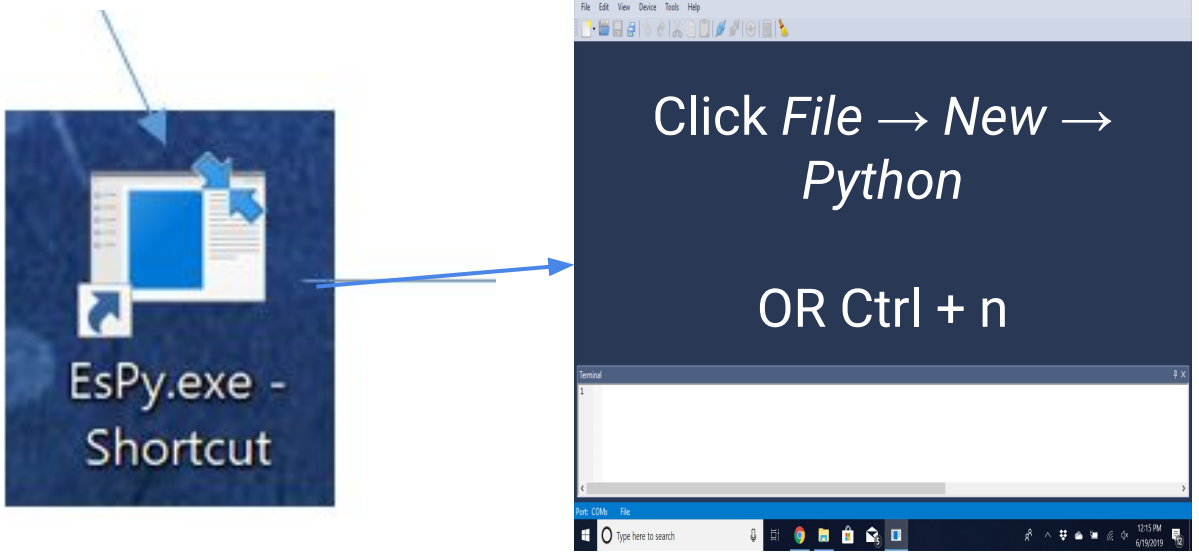


Table of Contents

Save new EsPy file your class folder as “mcu2-Grow Lamp”

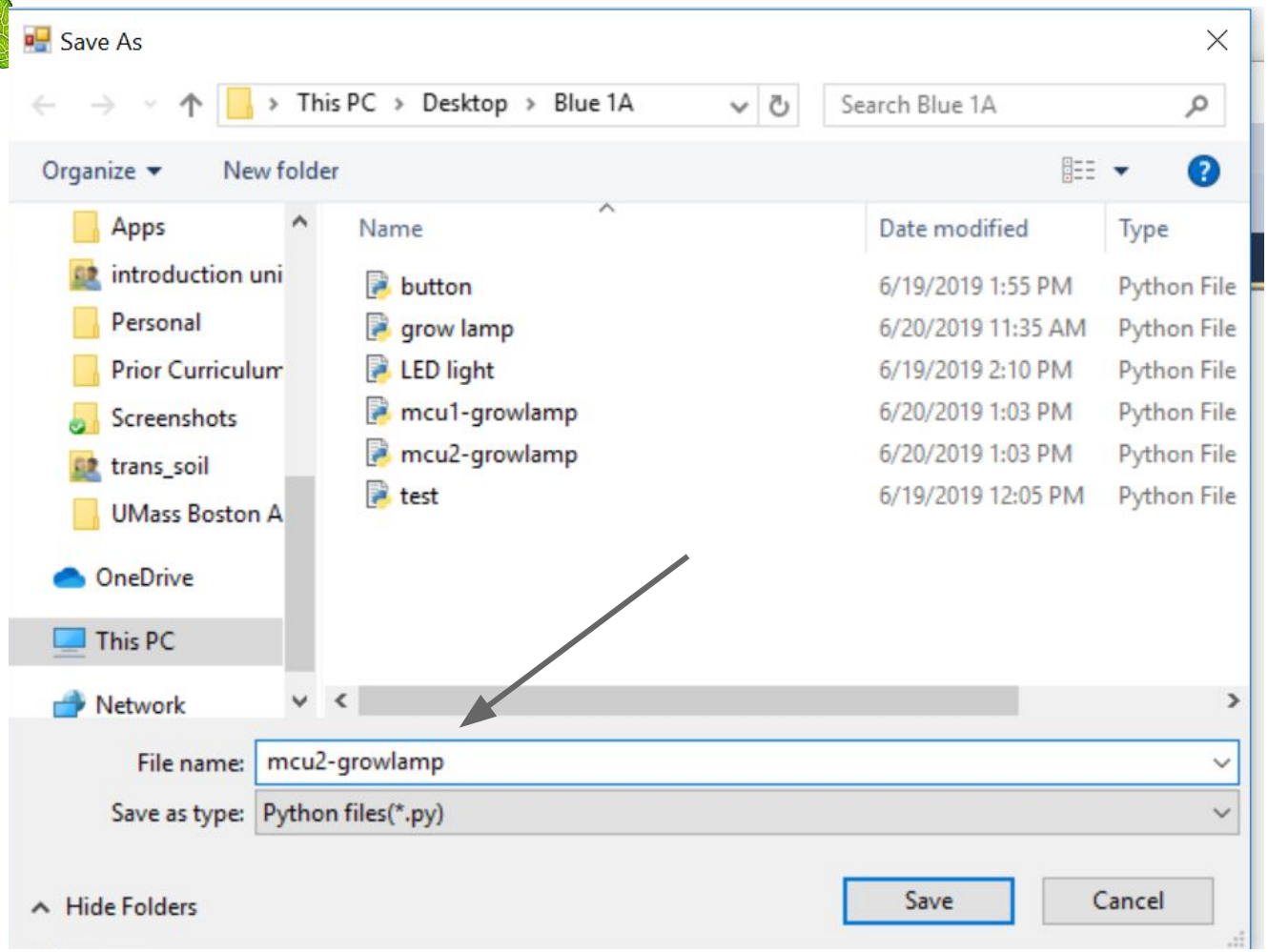


Table of Contents

Code for Grow Lamp

grow lamp.py*

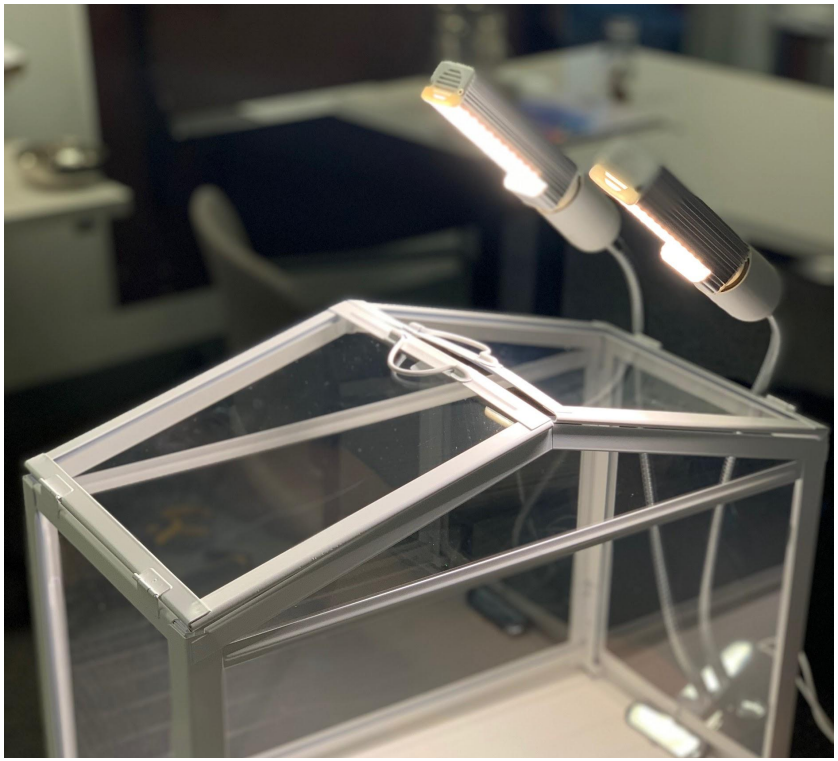
```
1   from actuators import Relay
2   import time
3
4   relaylight = Relay(2)
5
6   while True:
7       relaylight.on()
8       time.sleep(4)
9       relaylight.off()
10      time.sleep(4)
11
12
13
```

Type this code into EsPy

Table of Contents

Is it working??

If so, your Greenhouse lights should be ON and look like this:



If it's working, congrats!
Try these Challenges…

Table of Contents

Challenges: Relay & Grow Lamp

1. Change your code so the grow lamp will be **on for 3 seconds**, then **off for 5 seconds**, then **on for 2 seconds**, and then **off for 7 seconds**.

Hint: Remember you can copy and paste lines of code to have many lines!

2. This challenge involves material you learned yesterday... Can you have the **LED light Strip blink green** when the **relay is on (lights are on)** and **blink red** when the **relay is off (lights are off)**? Remember to put the LED light strip into Port 1!

Table of Contents

Exit Ticket

1. Did you get your lights on?
   If not how come?
2. How long will your timer be on?
3. How long will your timer be off?
4. Why did you choose those times?

Table of Contents

Lesson 3b: Light Sensor

OPTIONAL/EXTENSION

Table of Contents

Lesson 3b: Light Sensor

Image	Sensor/Device	Class	Ports
	Light Sensor	Sensors	6

Table of Contents

Do Now:

1. How would you explain to someone who has never coded before what a WHILE loop is?

2. What is wrong with the following code? Hint: There are 5 errors

Table of Contents

Problem of the day:

How can I have my Grow Lamps turn on if it is too dark for my plants?

Solution: Use a light sensor to turn lights on when it is too dark and off when it is too bright for my plants!

Table of Contents

Problem-Solving Process

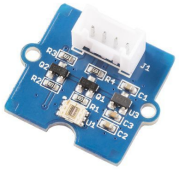
Table of Contents

Brainstorm: Light Sensor

- How do you connect the Grow Lamps to the Light Sensor?
- What kind of information will the Light Sensor give us?
- How much light does a plant need?

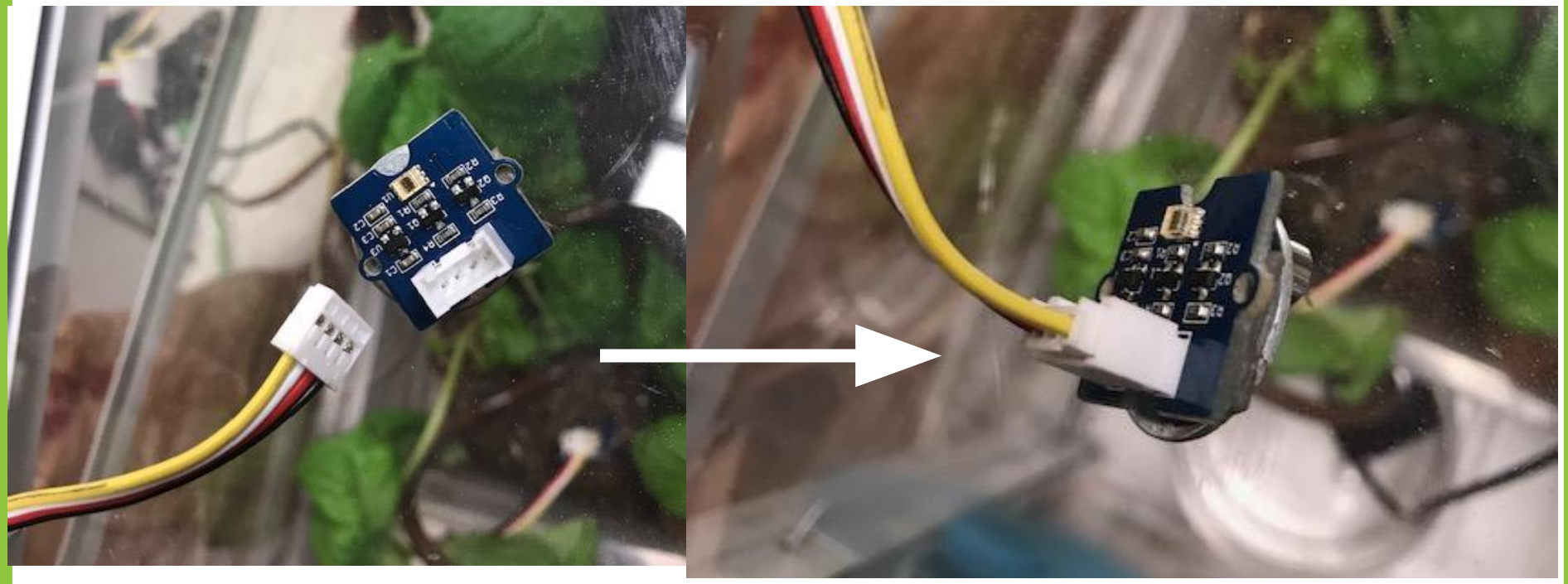
Table of Contents

Gathering Materials Lesson 4a: Temperature & Humidity Sensor

Device	Picture
MCU #2	
Two 4-Pin Connectors	 
Black USB cable	
Light Sensor	
I2c Hub	

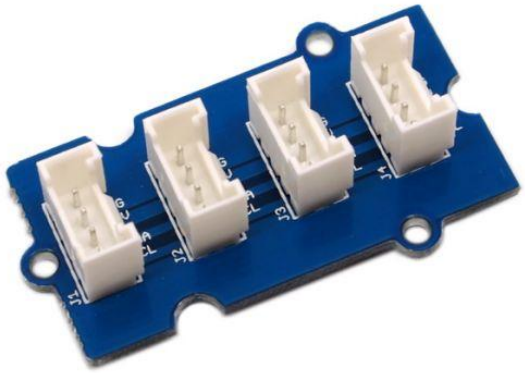
Assembling Light Sensor

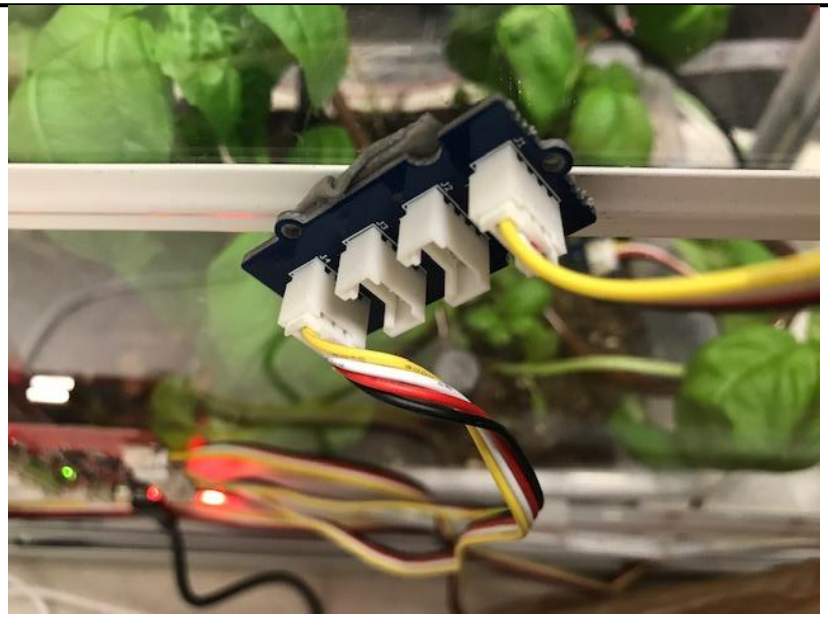
1. Attach 4 pin connector cable to light sensor



Assembling Light Sensor

2. Attach other end of 4 pin connector cable to a hub (extension board), then add a 2nd 4 pin connector cable to the end of the hub.





Assembling Light Sensor

3. Insert remaining end of 4 pin connector cable to <u>Port 6</u> of MCU #2

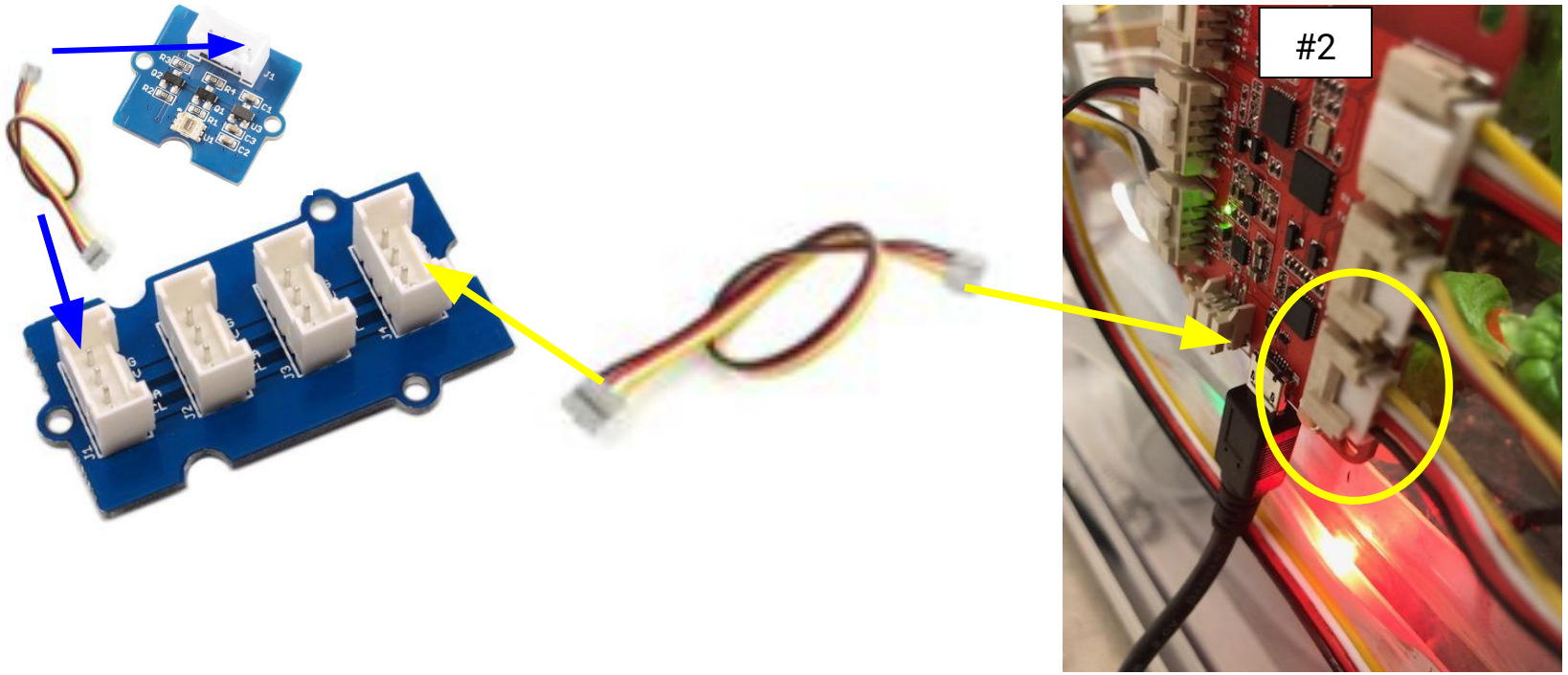


Table of Contents

Once connected to the computer, the **red light** on your MCU #2 should turn on!

Now you are ready to code…

*If you see a **blue light**, reflash your MCU. If no luck, get another MCU and re-label it.*

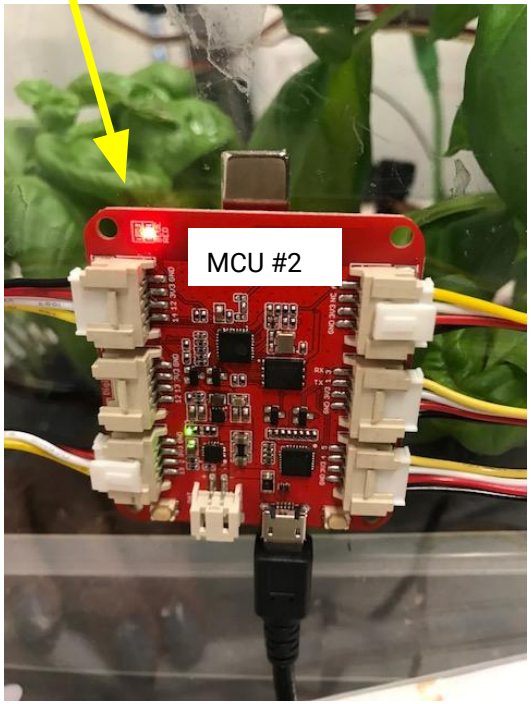


Table of Contents

Open EsPy from the desktop

For reminders on how to complete each step of the saving process, click **HERE.**

Click this from desktop

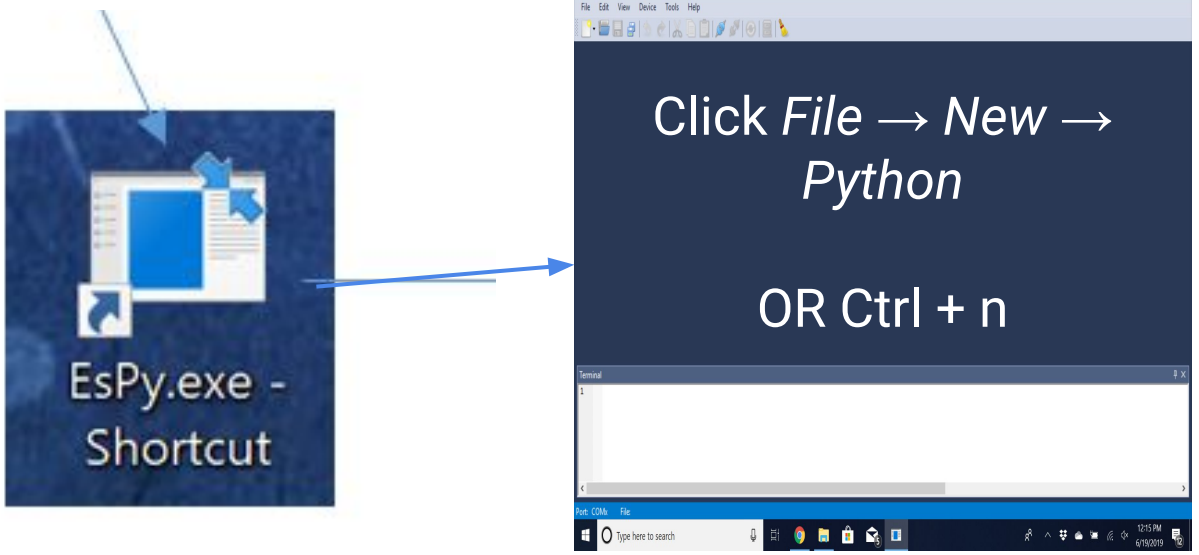


Table of Contents

Save new EsPy file your class folder as “mcu2-lightsensor”

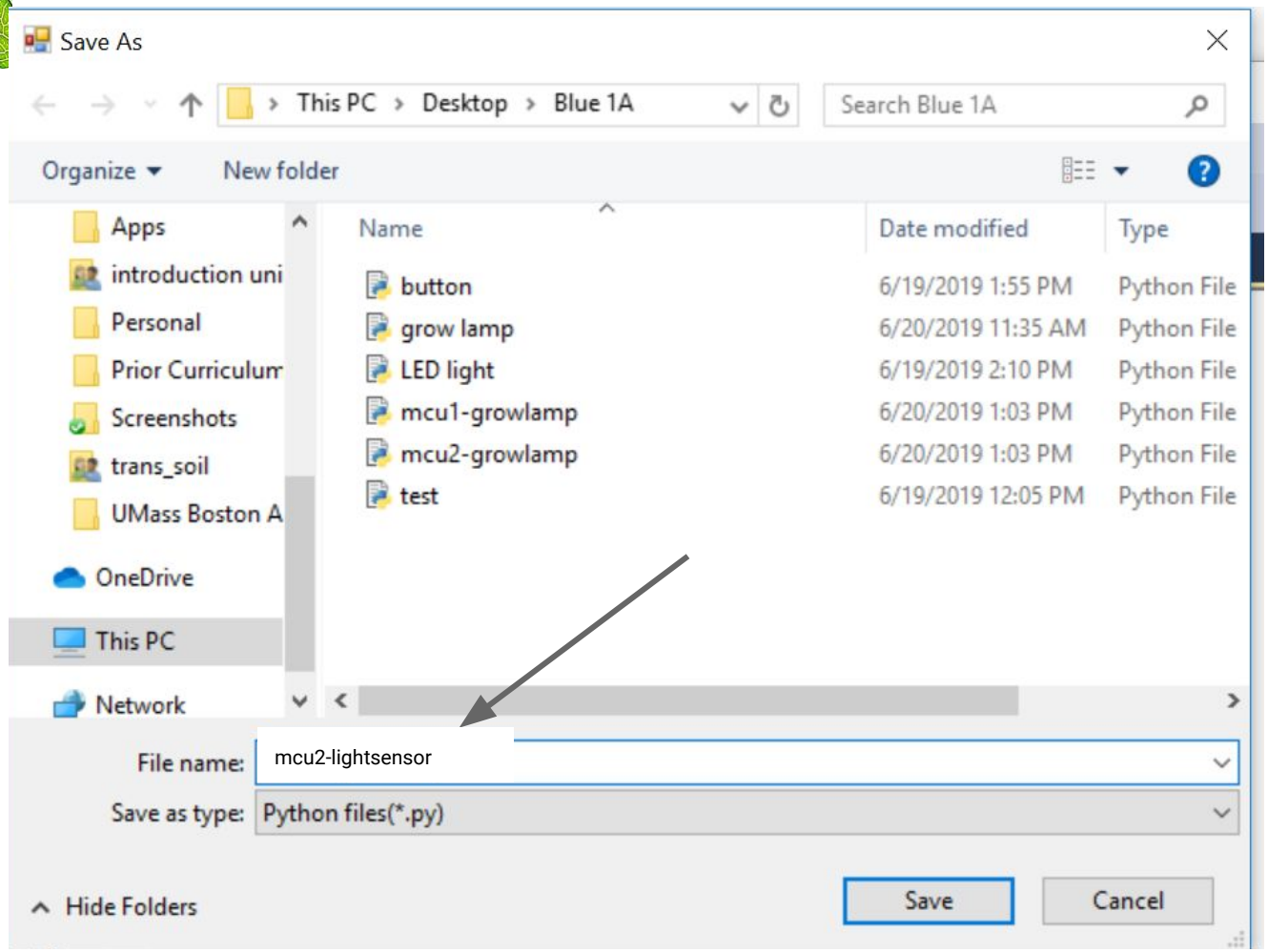


Table of Contents

Code for Light Sensor

```
1  from sensors import LightSensor
2  from actuators import Relay
3  import time
4
5  relaylight= Relay(2)
6  ls = LightSensor(6)
7
8  Lux_High = 1000
9
10 while True:
11     l = ls.get_lux()
12     if l < Lux_High:
13         relaylight.on()
14         print('light value is low.  Grow Lamp should be on.', l)
15         time.sleep(3)
16     else:
17         relaylight.off()
18         print('light value is good. Grow Lamp should be off.', l)
19         time.sleep(3)
20
```

This is a cursor!
Do not type!

