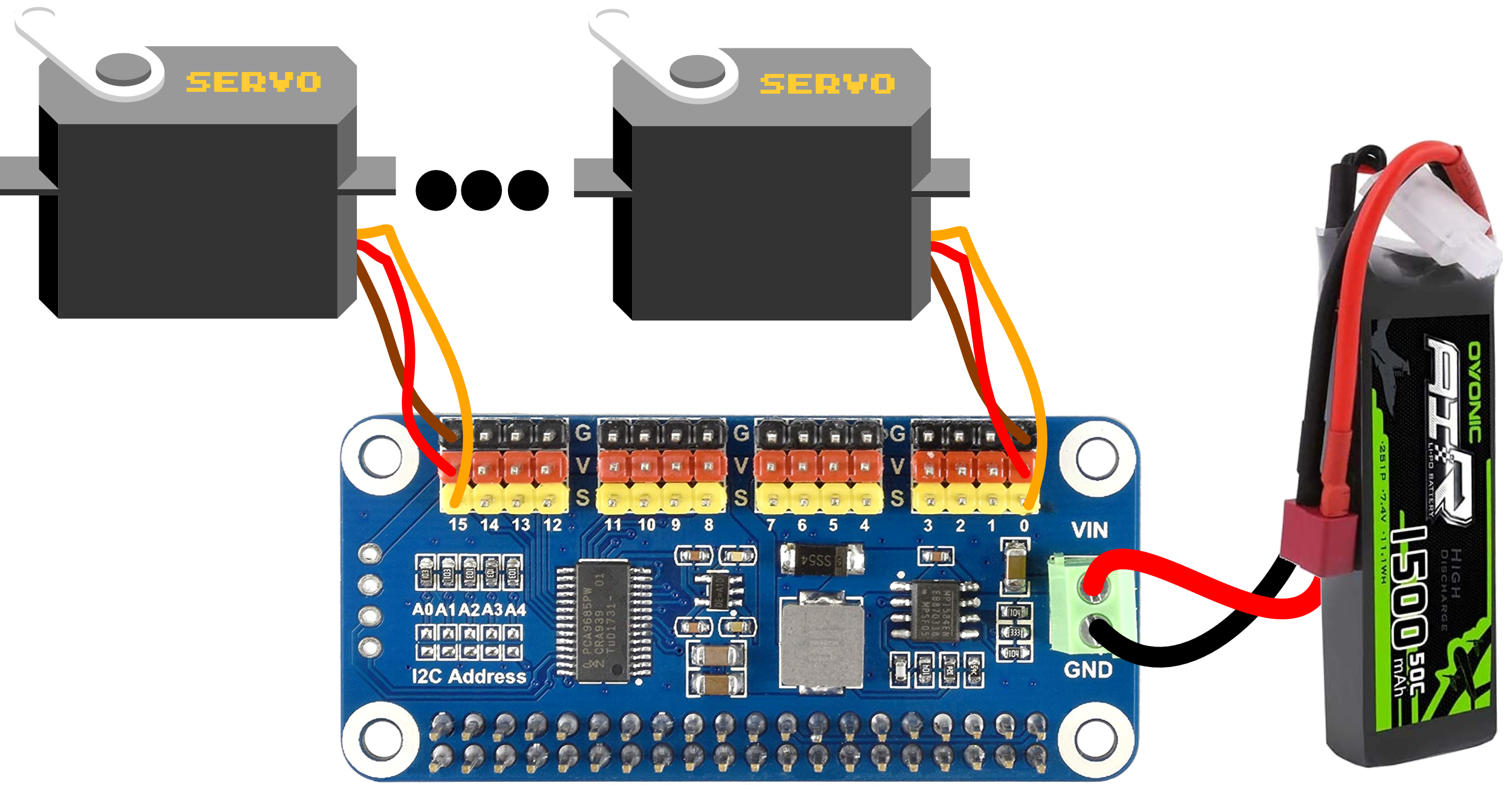
## Raspberry Pi Zero W

### Getting Started

1. Plug your SD Card into the Raspberry Pi
2. Connect your peripherals to the Raspberry Pi Zero W
   * Plug the power cable in last
3. Wait - This part may take a bit of time if it takes longer than 10 minutes. Disconnect the power cable and then reconnect
4. Follow the instructions in the dialog box, keep in mind if you plan on connecting to WiFi on Tufts Campus…
   * Connect to Tufts\_Wireless
   * Don’t Check for Updates (It won’t work as there is one more step to connect to the internet)
   * Use **ifconfig wlan0** to find the MAC Address of your Raspberry Pi and register it with Tufts Technology Services via the online Form