This is a small
“L” not a 1 !!!!

Data for Light Sensor

You should start to see light sensor data in your terminal…

Terminal

```
100
101     ls = LightSensor(6)
102
103     while True:
104         l = ls.get_lux()
105         print('light value is', l)
106         time.sleep(3)
107
108     light value is 5.8653
109     light value is 5.8653
110     light value is 5.8653
111     light value is 4.91481
112     light value is 4.91481
113     light value is 4.91481
114     light value is 5.8653
115
```

Activity Option

1. Put the light sensor right under the grow lamps, what happens? Why?

2. Start with the light sensor 1m away from the light. Record the lux value.

3. Move it 10cm closer and record the lux value.

4. Keep going until you are at the lights again. Graph your data.

Table of Contents

Exit Ticket (3b)

1. How might you want to use the light sensor in your greenhouse?

2. What would that code look like?

3. What is a potential downside of using the light sensor?

Table of Contents

Lesson 4a: Temp/Humidity Sensor

Table of Contents

Lesson 4a: Temp/Humidity Sensor

Images	Sensor/Device	Class	Ports
	Temperature/Humidity	Sensors	1, 2, 3
	Propeller Fans	Actuator	

Table of Contents

Do Now:

1. Why should we regulate the temperature and moisture in the greenhouse?
2. What might happen when our greenhouse is too hot?
3. What is an easy-fix to this problem?

Student Answers:

Table of Contents

Problem of the day:

How do we keep our greenhouse at the perfect temperature and humidity level?

Solution: We detect the temperature and humidity and use the Fans when needed!

Table of Contents

What is temperature?

- **Temperature** is…
  - a measure of the **kinetic energy** in a substance
    - ⇧ temp = molecules move **faster**
    - ⇩ temp = molecules move **slower**

movement

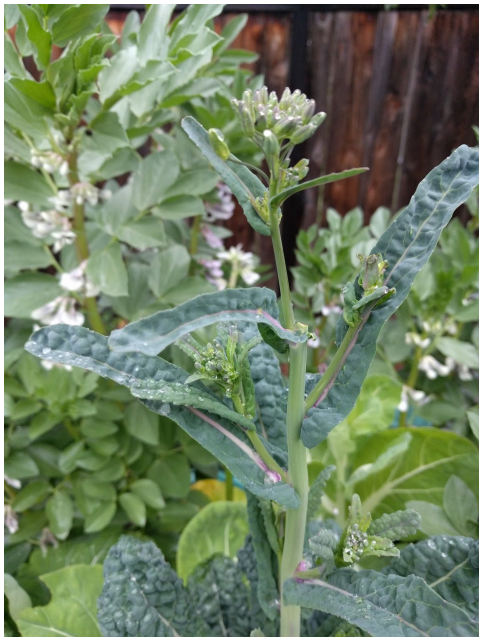
Ice (solid) < Water (liquid) < Steam (gas)

slowest moderate fastest

Table of Contents

Why is temperature important?

- If temperature is **too hot** plant will be stressed and may bolt
  - Bolt = produce seeds instead leaves
  - Transpire (sweat) more and will need more water to stay alive
  - Less food!



Greenhouses should be between 75-85 degrees during the day

Table of Contents

What is humidity?

- **Relative Humidity**
  - Amount of water in the air compared to what air can hold at a certain temperature.
  - Examples:
    - Air with a relative humidity of 50% contains half of the maximum amount of water
    - 10% relative humidity means there is very little water in the air and the air could hold much more water.

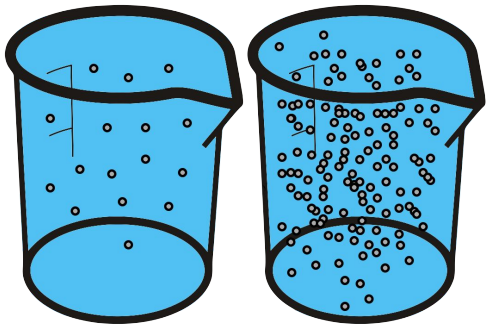


Table of Contents

Why is humidity important?

If humidity is too high...

- Water may condense on leaves and cause disease or fungus to grow on them.

What are good humidity levels?

- According to the University of Massachusetts Agriculture Department greenhouses should be around:

Record this chart in your journal!

Temperature (F)	Humidity
50°	83%
61°	89%
68°	91%
86°	95%

It is important to remember that every time you change the temperature, it will change the humidity, and vice versa.

Table of Contents

How are temperature and humidity related?

- As the air gets hotter it can hold more water, which results in higher relative humidity

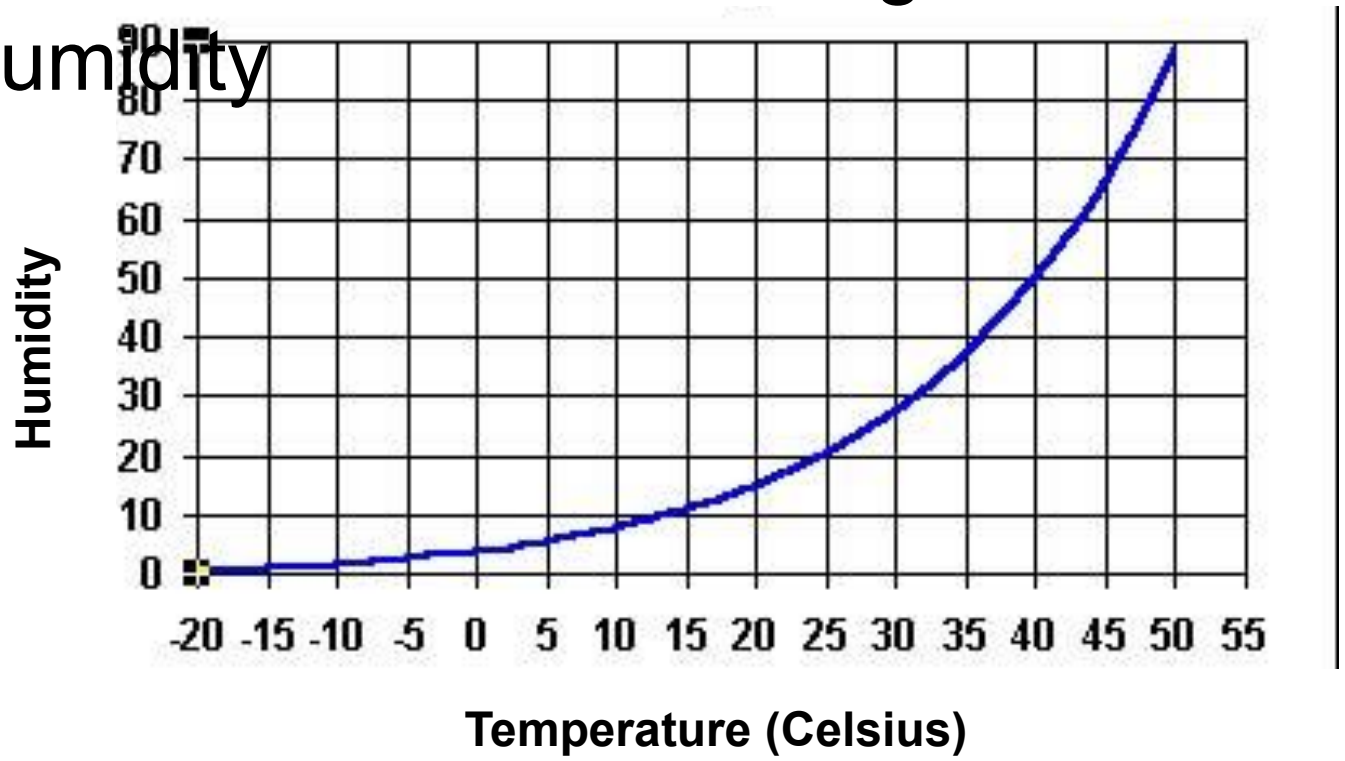


Table of Contents

Problem-Solving Process

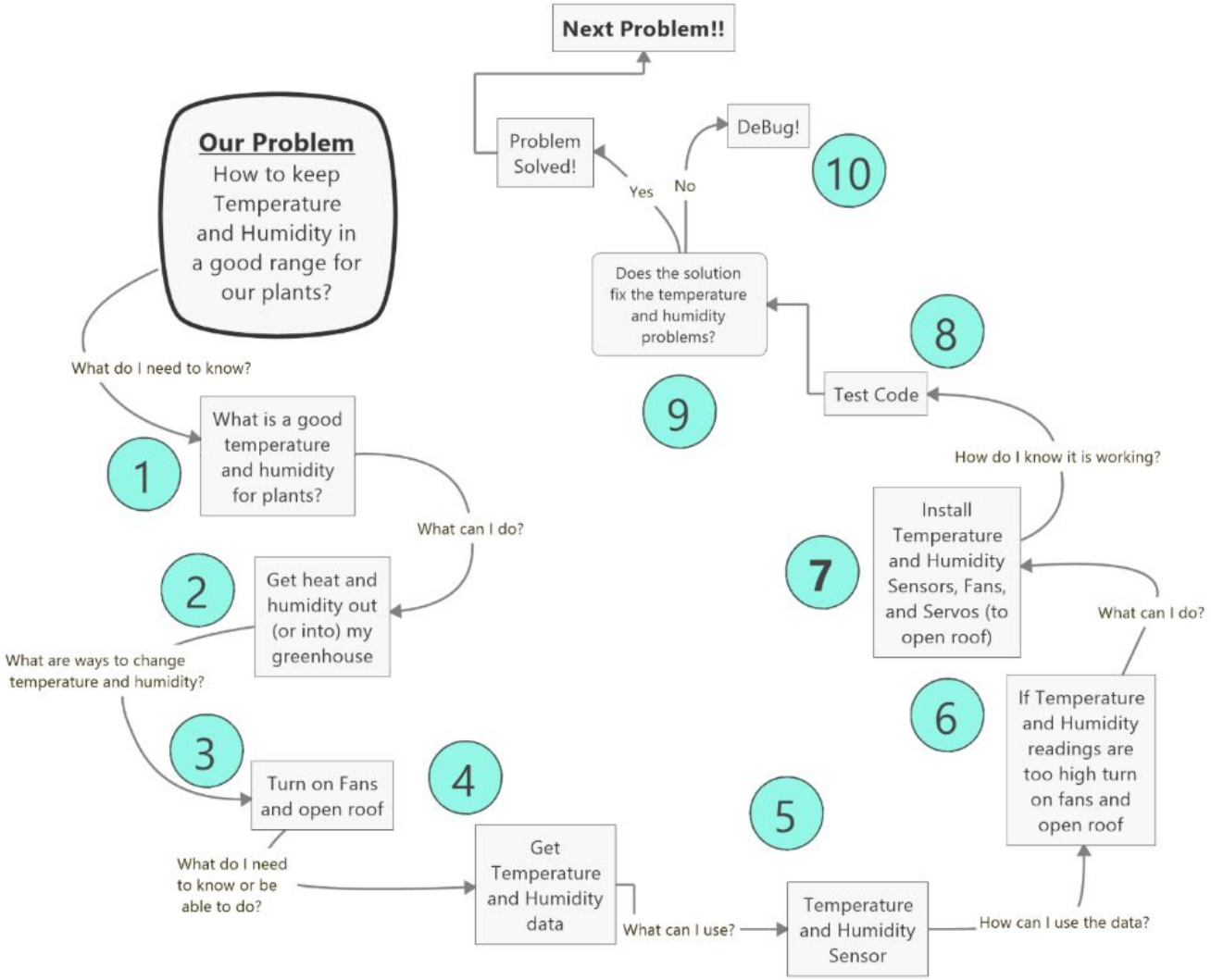


Table of Contents

Brainstorm:

- How much does temperature and humidity impact plant growth?
- What is the relationship between temperature and humidity?
- What can we use a temperature/humidity sensor for in our greenhouse?

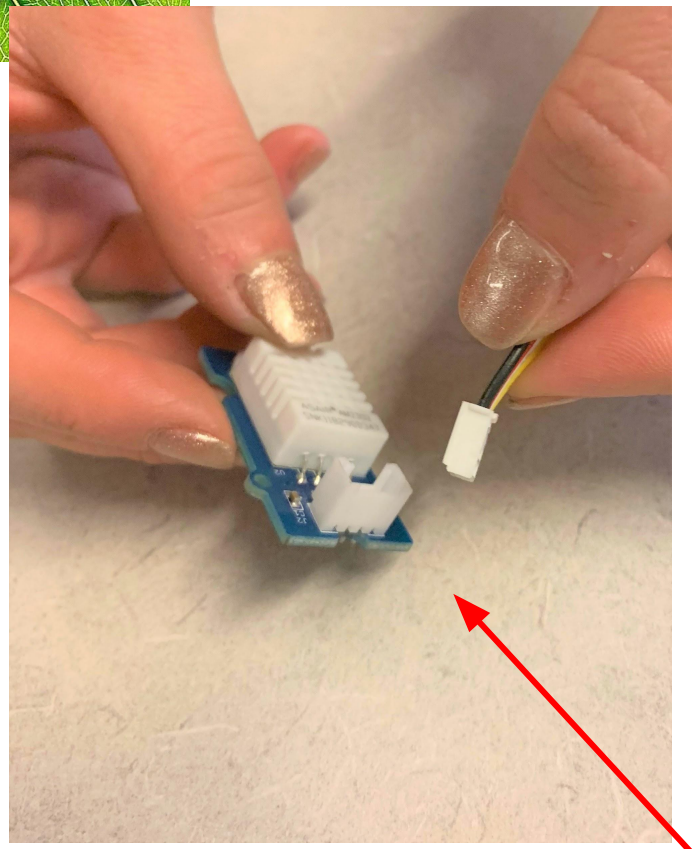
Table of Contents

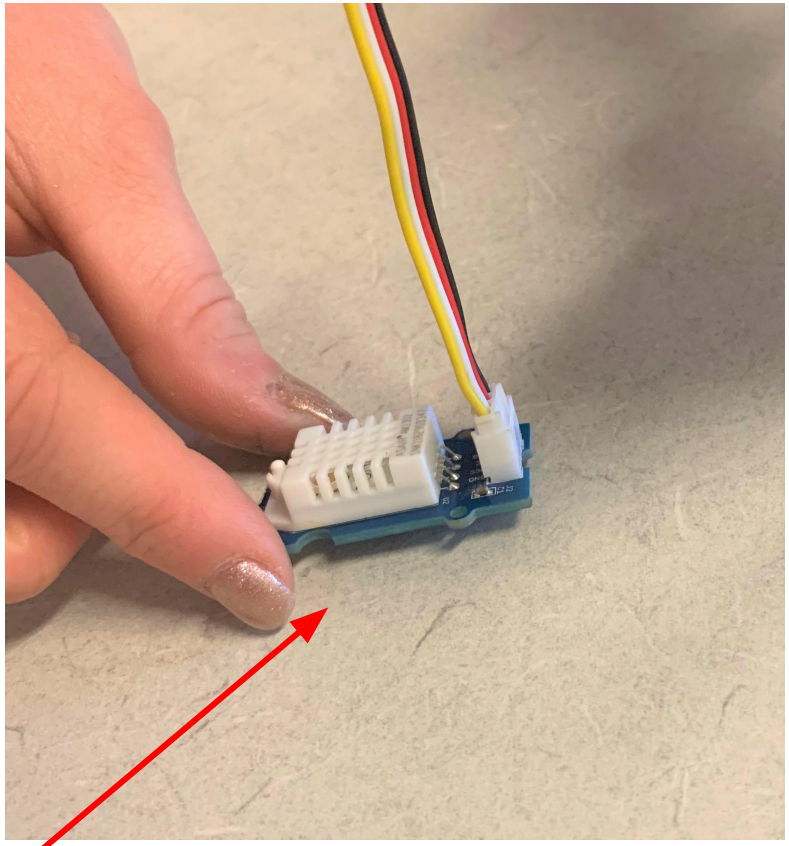
Gathering Materials Lesson 4a: Temperature & Humidity Sensor

Device	Picture
MCU #1	
4-Pin Connector	
Black USB cable	
Temperature & Humidity Sensor	

Table of Contents

Assembling Temperature & Humidity Sensor



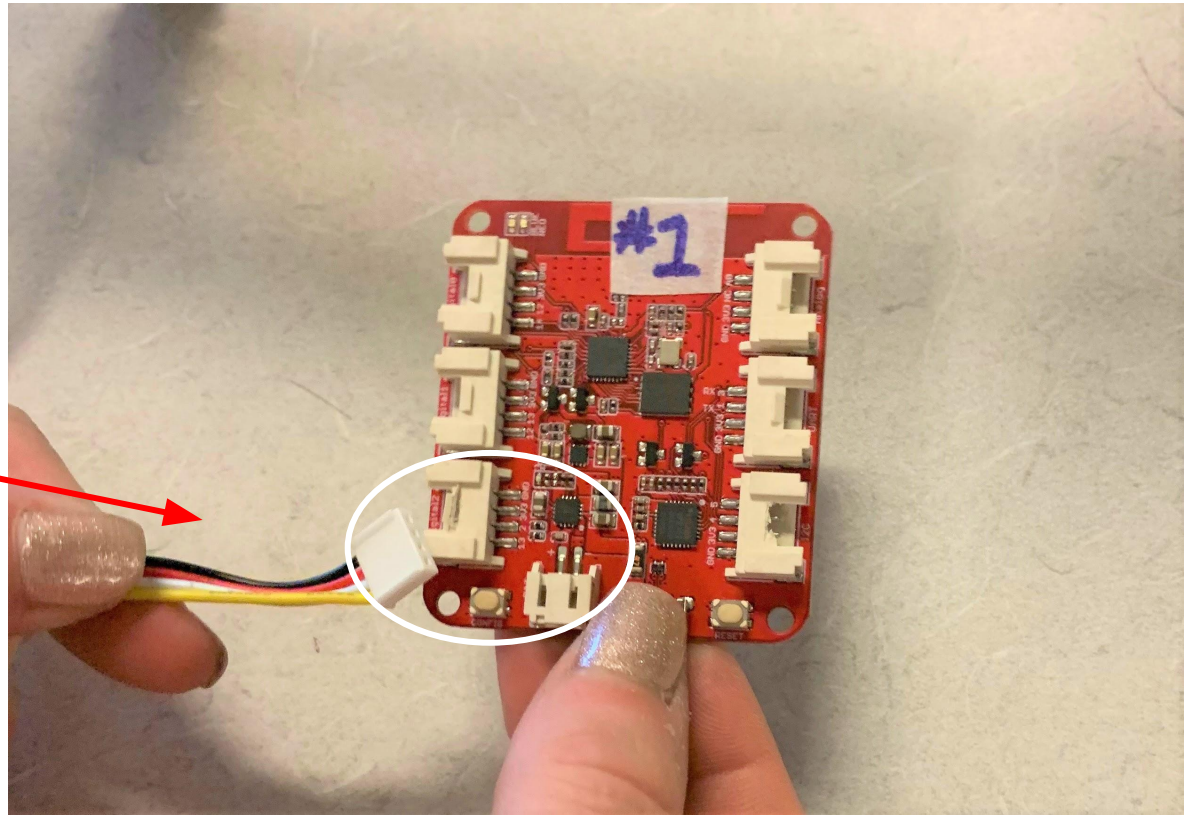


Put one side of the 4 pin wire connector into the moisture sensor

Table of Contents

Assembling Temperature & Humidity Sensor

Put the other end of the 4 pin wire connector in port #3



MAKE SURE YOU ARE DEALING WITH MCU #1

Table of Contents

Assembling Temperature & Humidity Sensor

Plug the black USB cable into MCU #1

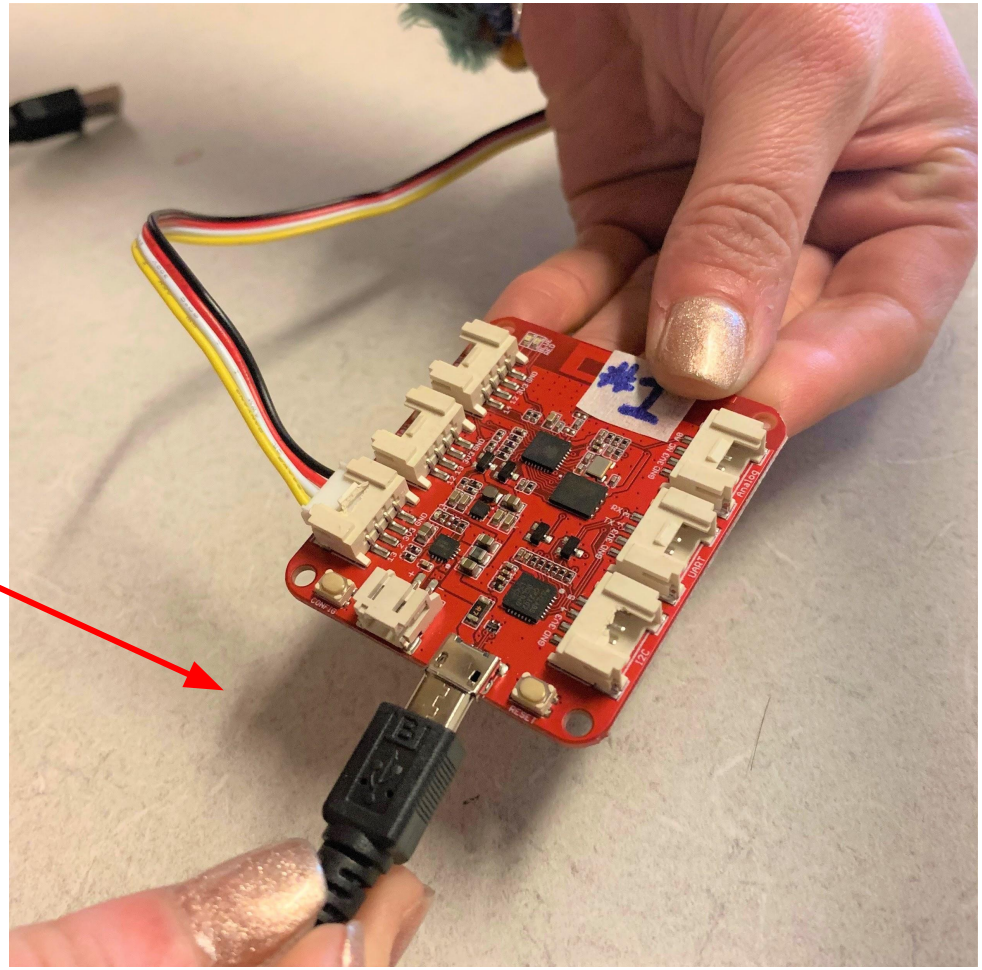


Table of Contents

Assembling Temperature & Humidity Sensor

Put the Temperature and Humidity Sensor inside the greenhouse and shut the roof.

Then place the MCU on your Greenhouse

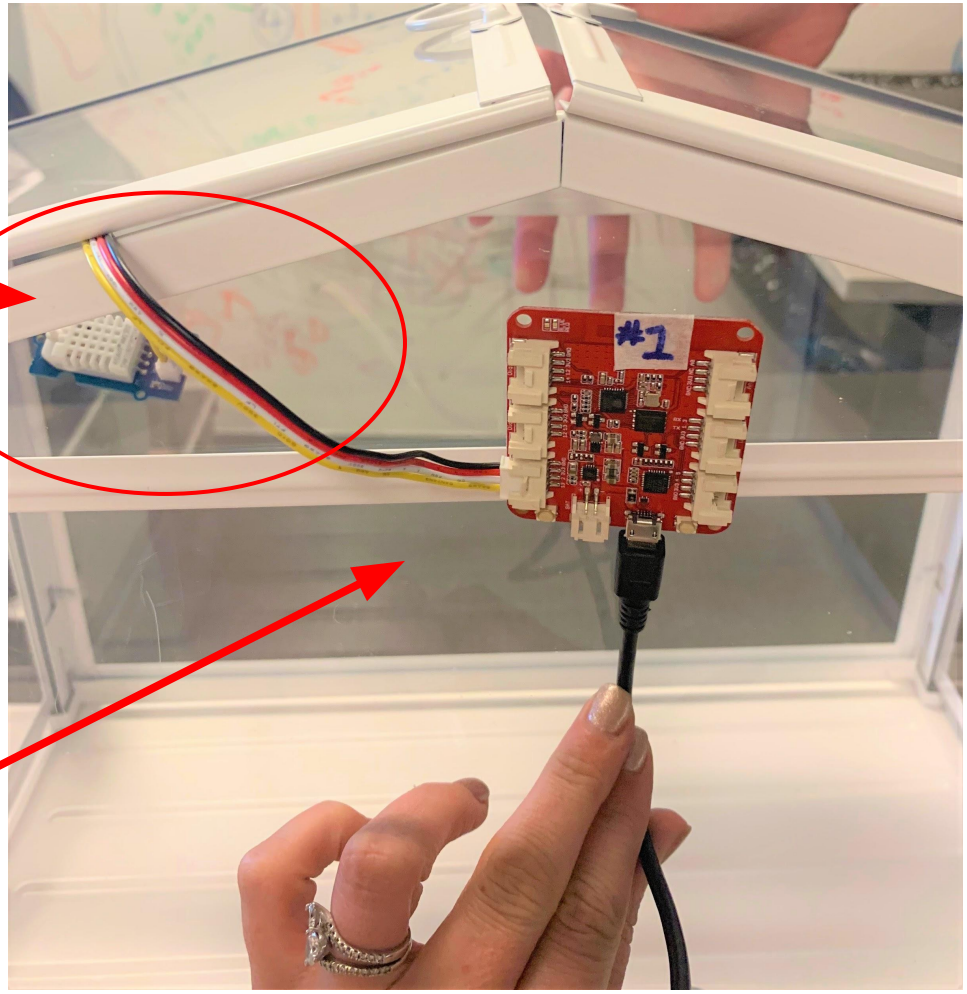
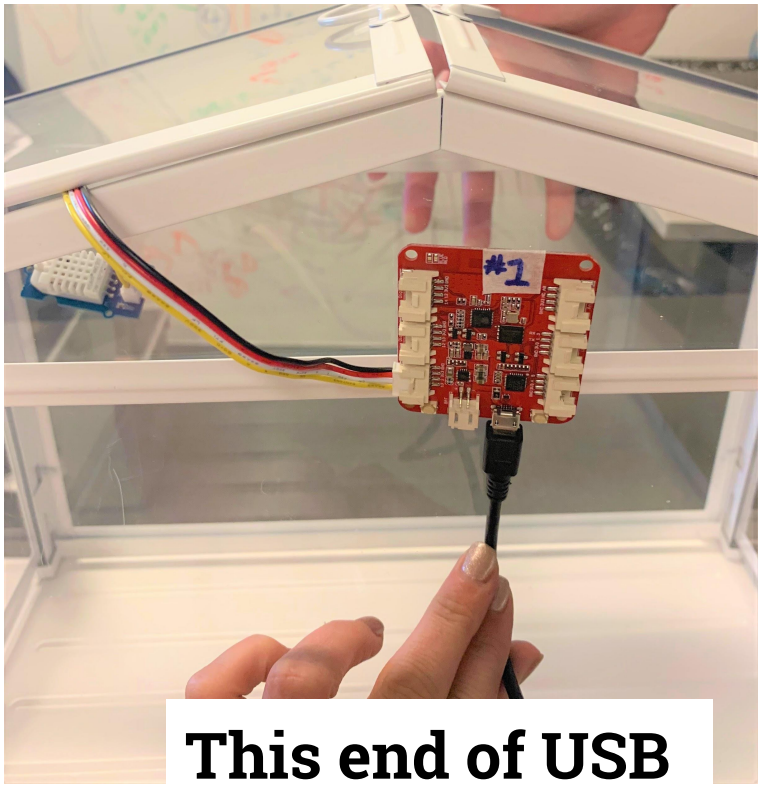


Table of Contents

Assembling Temperature & Humidity Sensor



This end of USB cord plugs into the computer

Table of Contents

Once connected to the computer, the **red light** on your MCU #1 should turn on!

Now you are ready to code…

*If you see a **blue light**, reflash your MCU. If no luck, get another MCU and re-label it.*

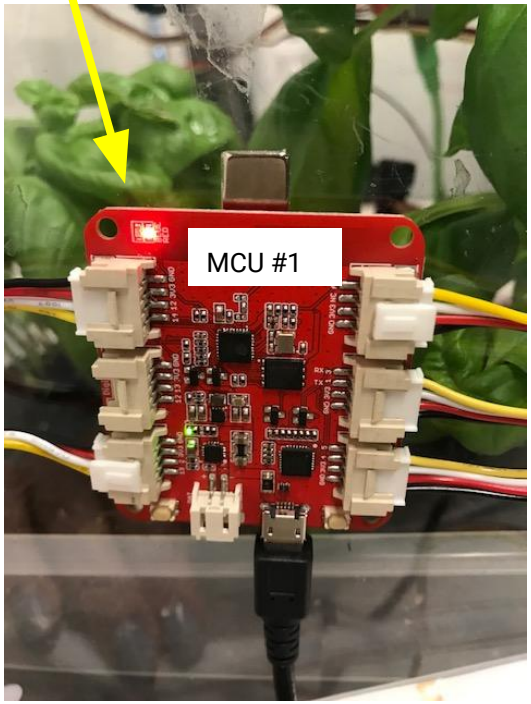


Table of Contents

Open EsPy from the desktop

For reminders on how to complete each step of the saving process, click **HERE.**

Click this from desktop

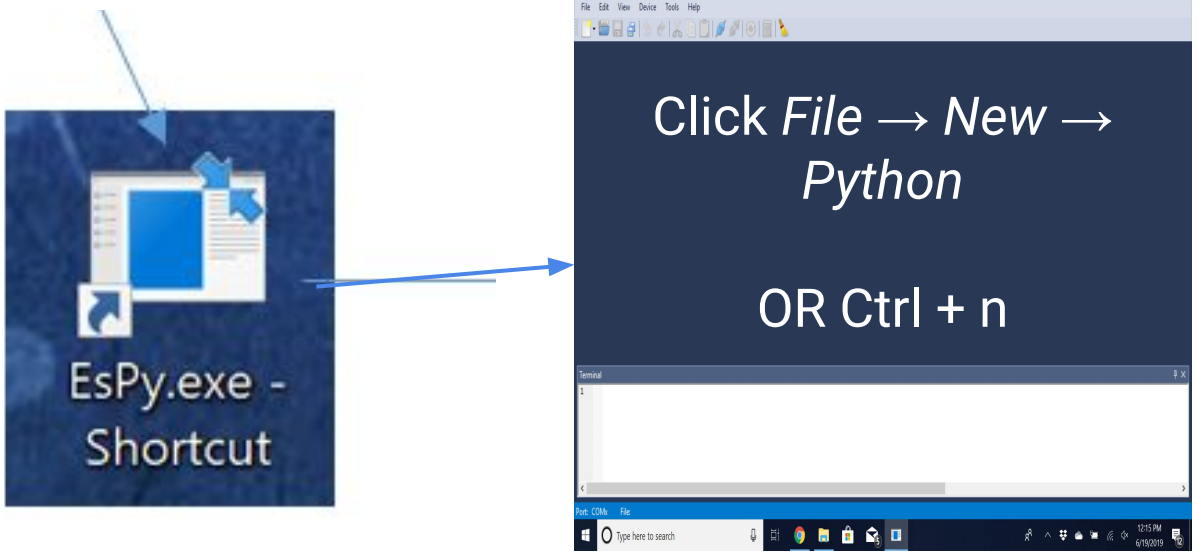


Table of Contents

Save new EsPy file your class folder as “mcu1-temphumidity”

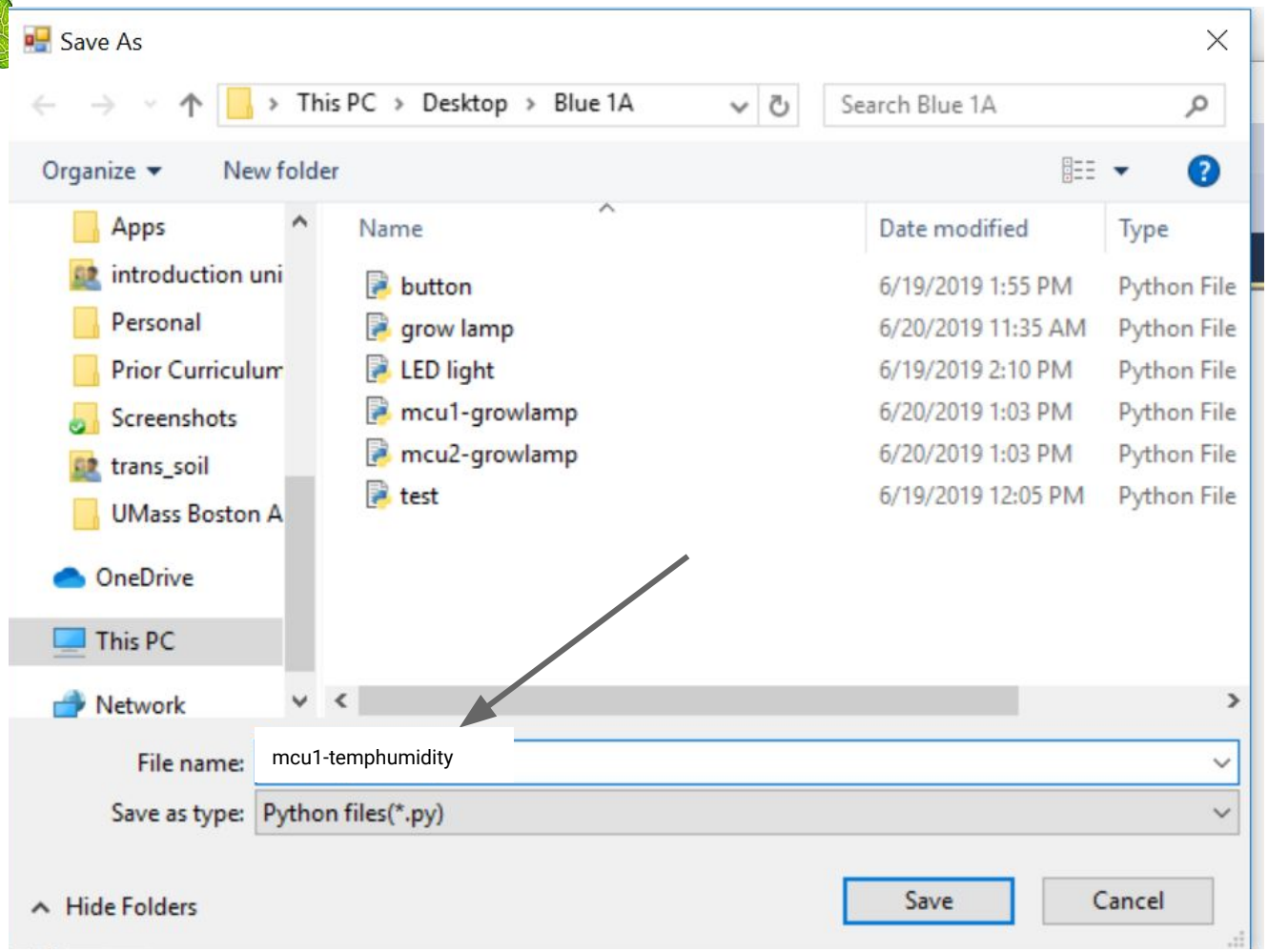


Table of Contents

Codes to get readings of temperature and humidity

```
1   from sensors import TemperatureSensorPro
2   import time
3
4   tempsensor = TemperatureSensorPro(3)
5
6   while True:
7       temp = tempsensor.get_temperature()
8       humidity = tempsensor.get_humidity()
9       print("temperature is:", temp, "humidity is:", humidity)
10      time.sleep(4)
```

Table of Contents

Challenges: 4a Temperature & Humidity

1. Put a cup of hot water in your Greenhouse.
2. Watch what happens to the temperature and humidity readings.
3. Graph your findings.

Table of Contents

Exit Ticket (4a)

1. What is the purpose of a Temperature & Humidity Sensor?

2. What happened to the Temperature and Humidity when you put a cup of hot water in your greenhouse? Why did this happen?

Table of Contents

Lesson 4b: Propeller Fans

Table of Contents

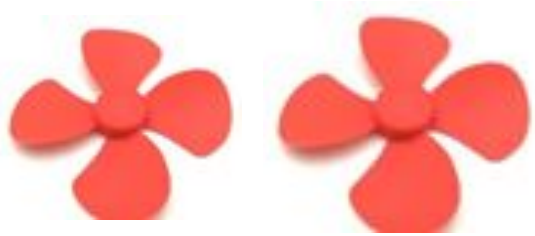
Lesson 4b: Propeller Fans

Images	Sensor/Device	Class	Ports
	Propeller Fans (from a Relay)	Actuator	1, 2, 3

Table of Contents

Do Now (Lesson 4b):

Identify and explain 3 reasons why having fans would be helpful in a greenhouse



Class Share-Out:

Table of Contents

Problem of the day:

How can we regulate the temperature/humidity in our greenhouse?

Solution: Use propeller fans to circulate the air!

Table of Contents

Problem-Solving Process

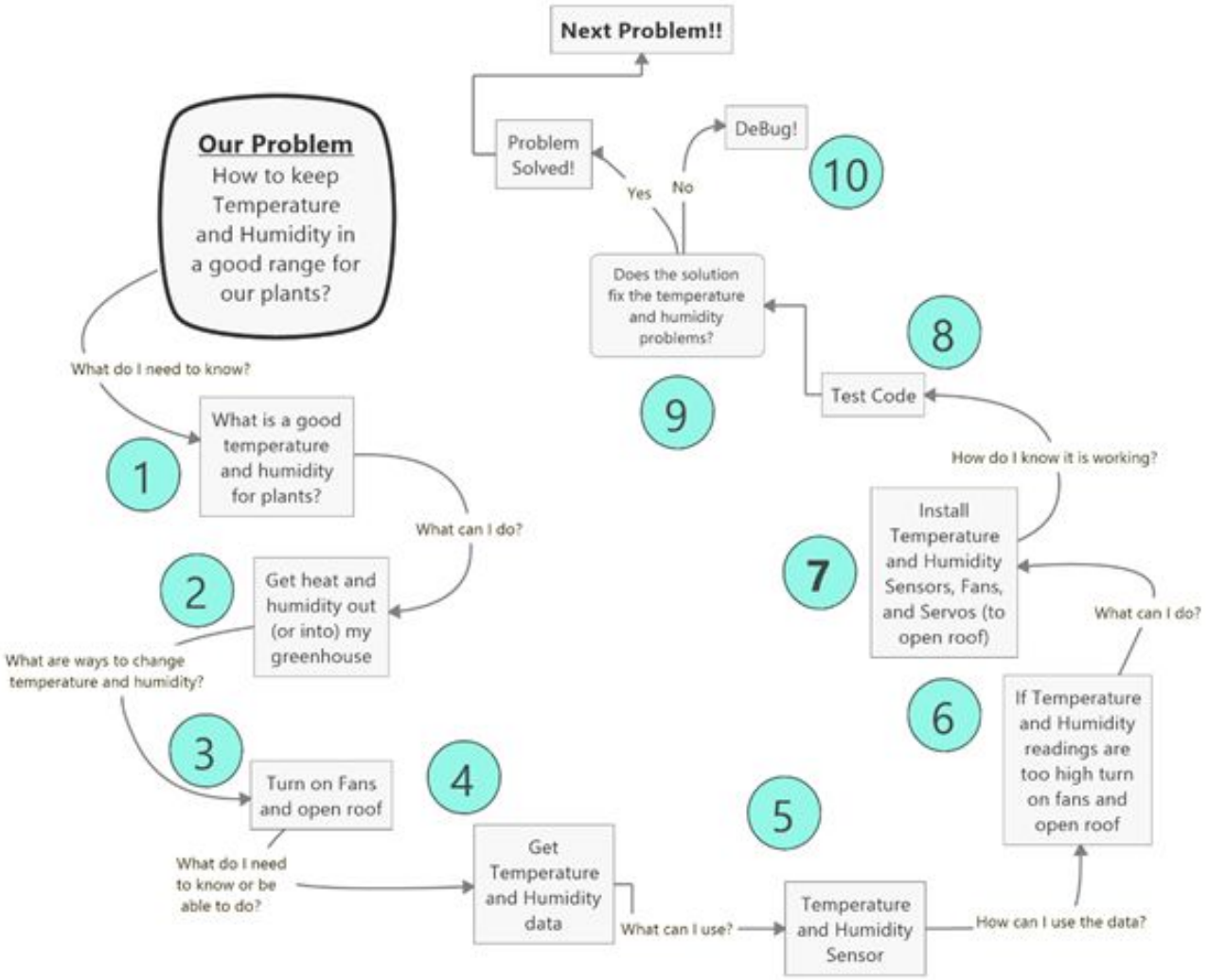


Table of Contents

Brainstorm: Propeller Fans

- When should we turn on the propeller fans?

- What device will control when the propeller fans turn on?

- Where should the two propeller fans be placed to maximize airflow within the greenhouse?

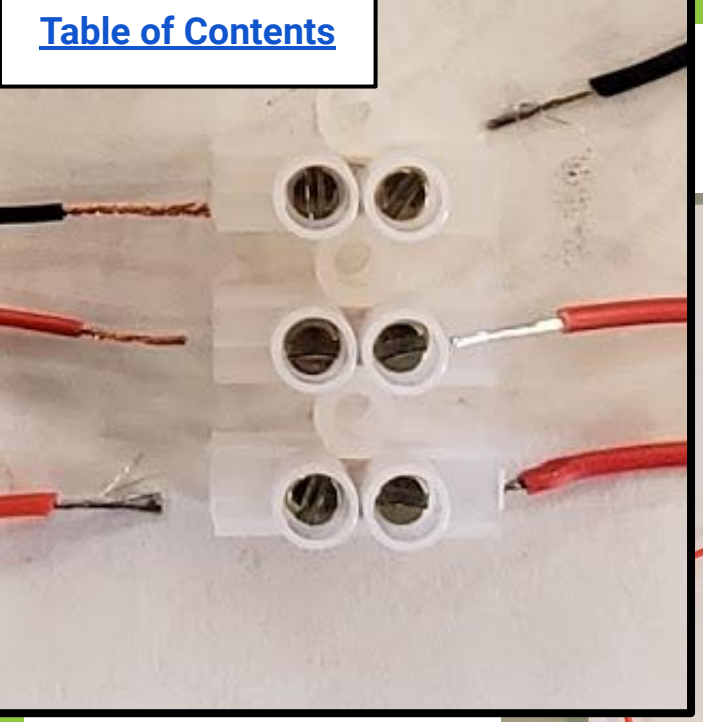
Table of Contents

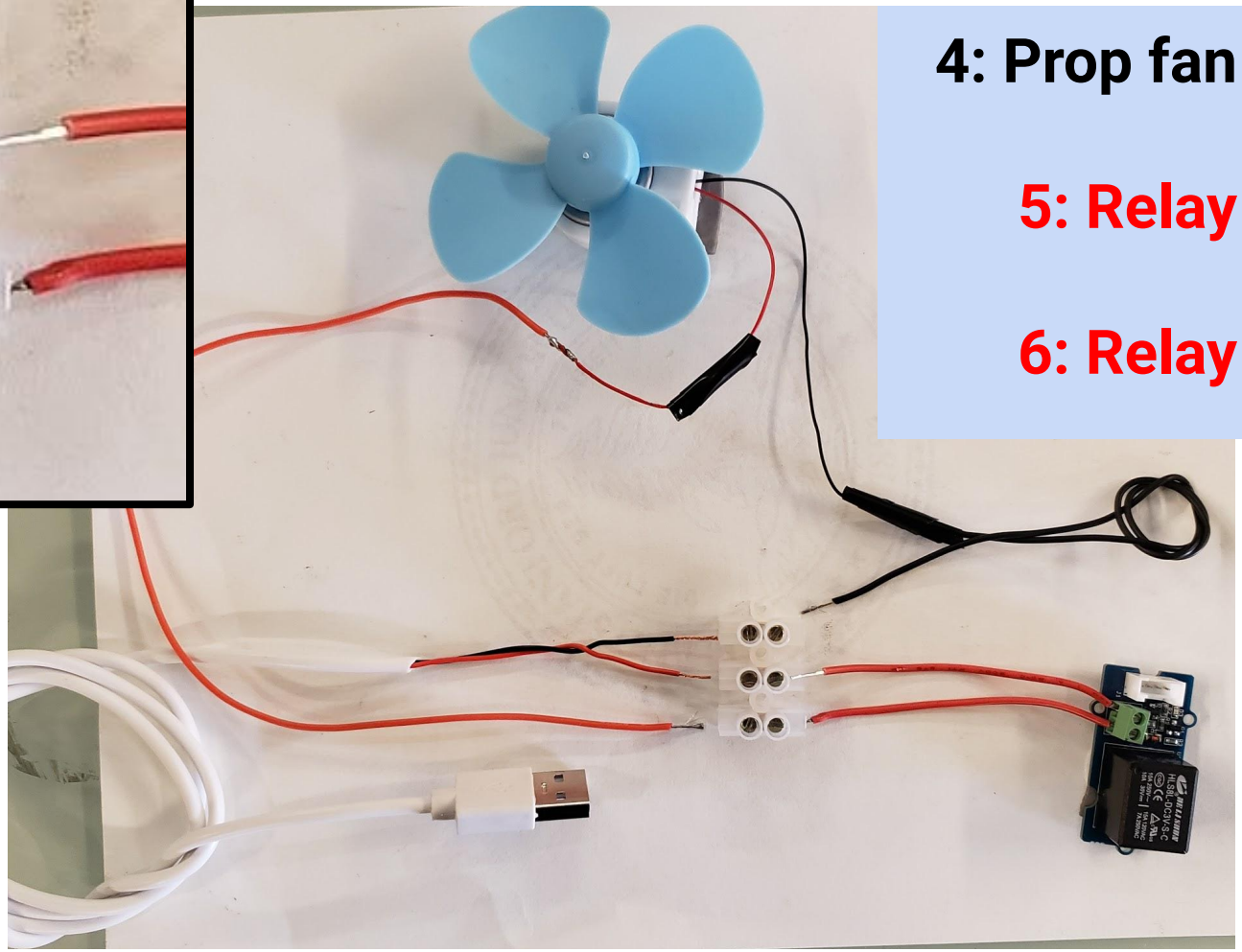
Materials: 4b Propeller Fans

Number/Name of Part	Picture of Part	Number/Name of Part	Picture of Part
1- Wio Link Board (MCU).		1- Temp Sensor	
1- Micro-USB Cord (to plug in the MCU)		1- Micro-USB Cord Stripped (wires exposed)	
1- Relay with 2 red wires		1- Screwdriver	
1- terminal block		3- 4 Pin Wire Connector	
May need a pair of wire strippers (unlikely) 		1-2 motors with fans (at least 1).	