5. Configure the following interface(s):
   * I2C (For Controlling the Servo Hat), Serial (Optional), VNC (Optional)

### Using the [Waveshare Servo Hat](https://www.waveshare.com/w/upload/1/1b/Servo_Driver_HAT_User_Manual_EN.pdf)

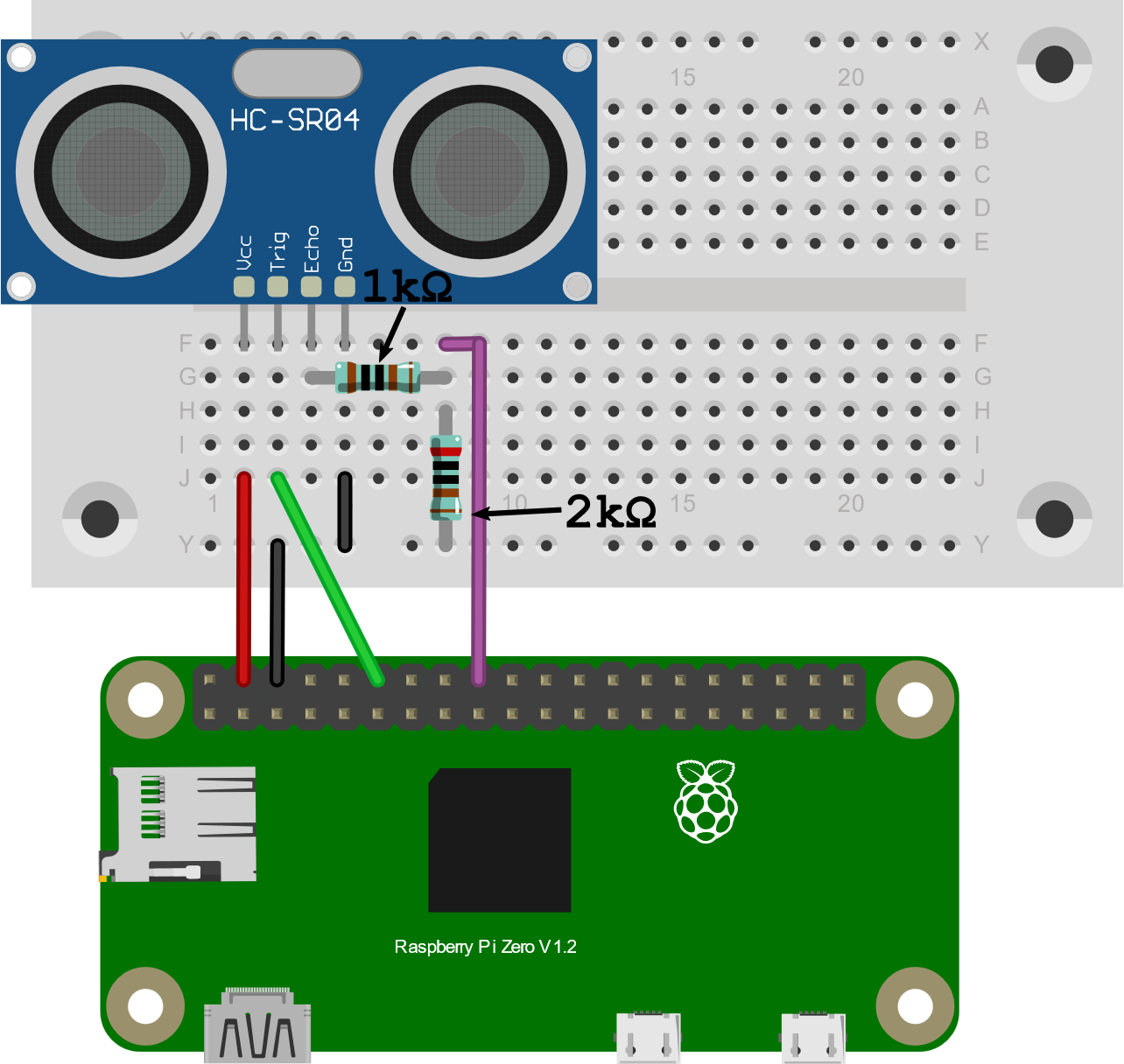
1. I2C should be enabled
2. Install the following libraries from the Terminal
   * **python-smbus, p7zip-full, RPi.GPIO**
3. Connect the hat to the top of the Raspberry Pi
4. Connect one or more servos to the hat, using this diagram.
5. The LiPo Battery Pack is only necessary if you want to operate without connecting the Raspberry Pi to power.
   * To make a connection between the LiPo and green terminal I soldered two jumper cables to the end of the connector.