Table of Contents

Wiring Propeller Fan





4: Prop fan

5: Relay

6: Relay

1: USB Power

2: USB Power

3: Prop fan

Black text = **black** wire; Red text = **red** wire

Table of Contents

Connect your Propeller Fan & Relay to your MCU #1

Place the 4 pin wire into the relay

Plug the other end of the 4 pin wire into port 2 of MCU 1

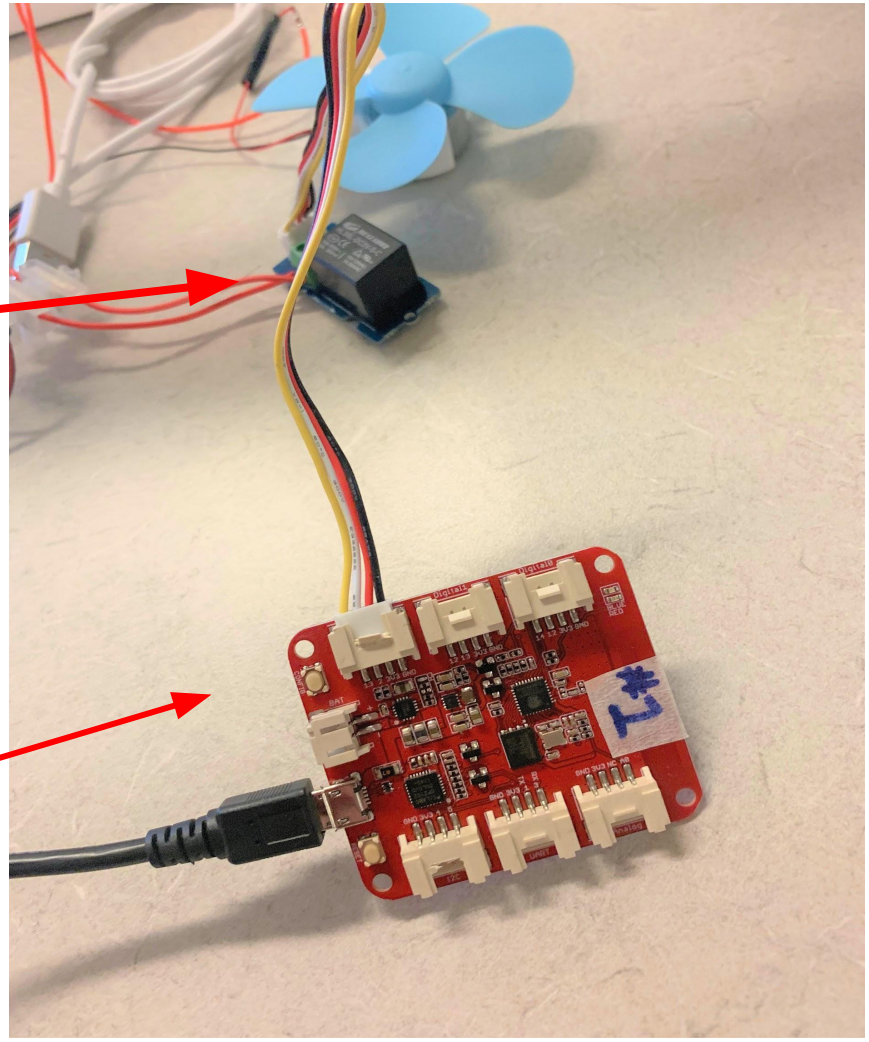


Table of Contents

Once connected to the computer, the **red light** on your MCU #1 should turn on!

Now you are ready to code…

*If you see a **blue light**, reflash your MCU. If no luck, get another MCU and re-label it.*

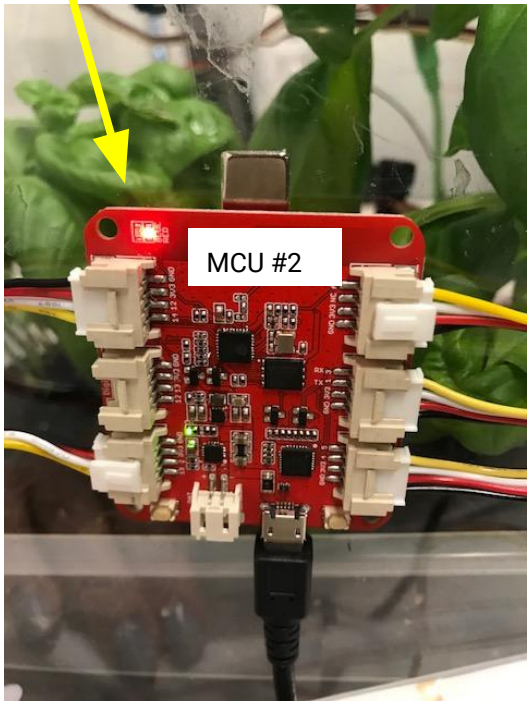


Table of Contents

Open EsPy from the desktop

For reminders on how to complete each step of the saving process, click **HERE.**

Click this from desktop

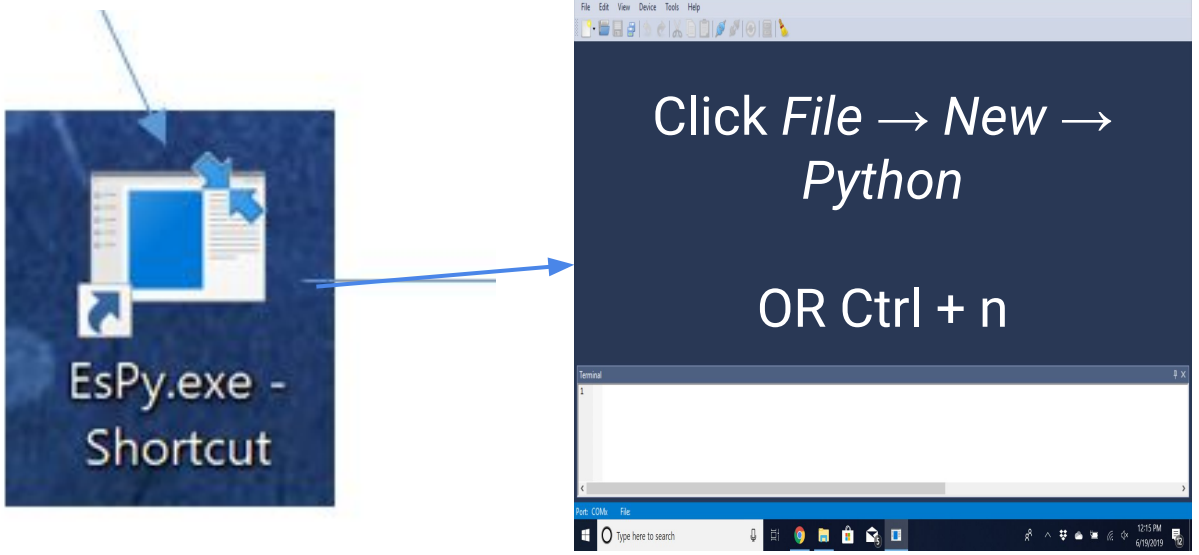


Table of Contents

Save new EsPy file your class folder as “MCU1-Propeller Fans”

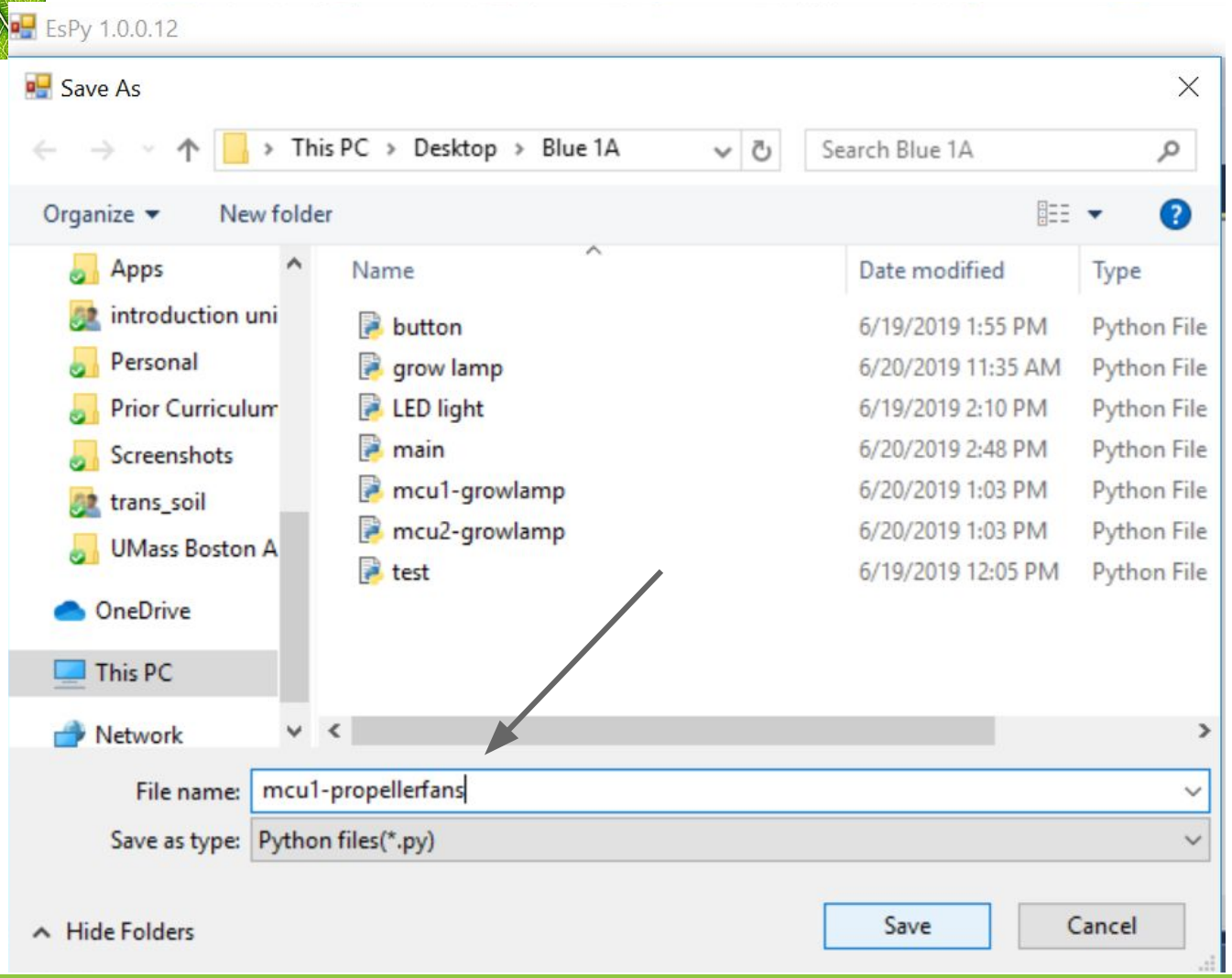


Table of Contents

```python
from actuators import Relay #NEW FOR TODAY
from sensors import TemperatureSensorPro
import time

tempsensor = TemperatureSensorPro(3)
relayfans = Relay(2) #NEW FOR TODAY

TempHigh = 80 #NEW FOR TODAY
HumidHigh = 60 #NEW FOR TODAY

while True:
    temp = tempsensor.get_temperature()
    humidity = tempsensor.get_humidity()

    if temp > TempHigh or humidity > HumidHigh: #NEW FOR TODAY
        relayfans.on()
        print("temperature is:", temp, "humidity is:", humidity)
        time.sleep(2)
    else:
        relayfans.off()
        print("temperature is:"), temp, "humidity is:", humidity)
        time.sleep(2)
```

Table of Contents

Exit Ticket

1. Did you get your propeller fans up and running? If not, what went wrong?

2. At what temperature will your fans come on? Why did you choose that temperature?

Table of Contents

Lesson 4c: Exhaust Fans

Table of Contents

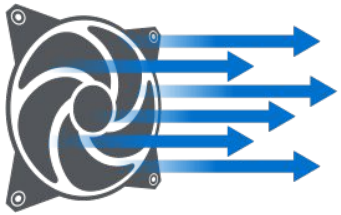
Lesson 4c: Adding Exhaust Fans

Images	Sensor/Device	Class	Ports
  	Temperature/Humidity Propeller Fans Exhaust Fans	Sensors Actuator Actuator	1, 2, 3

Table of Contents

Do Now (4c Servos):

1. Why is air flow so important in a greenhouse?



2. How do we get the hot or moist air OUT of the greenhouse?

Student Responses:

Table of Contents

Problem of the day:
How do I expel the hot or humid air from the greenhouse?

Solution: Have Exhaust Fans do it for you!!!

Table of Contents

Problem-Solving Process

Do plants like hot and humid air in their house?

Probably not if the temperature OR humidity is too HIGH

We want excessively hot and humid air out of the greenhouse!

What kind of device has been used for such a purpose?

Exhaust fans have been used on computers to effectively cool down really hot CPUs!

They must work for the greenhouse as well!

Table of Contents

Brainstorm:

- How can we get the hot or humid air out of the greenhouse?
- Can we wire a fan that sucks the air in and expels it outside the greenhouse?
- How do we wire the exhaust fan to the propeller fan so it happens simultaneously?

Table of Contents

Gather your materials…

2- terminal block		3- 4 Pin Wire Connector	
2- Black Exhaust Fans		1-	
2 long wires to connect your exhaust fans to your propellor fan		May need some washers to use to attach your magnets	

Table of Contents

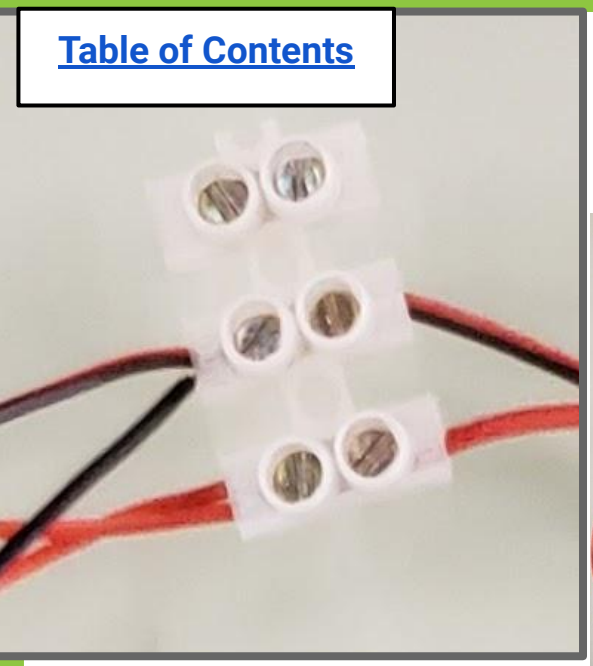
Wiring Exhaust Fans

You will be adding the exhaust fans to the current propeller fan setup.

No additional code is needed, because your propeller fan code will turn on your exhaust fans too.

You can simply insert the propeller fan wires into the original terminal block of your fans.

Table of Contents



Wiring Exhaust Fans

1: empty

2: BOTH Black wires from Exhaust fans

3: BOTH Red wires from Exhaust fans

4: empty

5: Black to get twisted with Black Prop fan

6: Red to get twisted with Red Prop fan

EsPy File name for Exhaust Fans

- You do not need to save a new file for the exhaust fans, since they are wired into the same setup as your propeller fans.

- Just open your propeller fan code (see next slide) and press play, everything should work together.

Table of Contents

Codes (No change from propeller fans):

```
from actuators import Relay
from sensors import TemperatureSensorPro
import time

tempsensor = TemperatureSensorPro(3)
relayfans = Relay(2)

TempHigh = 80
HumidHigh = 60

while True:
    temp = tempsensor.get_temperature()
    humidity = tempsensor.get_humidity()

    if temp > TempHigh or humidity > HumidHigh:
        relayfans.on()
        print("temperature is:", temp, "humidity is:", humidity)
        time.sleep(2)
    else:
        relayfans.off()
        print("temperature is:"), temp, "humidity is:", humidity)
        time.sleep(2)
```

Table of Contents

Exit Ticket

(4c exhaust fans)

1. Were you able to get your exhaust fans connected to your propeller fans? If not how come?

2. At what temperature and humidity will your propeller fan and exhaust fan come on? Why did you choose that temperature/humidity?

Table of Contents

Lesson 4d: Servos

Table of Contents

Lesson 4d: Servos

Image	Sensor/Device	Class	Ports
	Servo	Actuators	1, 2, 3

Table of Contents

Do Now (4d Servos):

1. What is the purpose of an exhaust fan?
2. Why is it important to let hot air or humid air out of the greenhouse?

Student Responses:

Table of Contents

Problem of the day:

What is another way to let hot or humid air out of the greenhouse without having to use fans?

Solution: Make the roof crack open when it's time to let hot air and humidity out by adding a **Servo**

Table of Contents

Problem-Solving Process

Do plants like hot and humid air in their house?

Probably not if the temperature OR humidity is too HIGH

We want excessively hot and humid air out of the greenhouse!

What kind of device has been used for such a purpose?

Opening the top of a greenhouse lets out hot air really quickly

We can install a servo to open and close the top door of the greenhouse!

Table of Contents

Brainstorm:

- How do we open the roof just a little bit?
- How will we close the roof once the “bad” air is out?
- How do we program it to do this based off the temperature and humidity sensors reading?
- How do you wire and connect the servo?

Table of Contents

Gather your Materials:

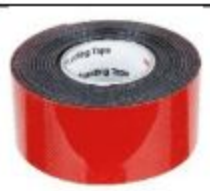
Number/Name of Part	Picture of Part	Number/Name of Part	Picture of Part
1- Wio Link Board (MCU).		1- Temp Sensor	
1- Micro-USB Cord (to plug in the MCU)		1- Shelf for servo (2 if using 2 servos)	
1- Relay with 2 red wires		1- Terminal block (if using 2 servos for your greenhouse)	
1- Tape		1- Pin Wire Connector (for temp sensor)	

Table of Contents

Installing your Servo to MCU 1

Plug the Servo into port 1 of MCU 1

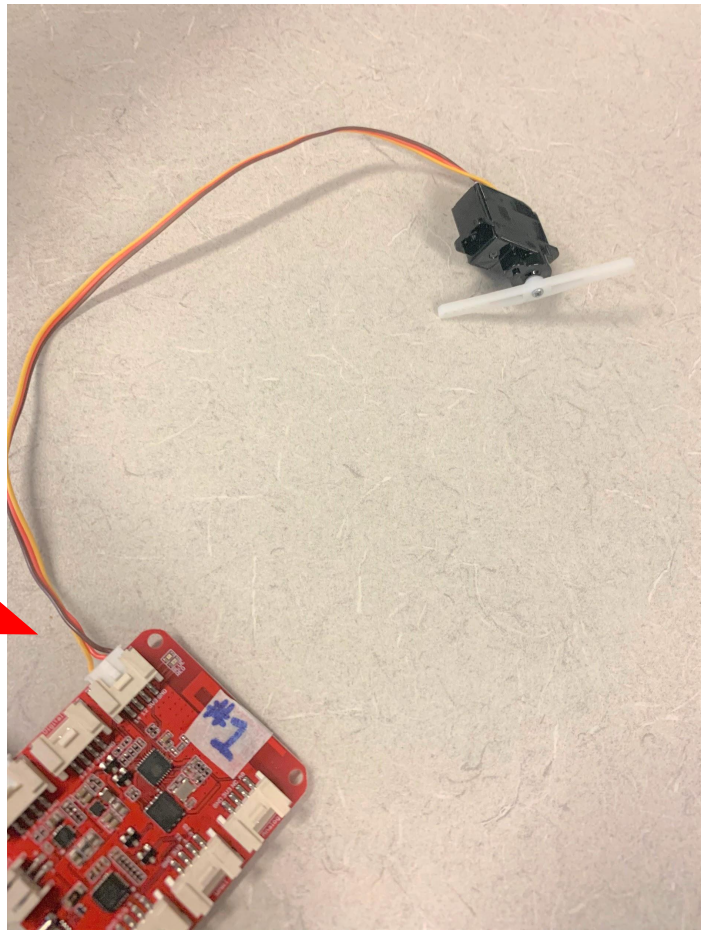


Table of Contents

Installing your Servo to MCU 1

Put the Servo on the wooden block with double sided tape

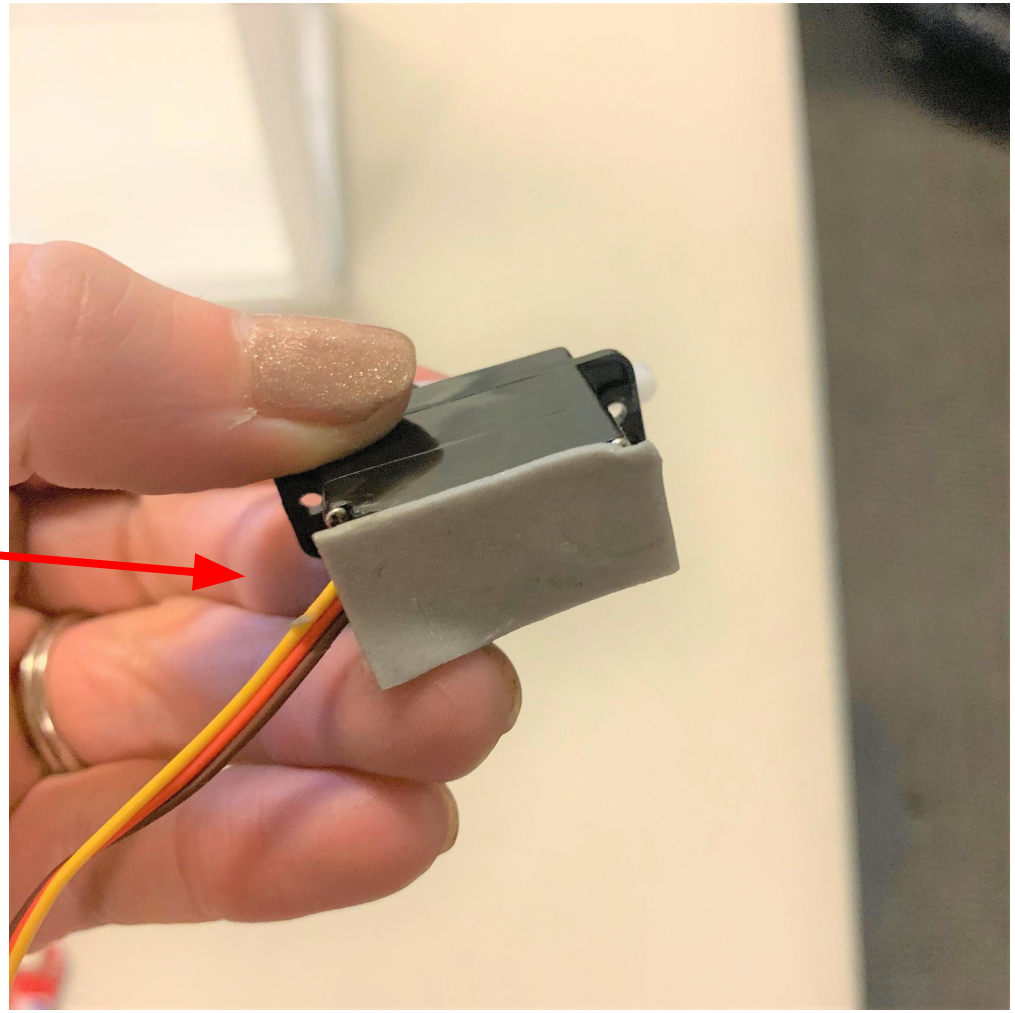


Table of Contents

Installing your Servo to MCU 1

Place the Servo on the top of the greenhouse like this →

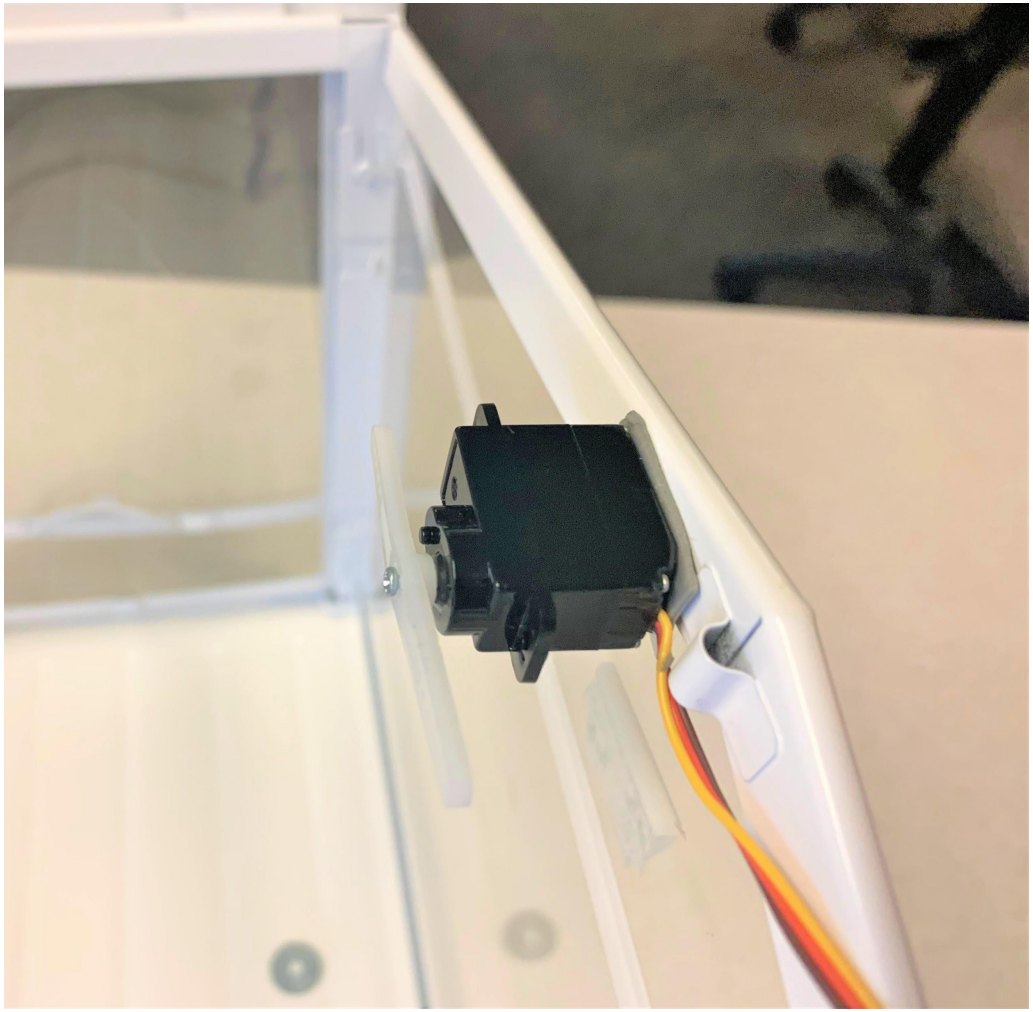
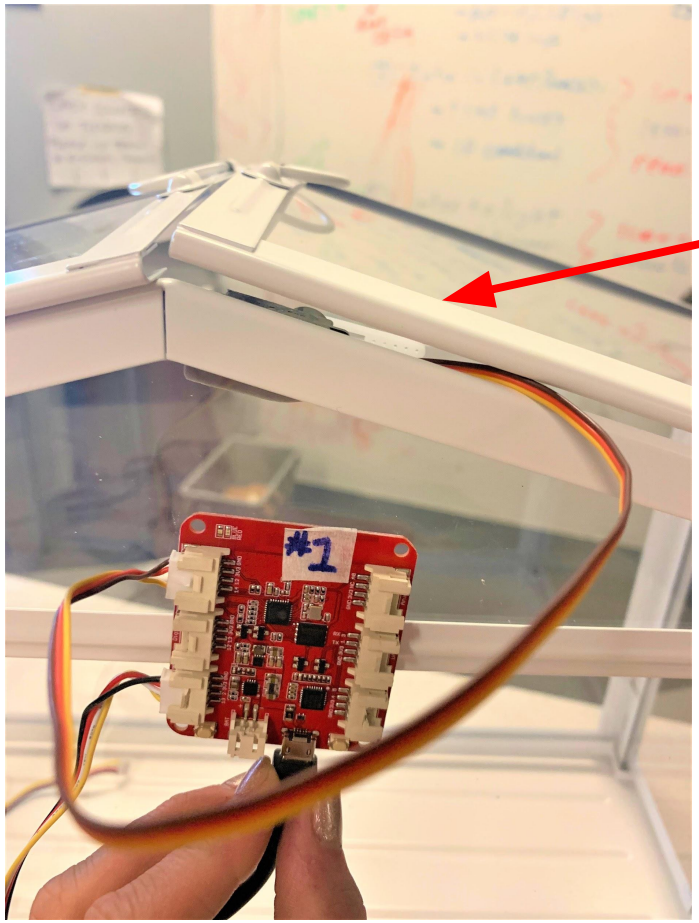


Table of Contents

Installing your Servo to MCU 1



Close roof

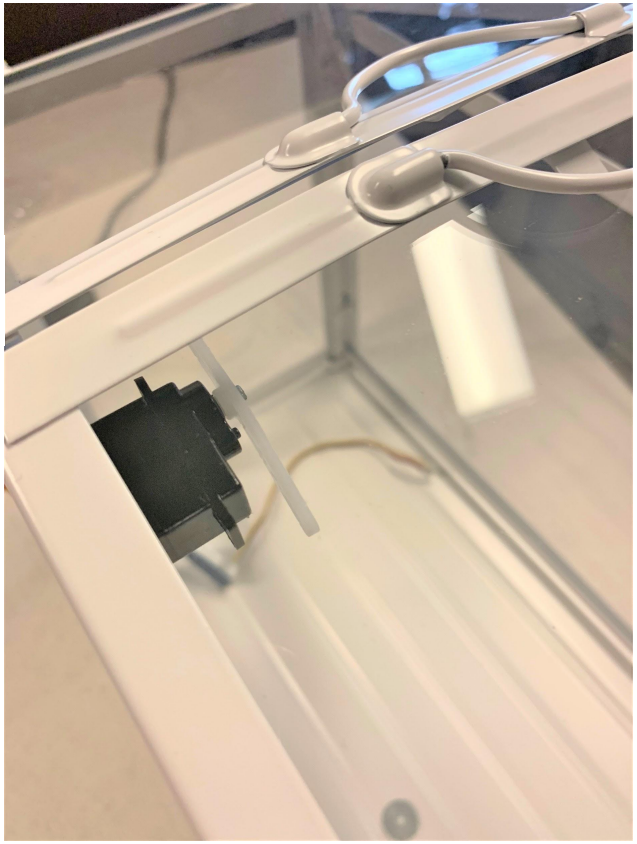
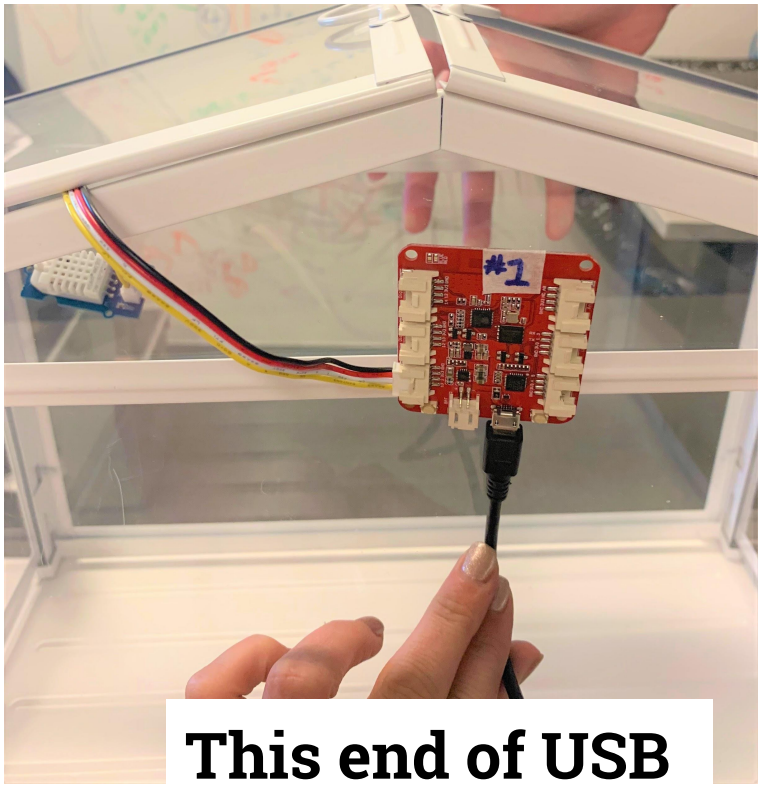


Table of Contents

Installing Your Servo



This end of USB cord plugs into the computer

Table of Contents

Once connected to the computer, the **red light** on your MCU #1 should turn on!

Now you are ready to code…

*If you see a **blue light**, reflash your MCU. If no luck, get another MCU and re-label it.*

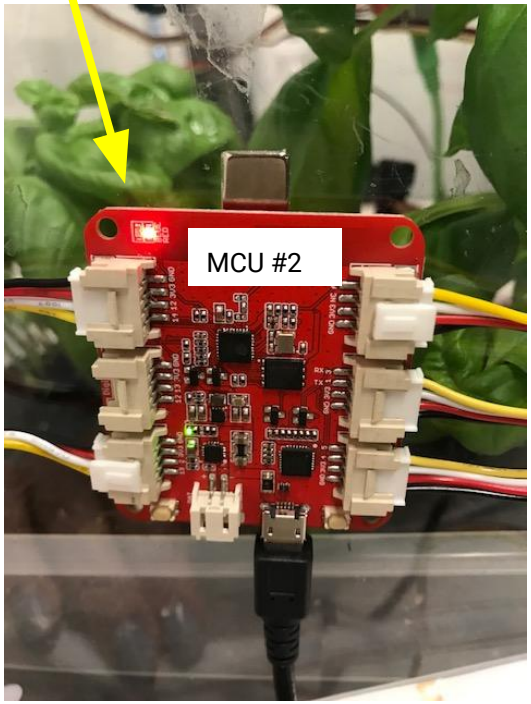


Table of Contents

Open EsPy from the desktop

For reminders on how to complete each step of the saving process, click **HERE.**

Click this from desktop

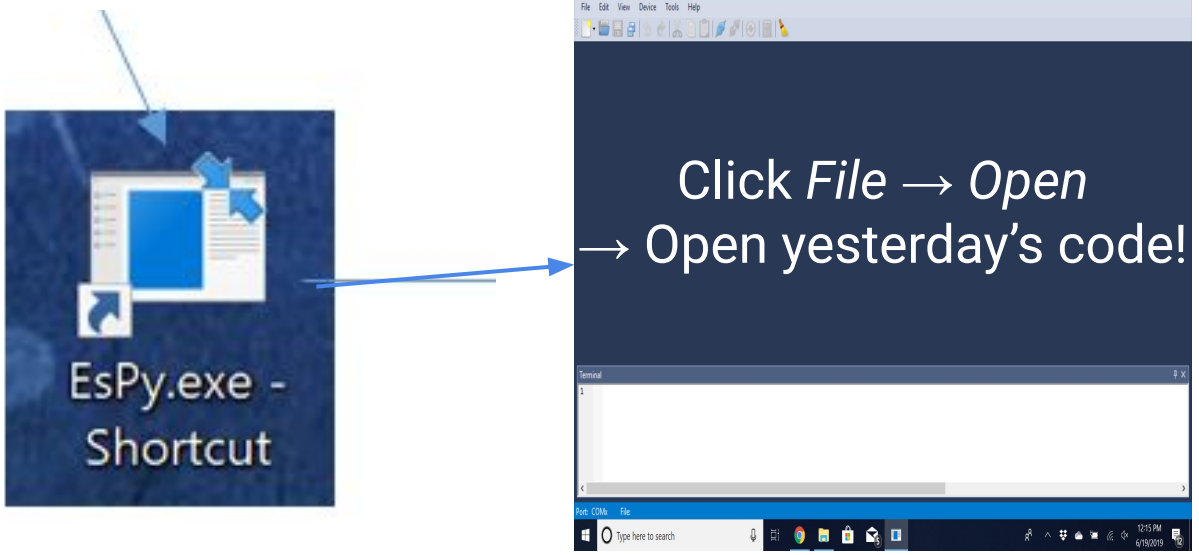


Table of Contents

Code for Servo

```
1   from actuators import Relay
2   from actuators import Servo #NEW FOR TODAY
3   from sensors import TemperatureSensorPro
4   import time
5
6   tempsensor = TemperatureSensorPro(3)
7   relayfans = Relay(2)
8   servo = Servo(1, position=0) #NEW FOR TODAY
9
10  TempHigh = 80
11  HumidHigh = 60
12
13  while True:
14      temp = tempsensor.get_temperature()
15      humidity = tempsensor.get_humidity()
16
17      if temp > TempHigh or humidity > HumidHigh:
18          relayfans.on()
19          servo.set_position(90) #NEW FOR TODAY
20          print("temperature is:", temp, "humidity is:", humidity)
21          time.sleep(2)
22      else:
23          relayfans.off()
24          servo.set_position(0) #NEW FOR TODAY
25          print("temperature is:"), temp, "humidity is:", humidity)
26          time.sleep(2)
```

Table of Contents

Save updated EsPy file your class folder as “MCU1-Servo”

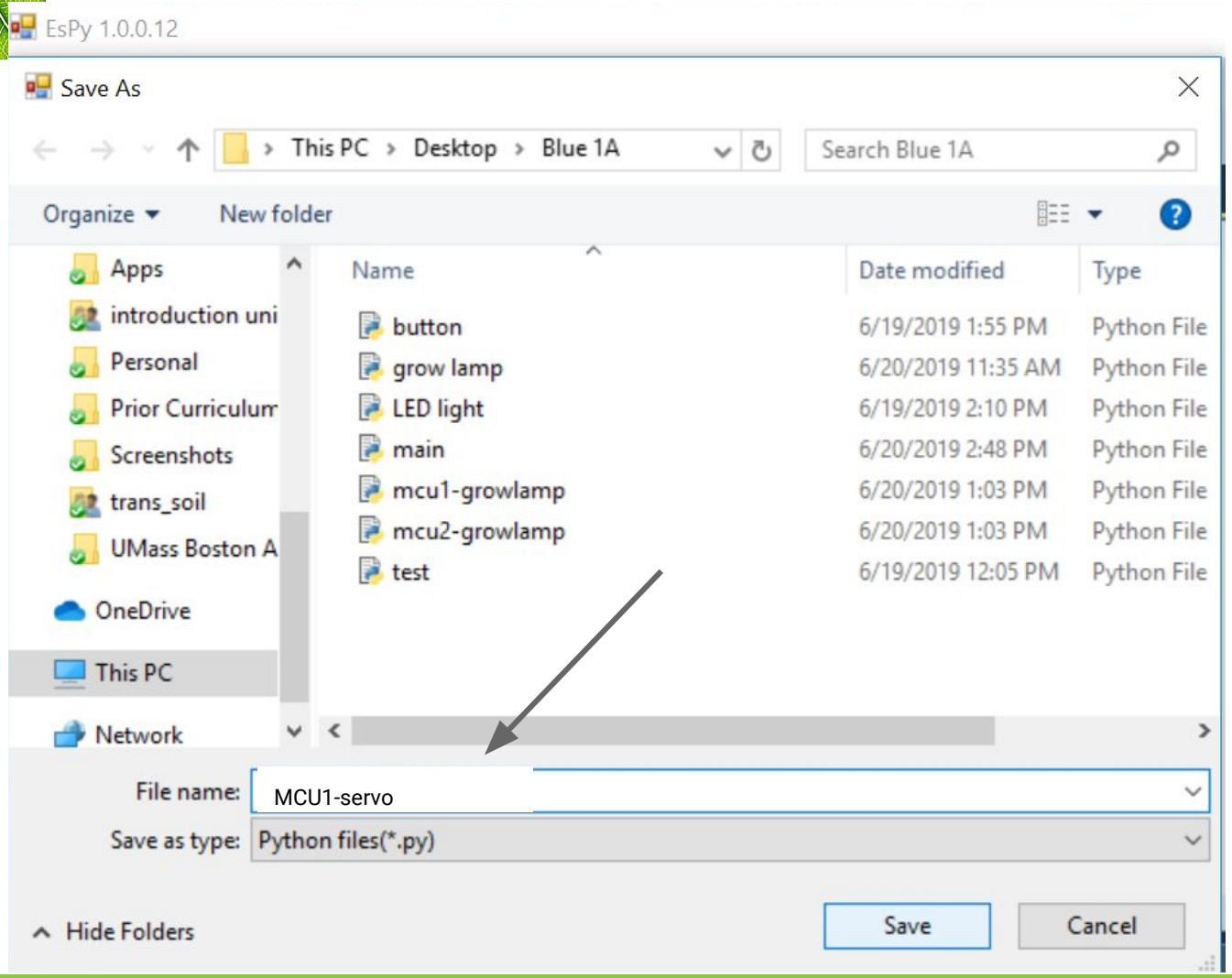
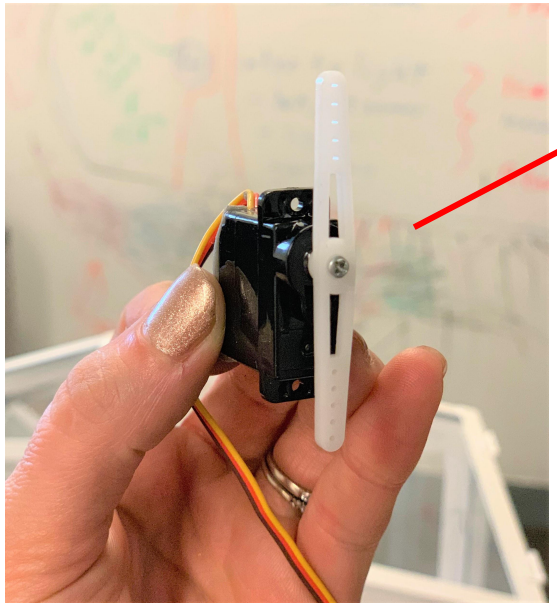


Table of Contents

Servo: What you should see!

When Servo is at a vertical 90 degree angle, the roof will be open



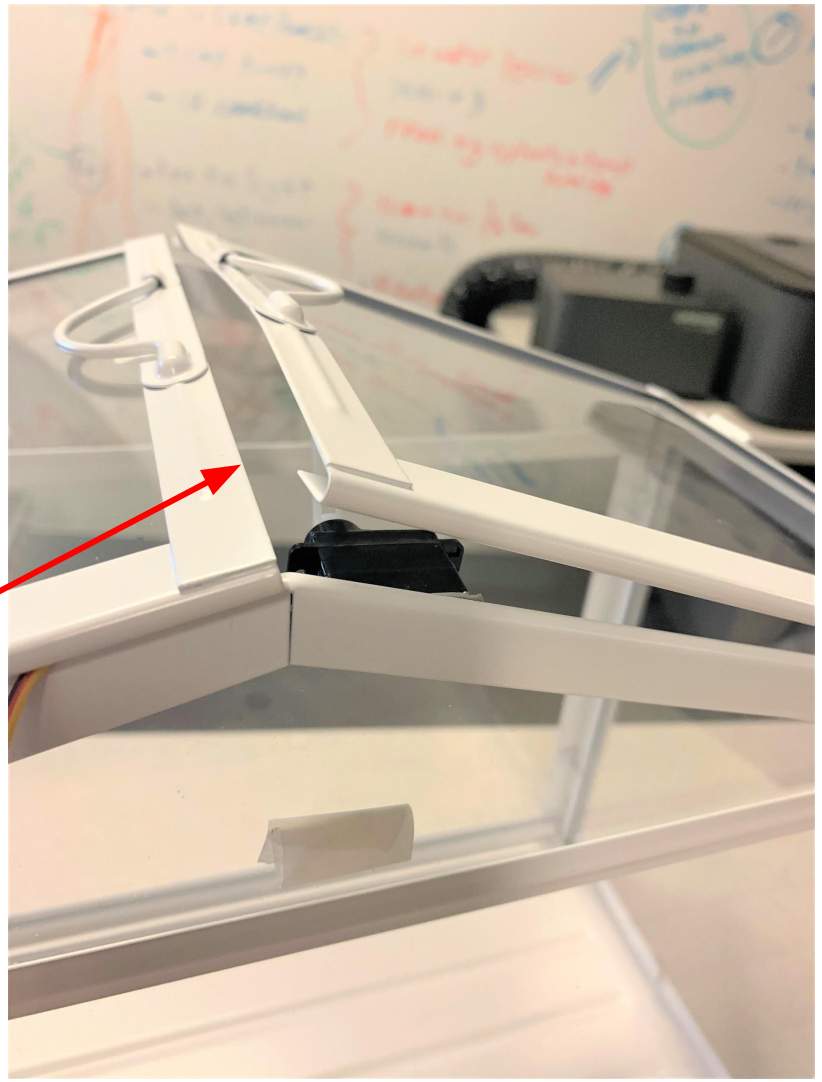
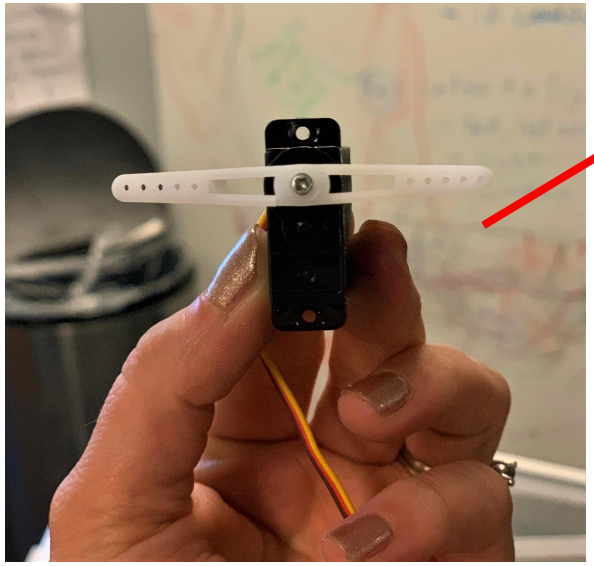


Table of Contents

Servo: What you should see!

When Servo is at a Horizontal 180 degree angle, the roof will be closed



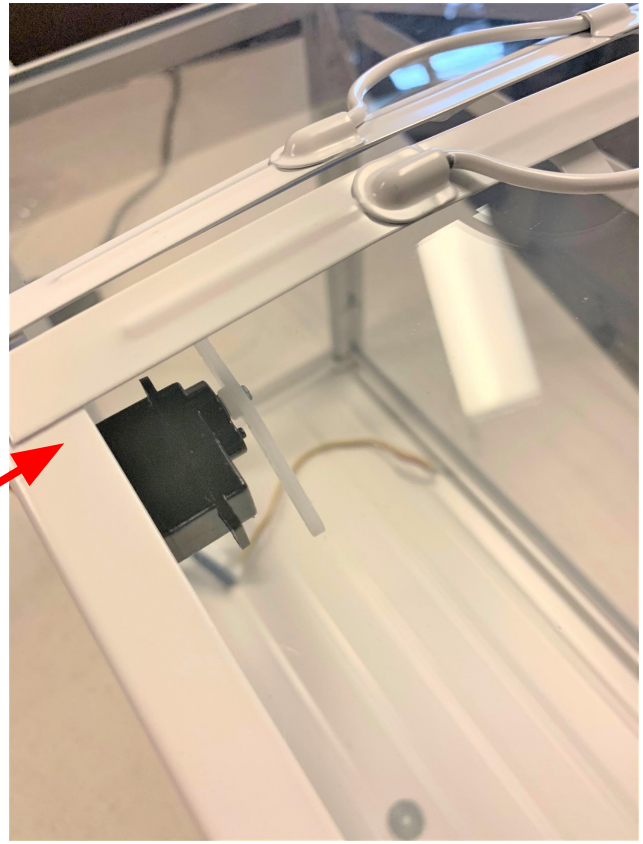


Table of Contents

Exit Ticket (4d servo)

1. Did you get you Servo up and running? If not how come?
2. At what angle and temperature is your roof cracked open?
3. At what angle and temperature is your roof totally shut?

Table of Contents

Lesson 5: Moisture Sensor

Table of Contents

Lesson 5: Moisture Sensor

Image	Sensor/Device	Class	Ports
	Moisture Sensor	Sensors	4 (only)

Table of Contents

Do Now (5 Moisture Sensor):

1. How do you plan to water your soil plants when you aren't around to do it?
2. How do you know how often to water your plants?
3. How can you tell when the soil is too dry?

Table of Contents

Problem of the day: How to automatically water plants when the soil is too dry!

Solution: Install a moisture sensor and automated drip system!!

Table of Contents

Problem-Solving Process

Plant need a certain amount of water access to survive

You may not always be around to continuously water your plants

We can install a sensor to track the moisture level of soil.

We can install a water pump with a drip system in the soil

The moisture sensor can be coded to turn on the pump/drip system when the water levels are low

Now your plants can be happily watered without you present!

Table of Contents

Brainstorm:

- Where do we put the moisture sensor?
- How do we instal the drip system?
- How do we know what a good moisture level is?
- How do we connect the pump and sensor to the MCU to code it?

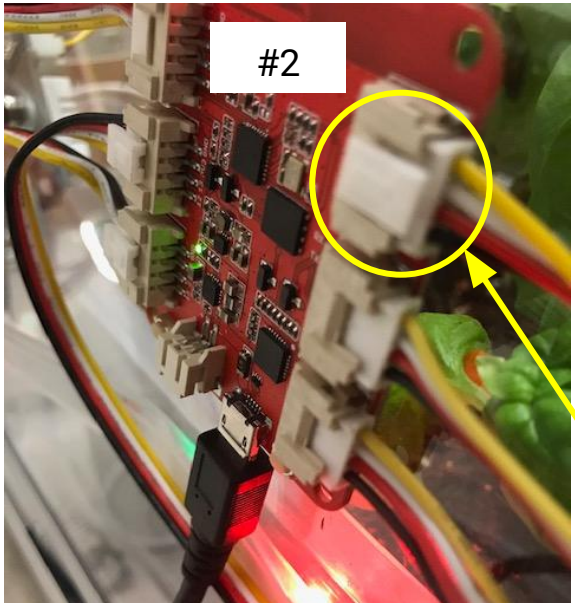
MATERIALS

Number/Name of Part	Picture of Part	Number/Name of Part	Picture of Part
1- Wio Link Board (MCU).		1- Water Pump	
1- Micro-USB Cord (to plug in the MCU)		1- Micro-USB Cord Stripped (wires exposed)	
1- Relay with 2 red wires		1- Screwdriver	
1- terminal block		1 Four Pin Wire Connector	
May need a pair of wire strippers (unlikely)		1 –Moisture sensor	
1 water pump with stripped wires		Tubing (about 8 inches) with 2-3 T connectors and 3-4 drip connectors	
1 container for your water		1 - LED strip	

Table of Contents

Moisture Sensor Setup

Attach the 4 pin connector wire to the moisture sensor, and plug in to **port 4** on MCU #2. Put the sensor into the dirt, keeping the blue above the ground.



Plug in to top of moisture sensor

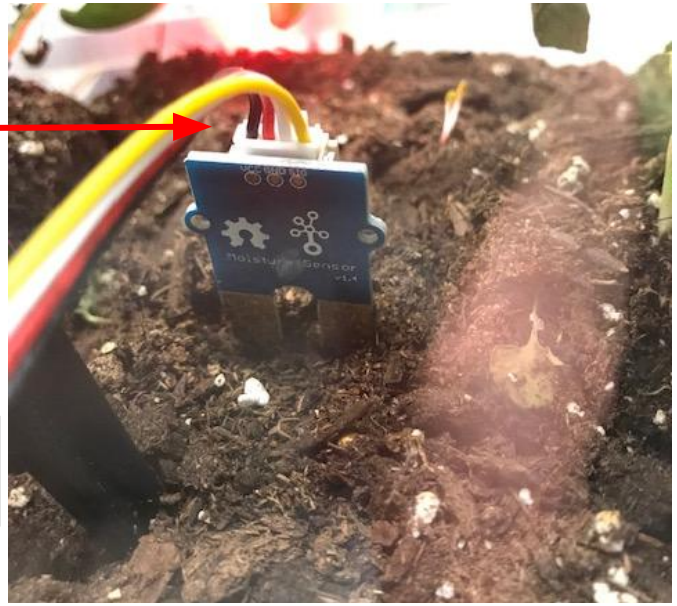
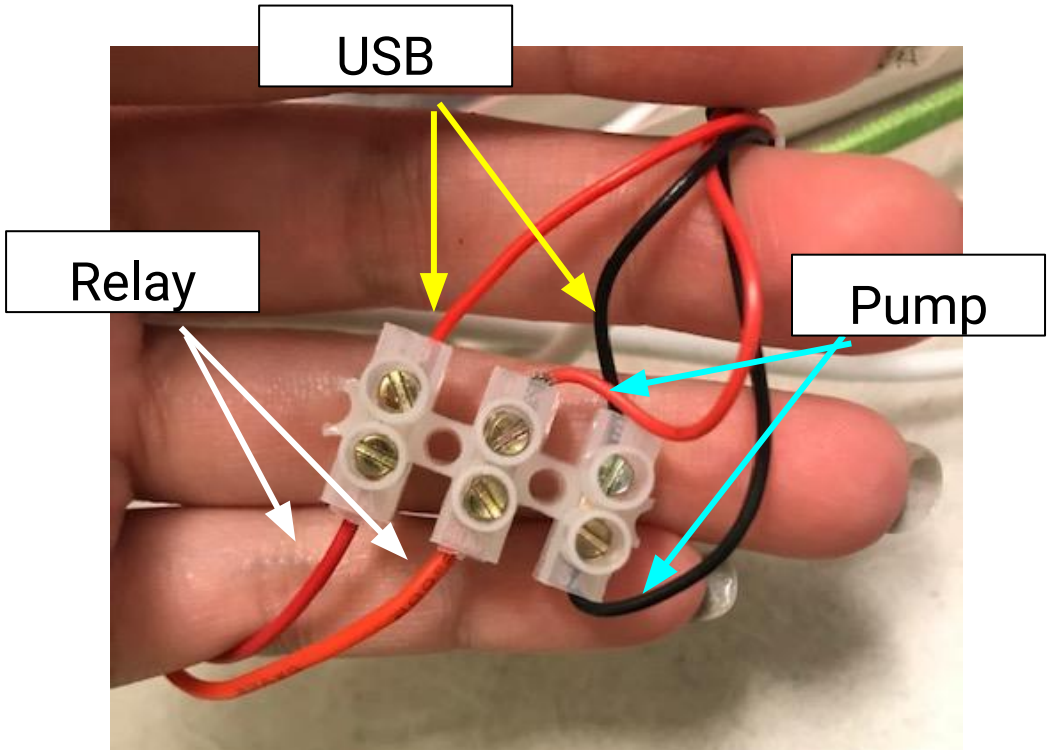
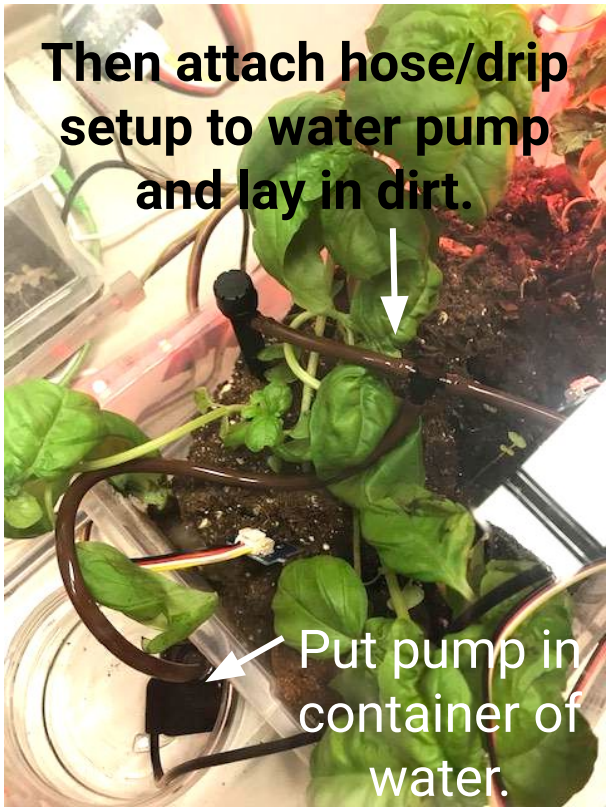


Table of Contents

Pump Setup

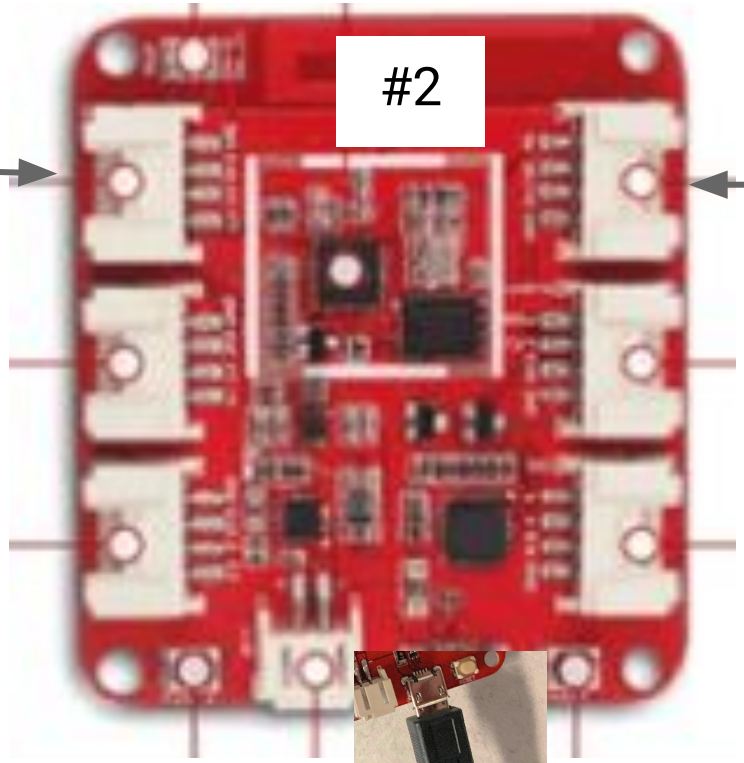
Just like for the fans/lights, take a USB cable, the water pump, and a relay, and wire into a terminal block like this.





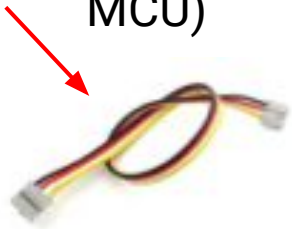
Connecting pump to MCU

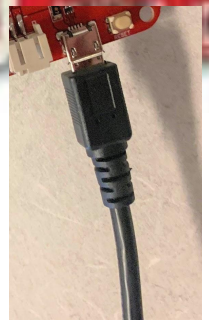
Relay for water pump



Moisture sensor

(Use a 4-pin connector to connect relay to MCU)





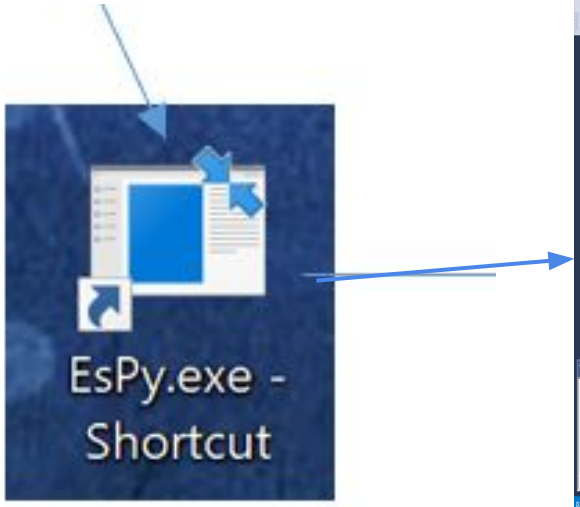
← Connect black USB here to the computer

Table of Contents

Open EsPy from the desktop

For reminders on how to complete each step of the saving process, click **HERE.**

Click this from desktop



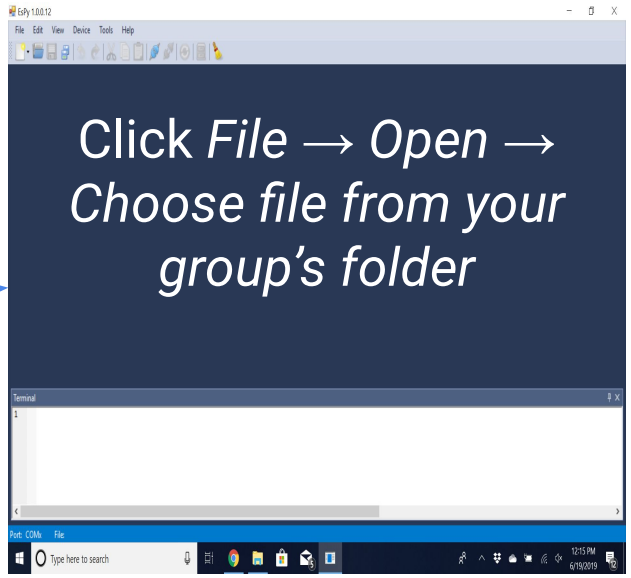


Table of Contents

Code for water pump (MCU 2) + LED strip

Save in your folder as: “waterpump-mcu2”

```
from actuators import Relay
from sensors import MoistureSensor #NEW FOR TODAY
from displays import GrowLight
import time

MoistHigh = 500 #NEW FOR TODAY
MoistLow = 300 #NEW FOR TODAY

ms = MoistureSensor(4) #NEW FOR TODAY
relaywater = Relay(1) #NEW FOR TODAY
relaylight = Relay(2)
gl = GrowLight(5)

while True: #NEW FOR TODAY
    m = ms.get_moisture() #NEW FOR TODAY
    relaylight.on()
    time.sleep(4)
    relaylight.off()
    time.sleep(4)

    if m > MoistHigh: #NEW FOR TODAY
        relaywater.off()
        gl.blink(color=[0,0,255])
        print("Moisture Level is HIGH", m)
        time.sleep(2)
    elif MoistLow < m < MoistHigh: #NEW FOR TODAY
        relaywater.on()
        gl.blink(color=[0,255,0])
        print("Moisture Level is GOOD", m)
        time.sleep(2)
    else: #NEW FOR TODAY
        relaywater.off()
        gl.blink(color=[255,0,0])
        print("Moisture Level is LOW", m)
        time.sleep(2)
```

Table of Contents

Save new EsPy file your class folder as “MCU2-Moisture”

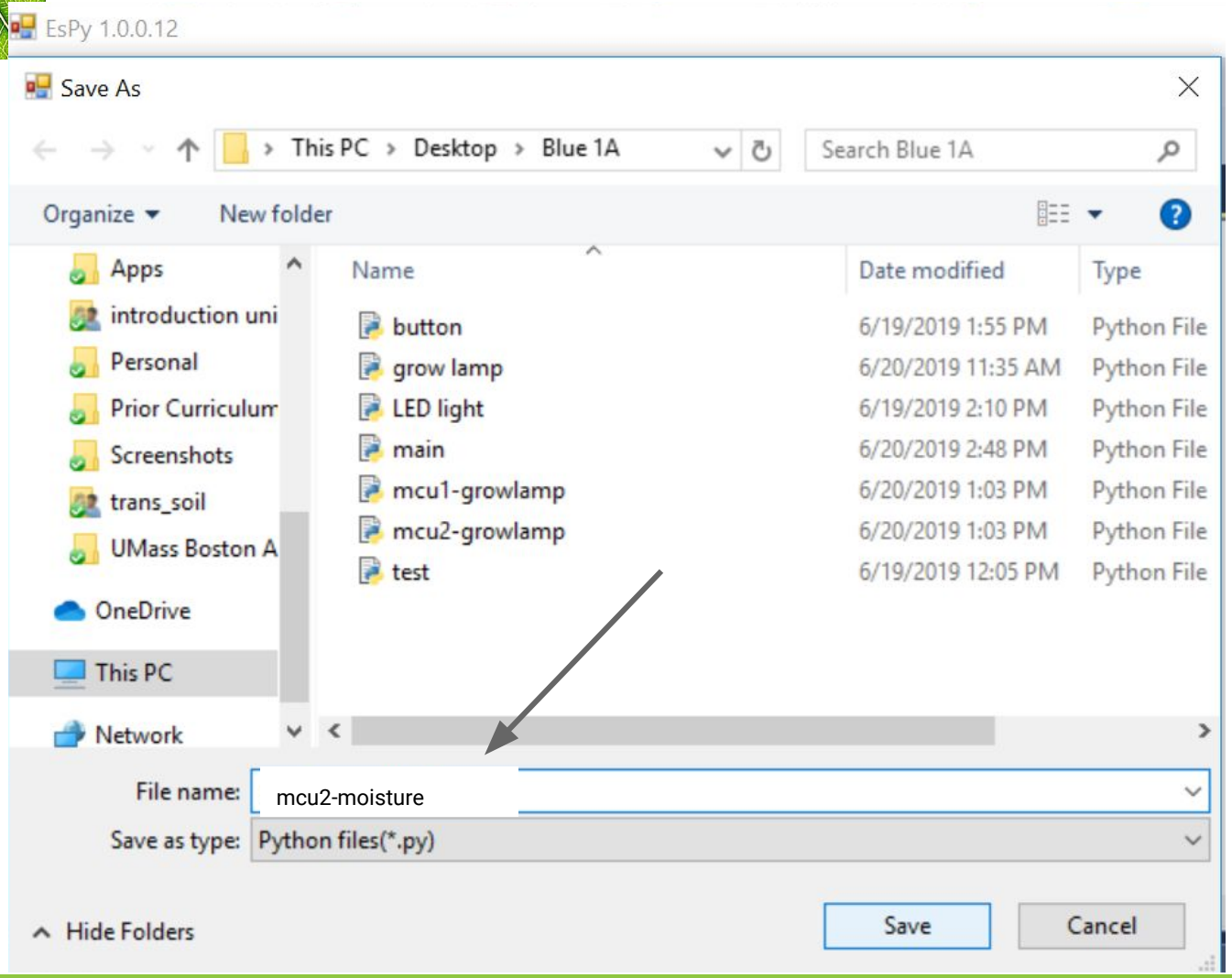


Table of Contents

Exit Ticket

(5 moisture sensor)

1. What is one problem that could arise with your moisture sensor?

2. Why do you think there is a minimum moisture level for the sensor (instead of it turning on when it's below a certain number)?

Table of Contents

Reference Section

For quick help when you need it!

- How to open & save an EsPy file
- How to upload a file as “main”
- How to reflash your MCU
- Wiring setups

Table of Contents

How to Upload a File as “Main”

Required to save the code to the MCU, so it can be unplugged from the computer (& plugged into another power source) and still run the code.

Table of Contents

How to Upload a File as "Main"

1. After your code is saved under its usual name ex. "MCU2-growlamp", you can go to File → Save as...

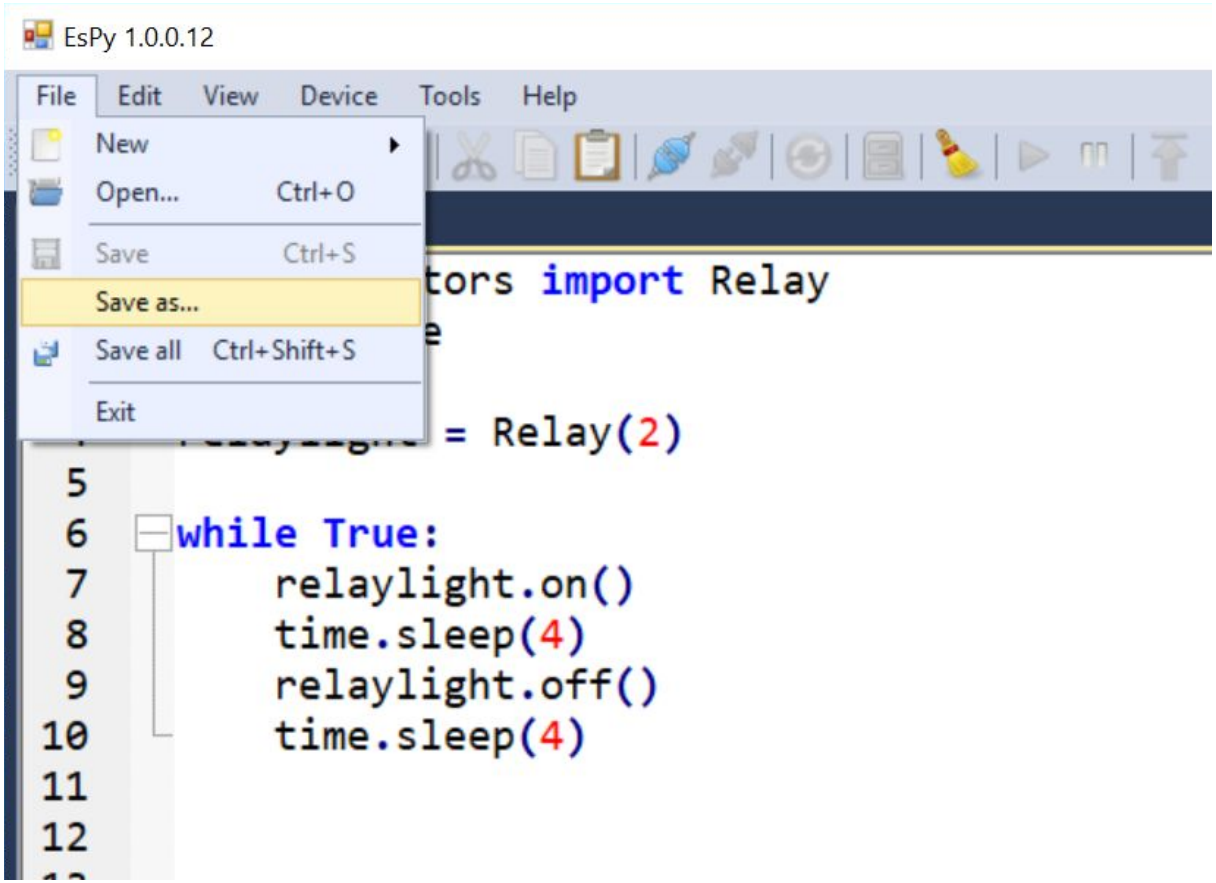


Table of Contents

How to Upload a File as "Main"

2. Double click on your group folder. Type "main" as the new file name.

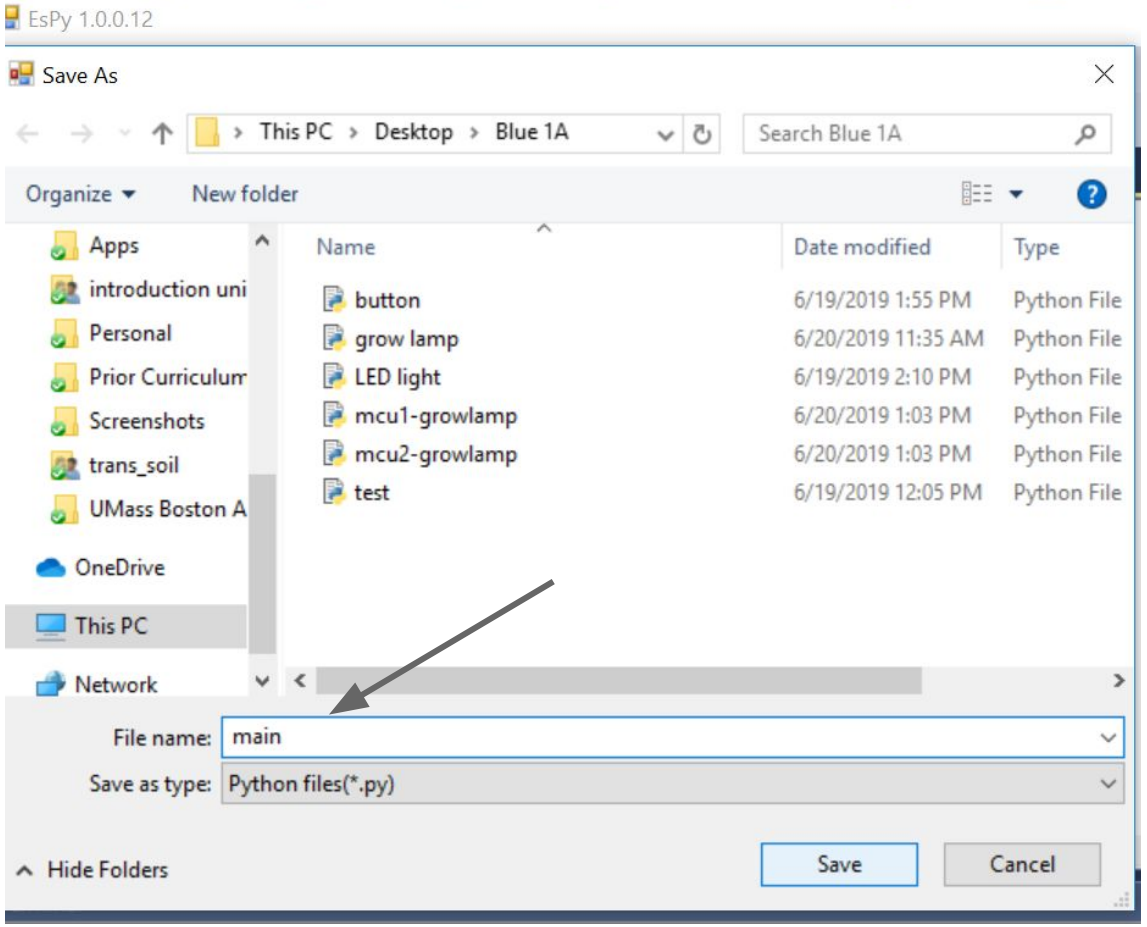
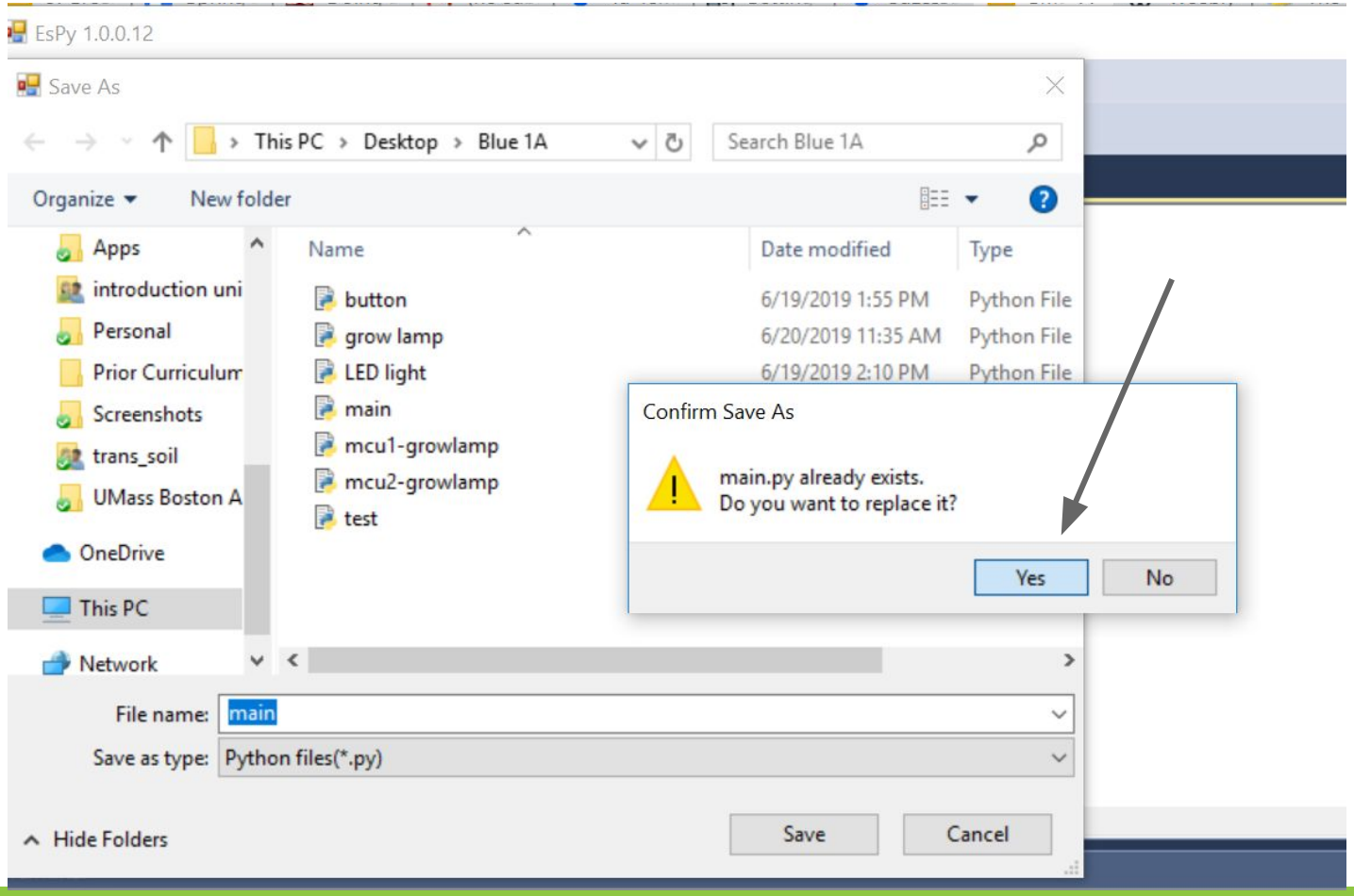


Table of Contents

How to Upload a File as "Main"

2b. If it tells you that the file already exists, and ask if you want to replace it, click "Yes"



How to Upload a File as "Main"

3. Click "Connect"

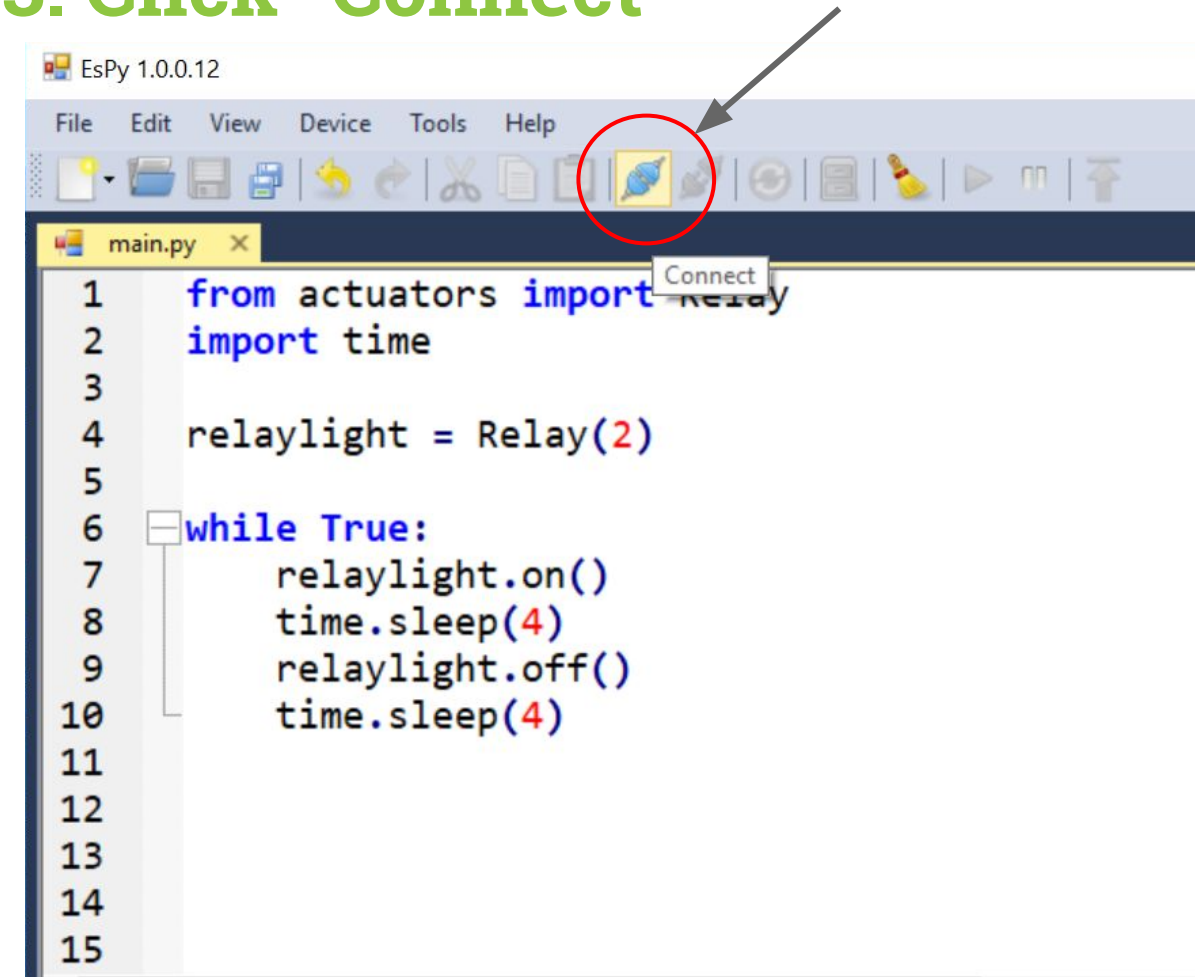


Table of Contents

How to Upload a File as "Main"

3. Instead of "Run," click "Upload"

EsPy 1.0.0.12

File Edit View Device Tools Help

main.py

Upload to the current directory

```
1  from actuators import Relay
2  import time
3
4  relaylight = Relay(2)
5
6  while True:
7      relaylight.on()
8      time.sleep(4)
9      relaylight.off()
10     time.sleep(4)
11
12
13
14
15
```

Table of Contents

How to Upload a File as “Main”

4. Now unplug your MCU’s black cable from the computer, and plug it into your USB power hub to see if your upload worked! (Your greenhouse should turn back on if the upload was successful)

Unplug MCU from computer

Plug in to USB power hub!

Table of Contents

How to Re-Flash an MCU

Required when a file has been saved as “main” in order to test new codes.

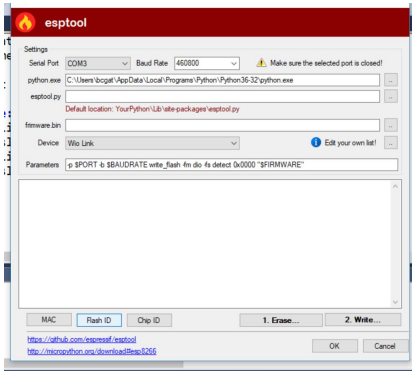


Table of Contents

How to Re-Flash an MCU

1. Make sure your MCU is plugged into the computer, and an EsPy window is open.

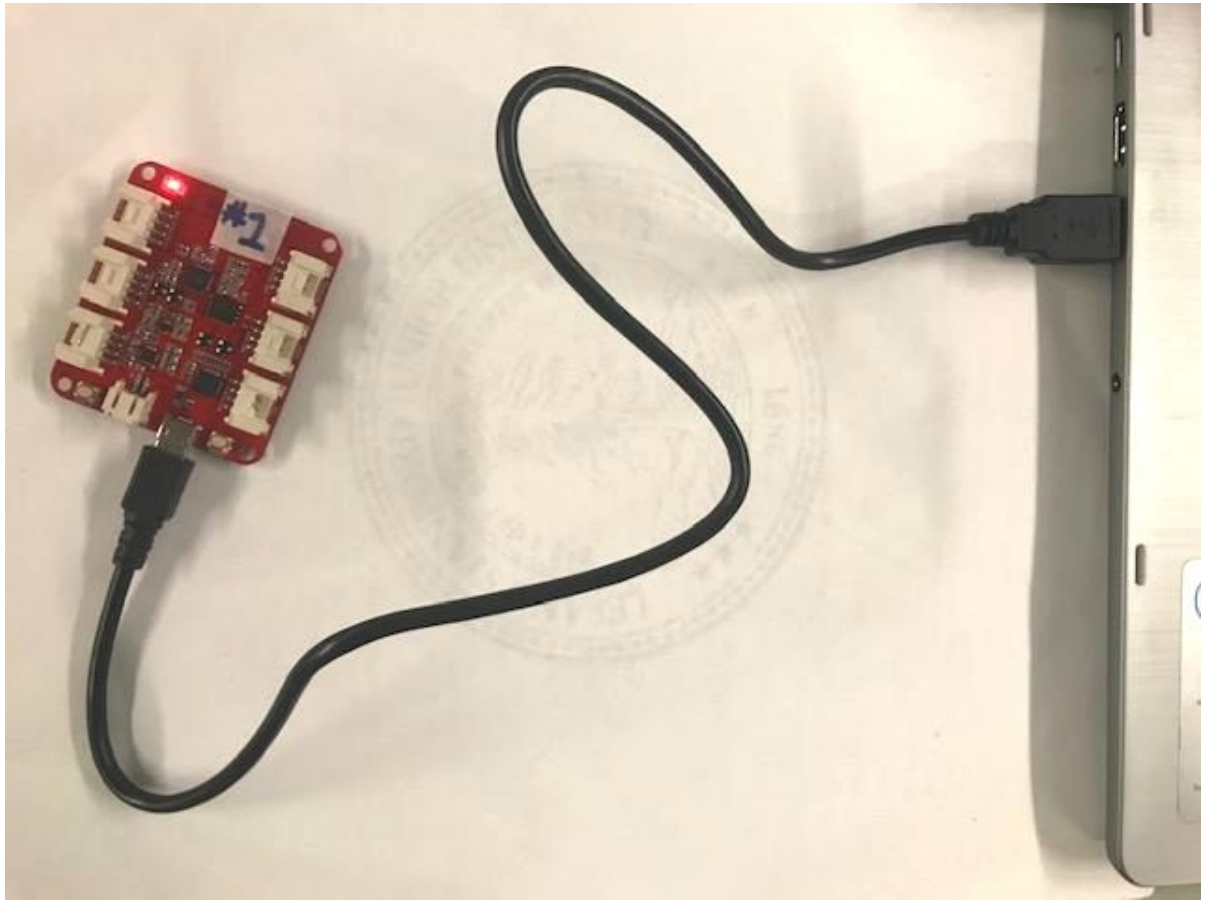


Table of Contents

How to Re-Flash an MCU

2. Click Device → EspTool...

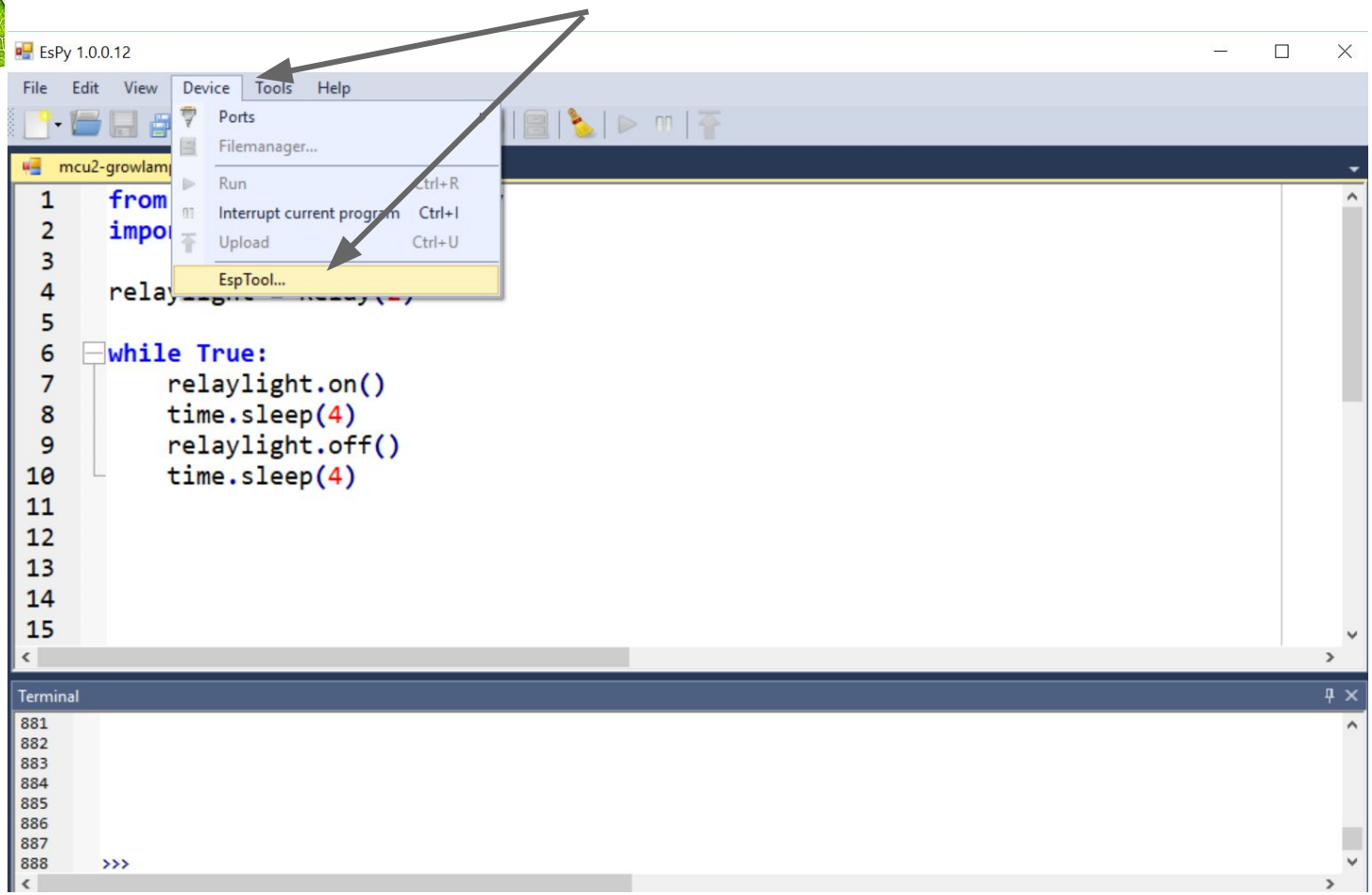


Table of Contents

How to Re-Flash an MCU

3. Click “Flash ID”

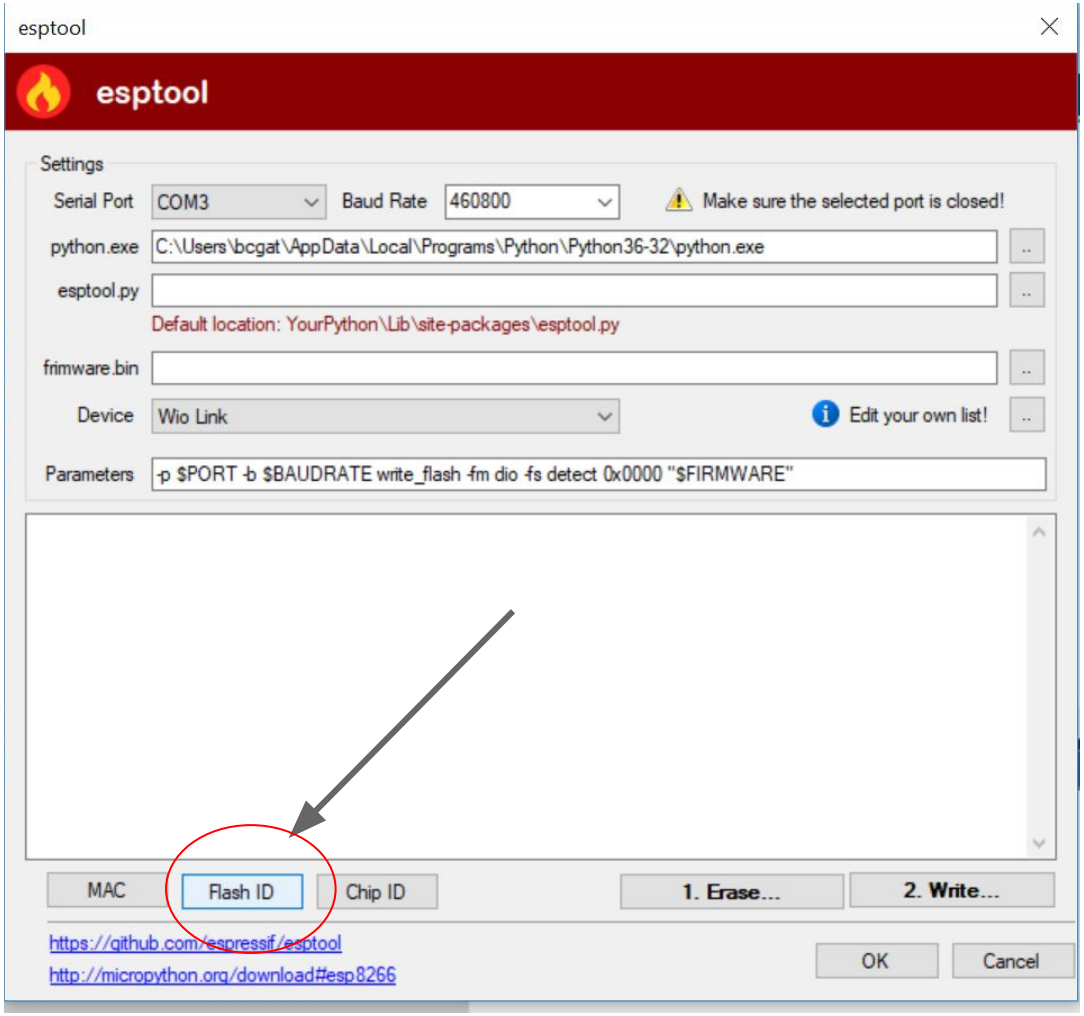


Table of Contents

How to Re-Flash an MCU

4. Click “1. Erase…”and wait 10 seconds

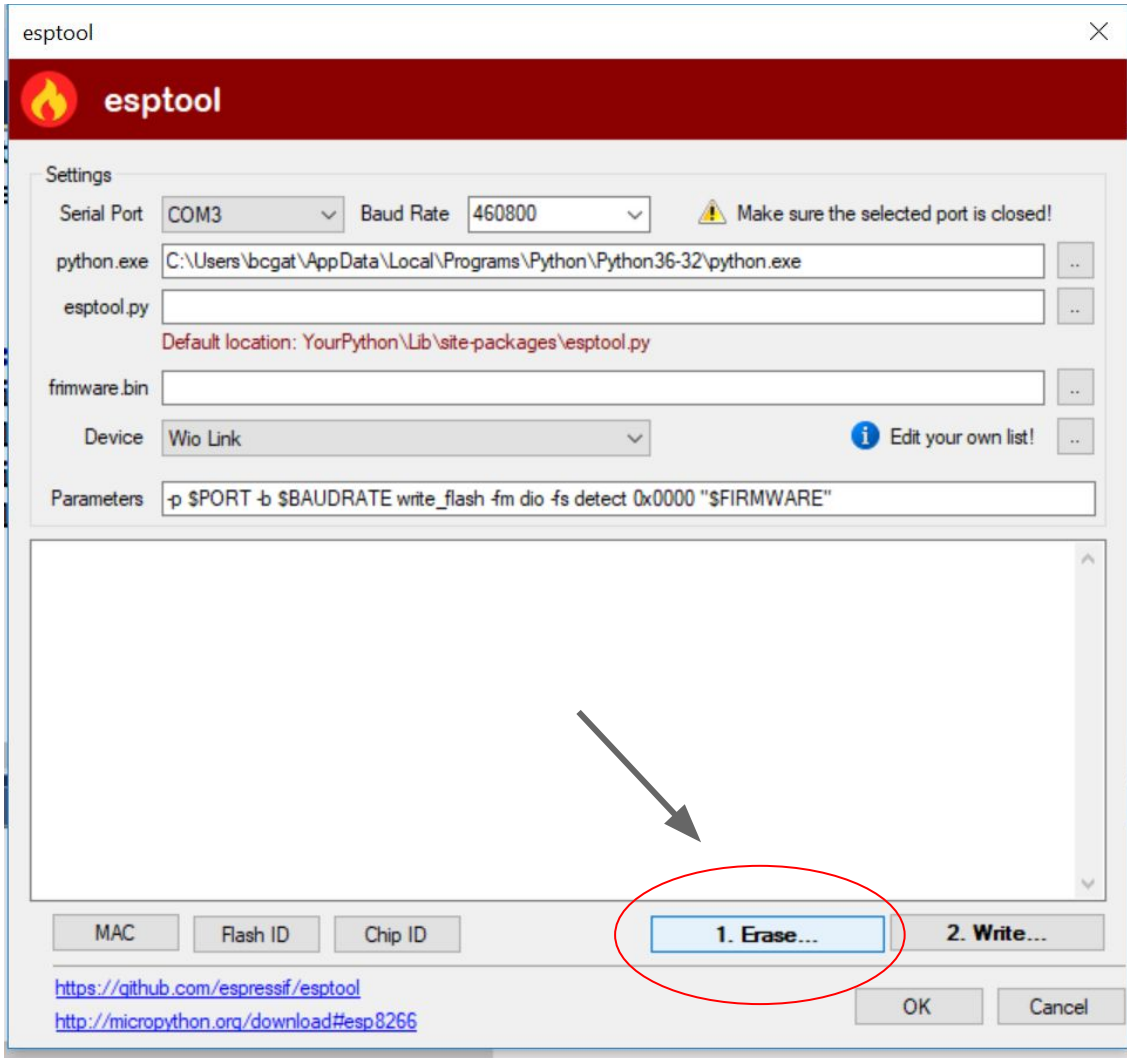


Table of Contents

How to Re-Flash an MCU

4. Click “2. Write…”and wait 10 seconds

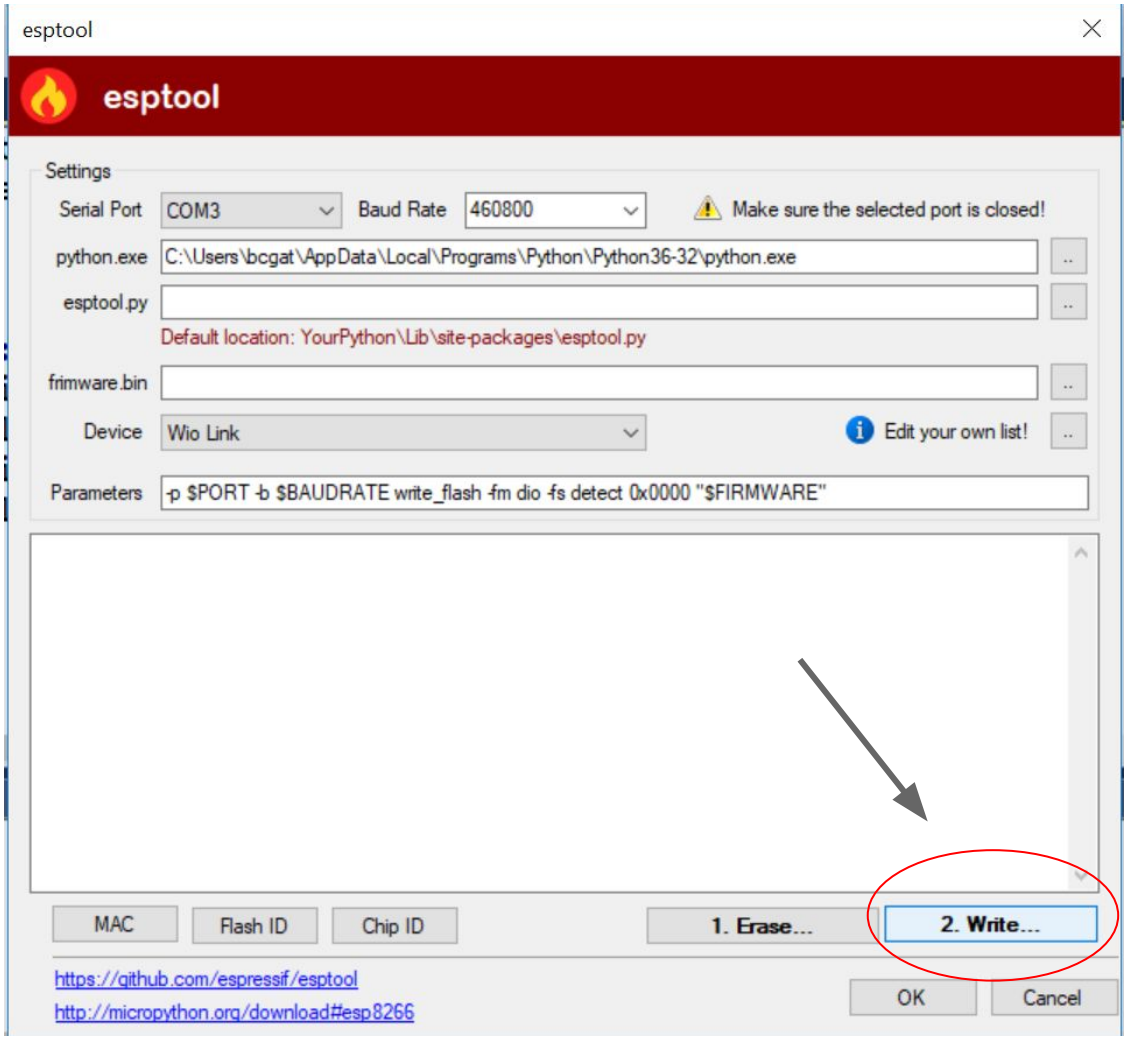


Table of Contents

How to Re-Flash an MCU

5. Click "Ok" and you're all set!

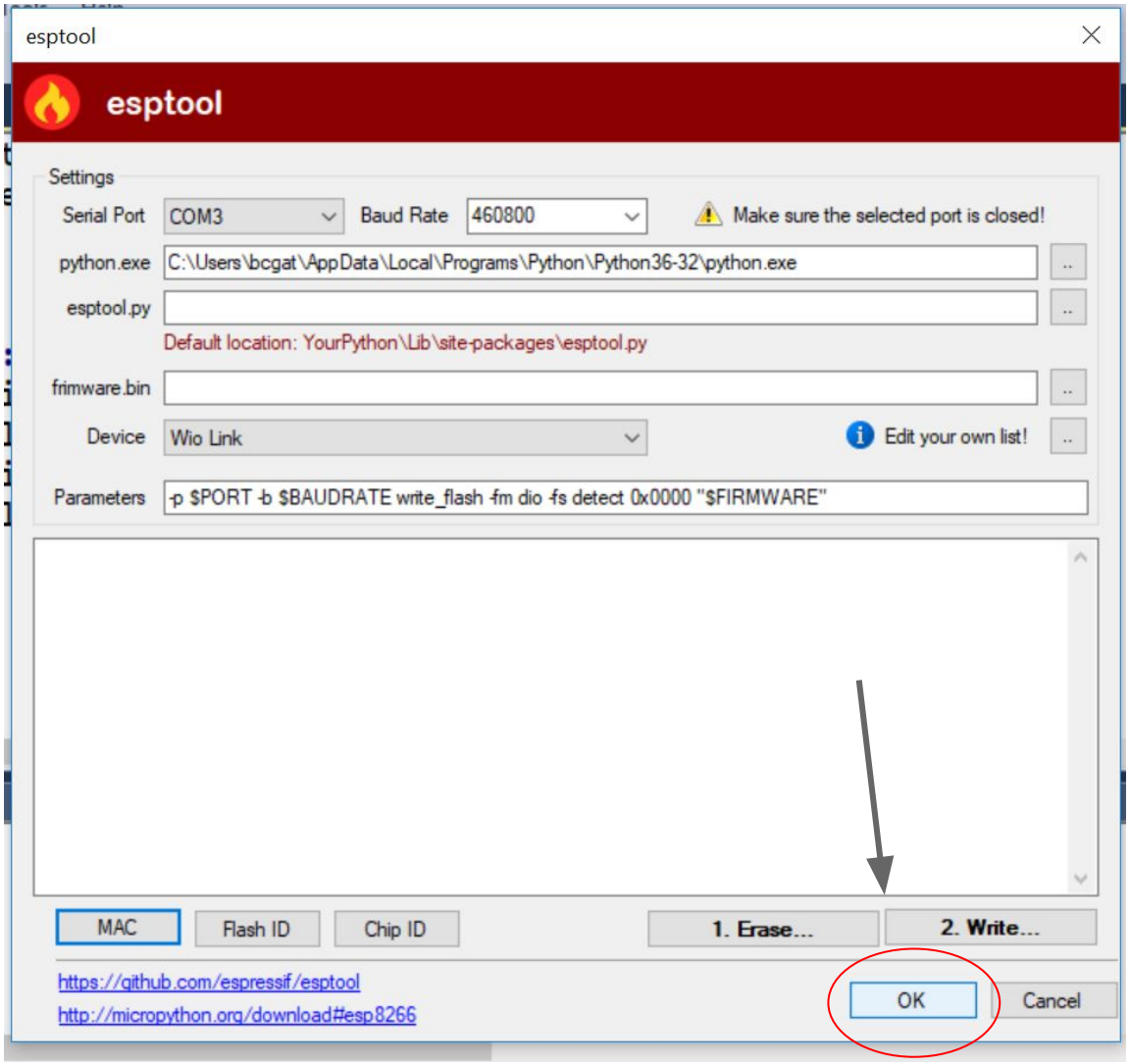


Table of Contents

Wiring Setups

Images of final setups after proper wiring!

- Terminal block labeling
- Grow Lamp wiring setup
- Propeller Fan wiring setup
- Exhaust Fan wiring setup

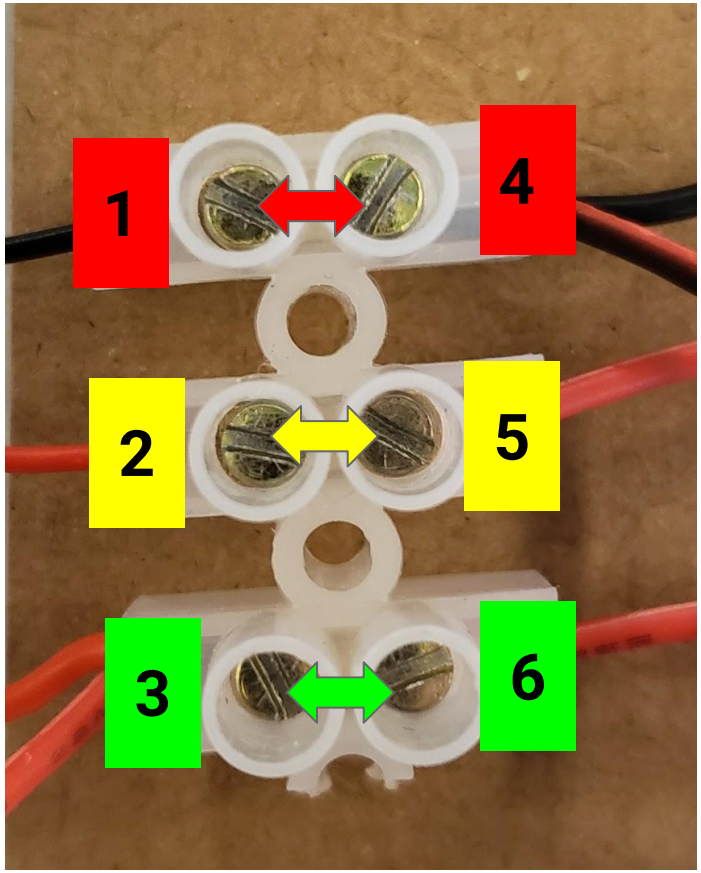
Table of Contents

Terminal Block

A terminal block provides a way to securely connect two wires.

The wire in location 1 is now connected with the wires in location 4.

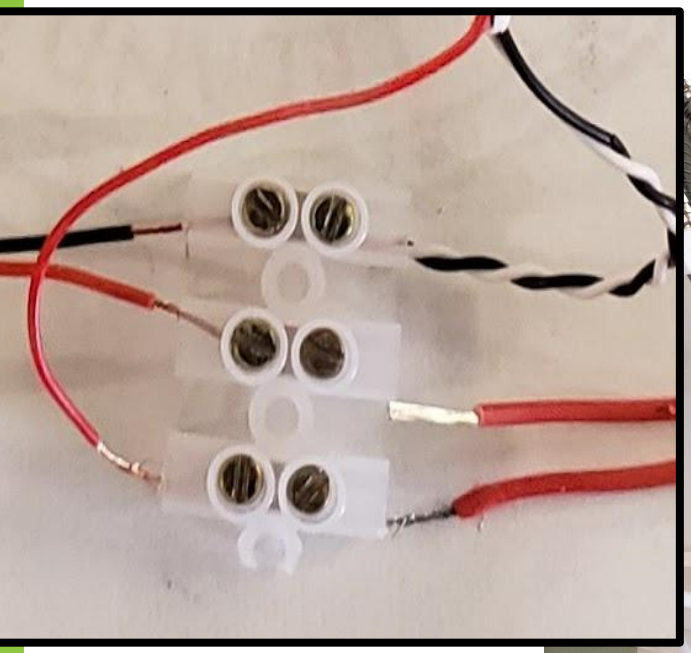
Make sure the screw is pressing on the wire, not the plastic.



The arrows ⟷ represent which wires are now connected through a circuit

Table of Contents

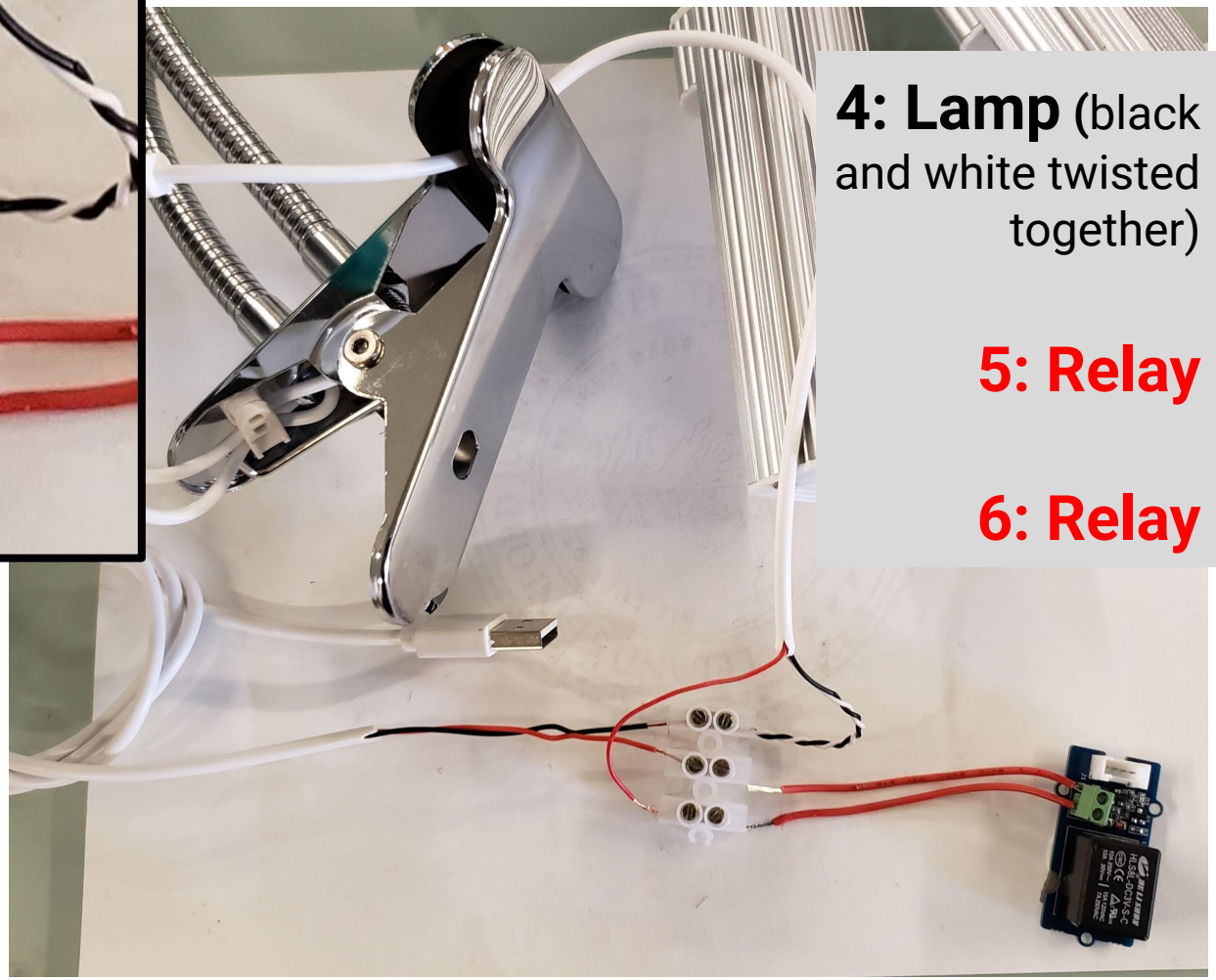
Wiring the Grow Lamps



1: USB Power

2: USB Power

3: Lamp



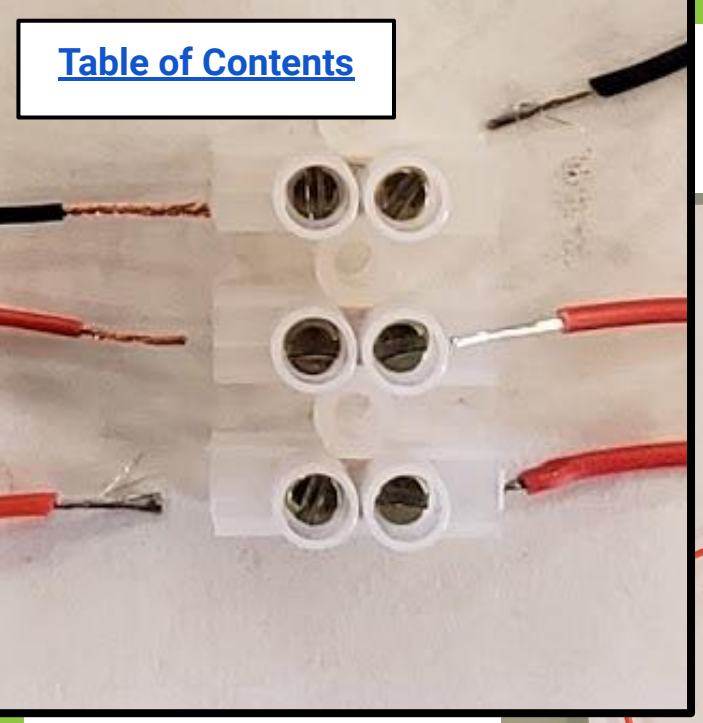
4: Lamp (black and white twisted together)

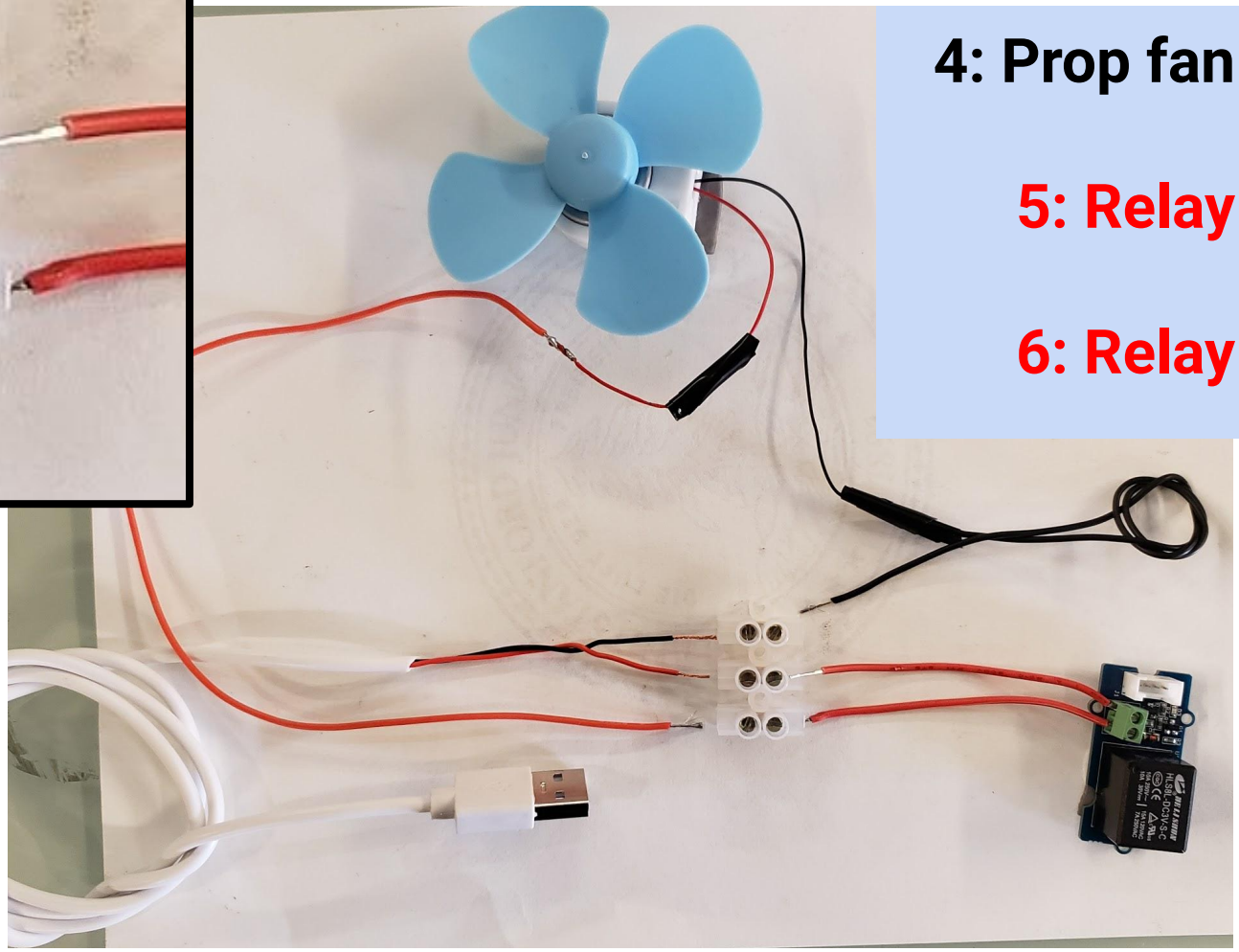
5: Relay

6: Relay

Table of Contents

Wiring Propeller Fan





4: Prop fan

5: Relay

6: Relay

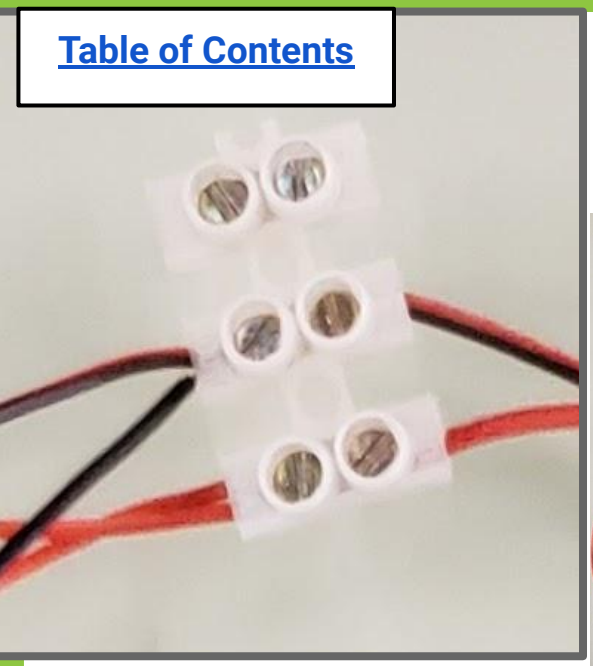
1: USB Power

2: USB Power

3: Prop fan

Black text = **black** wire; Red text = **red** wire

Table of Contents



Wiring Exhaust Fans

1: empty

2: BOTH Black wires from Exhaust fans

3: BOTH Red wires from Exhaust fans

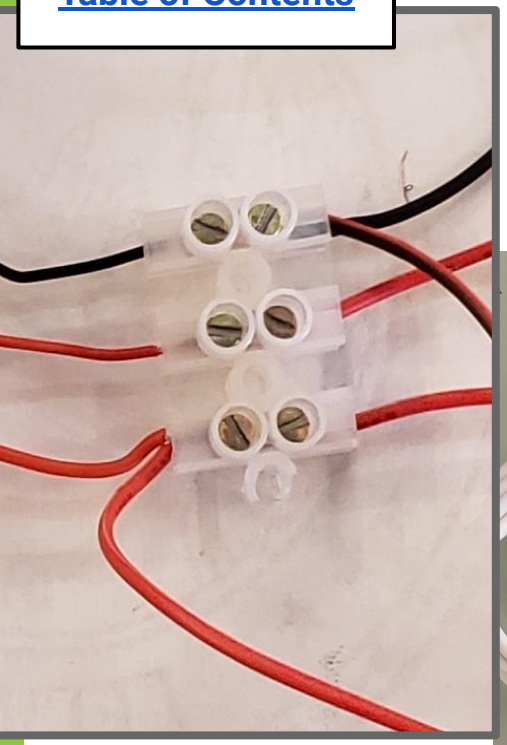
4: empty

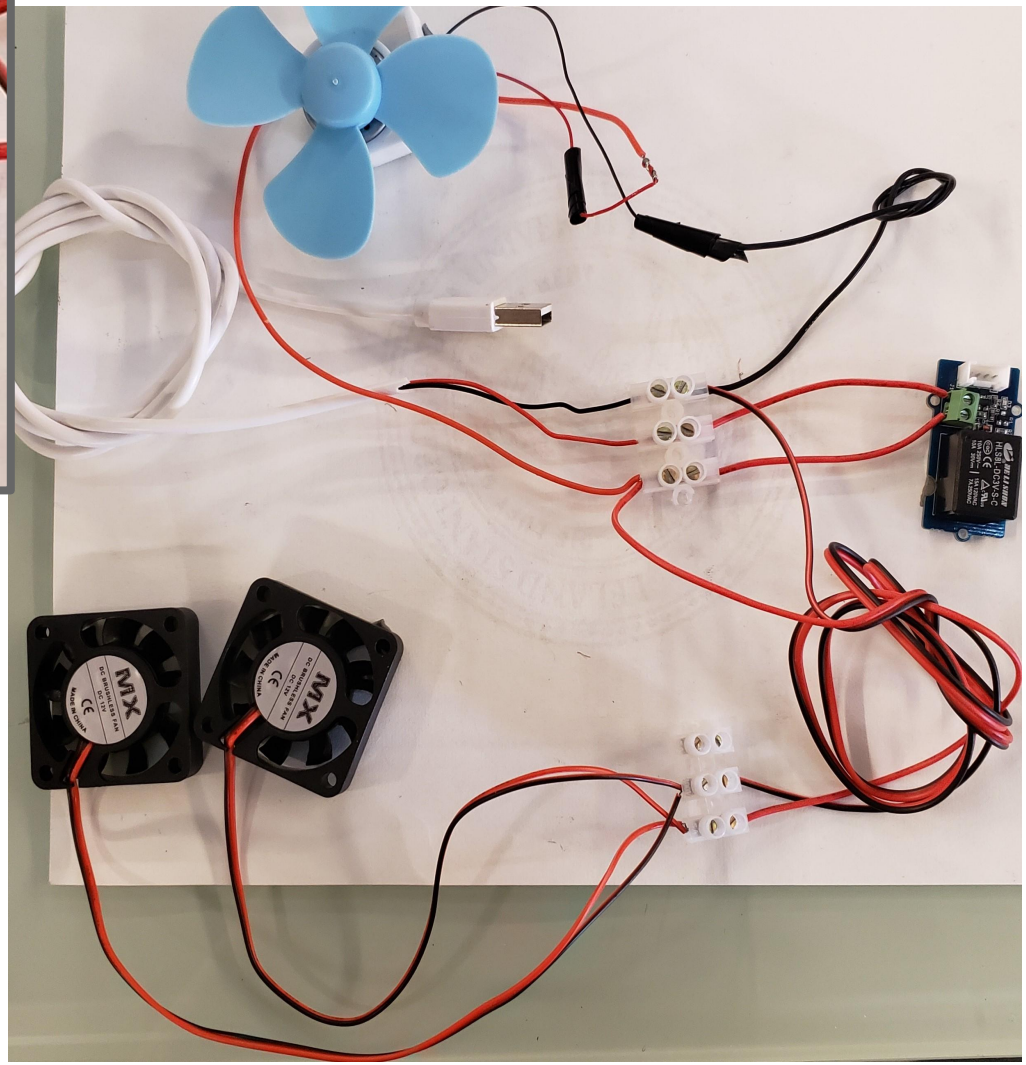
5: Black to get twisted with Black Prop fan

6: Red to get twisted with Red Prop fan

Table of Contents

Wiring Propeller Fan and Exhaust Fans





1: USB Power

2: USB Power

3: Prop fan

4: Prop fan

5: Relay

6: Relay

Table of Contents

Troubleshooting

Something not working? Try these things…

(in this order!)

1) Check your code for any errors (capitalization, indents, etc.)
2) Check to make sure everything is plugged in!
3) Unplug and replug in the MCU.
4) Check your wiring to make sure all wires are indeed connected and not crossing.
5) Press disconnect and run again.
6) Re-flash your MCU then try again.
7) Ask your partners for help!
8) Then finally… Ask your teacher. :)