Code -> http://www.waveshare.net/w/upload/6/6c/Servo\_Driver\_HAT.7z

### Getting Readings from the [Lidar Sensor](https://learn.adafruit.com/adafruit-vl53l0x-micro-lidar-distance-sensor-breakout/python-circuitpython)

1. pip install the **adafruit-circuitpython-vl53l0x** package
2. To get readings, import adafruit\_vl53l0x. This package has a ton of documentation online if you are looking for specific commands.
3. Refer to the diagram on the right for wiring it up.

### Getting Readings from the Ultrasonic Sensor HC-SR04

There is no specific package used to manage the Ultrasonic Sensor HC-SR04. Instead, you should wire it up as shown below and look for example code on the internet!

* VCC Connects to Pin 2 (5v)
* Trig Connects to Pin 7 (GPIO 4)
* Echo Connects to R1 (1k Ω)
* R2 (2k Ω) Connects from R1 to Ground
* Wire from R1 and R2 connects to Pin 11
* GND connects to Pin 6 (Ground)

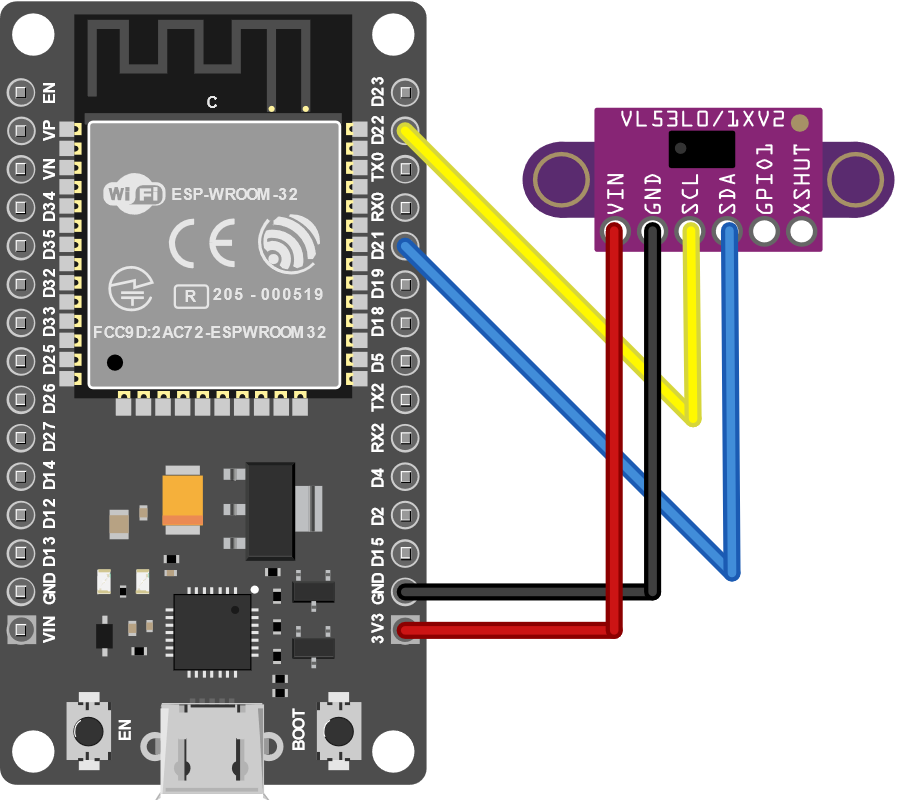
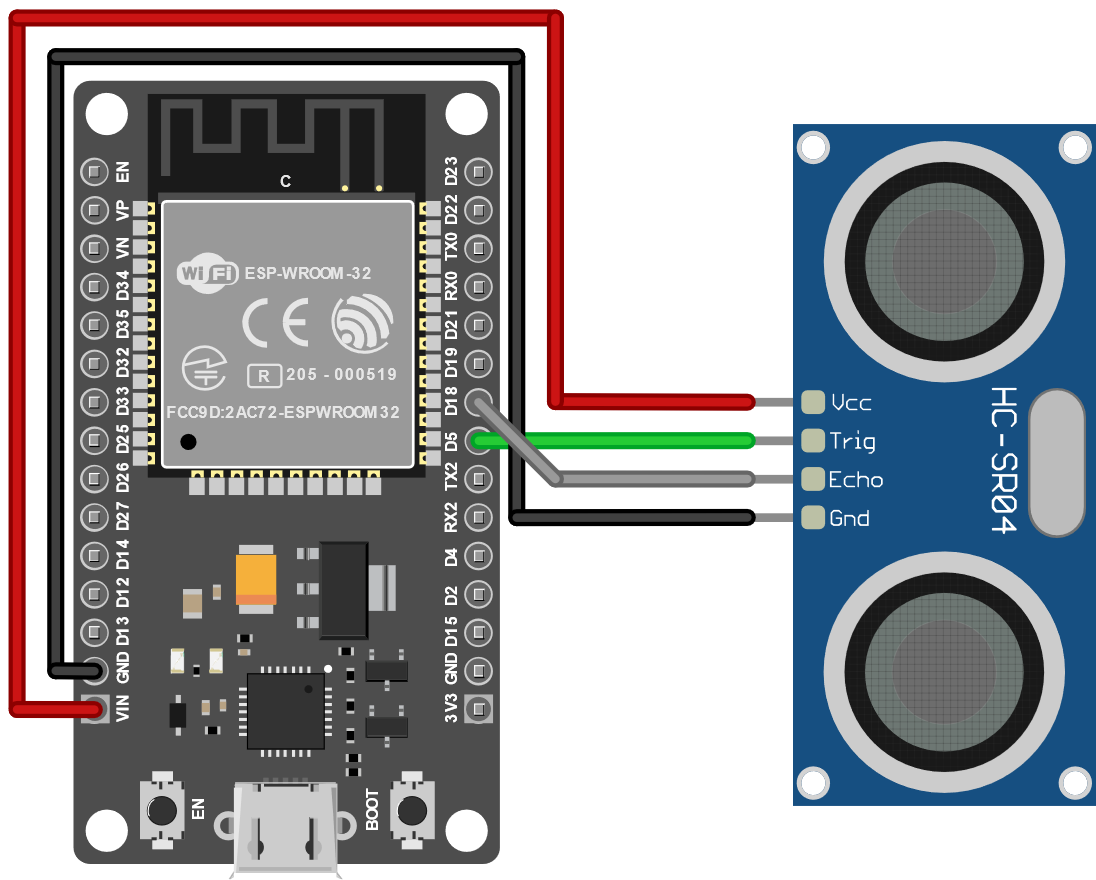
## ESP 32

### Getting Started

1. If you don’t have it already, download and install the [Arduino IDE](https://www.arduino.cc/en/software)
2. Add the ESP32 Board to the Arduino Board Manager by [URL](https://randomnerdtutorials.com/installing-the-esp32-board-in-arduino-ide-windows-instructions/). Use the Package Maintained by Espressif
3. Select the Board and Port in the **Tools > Board** menu (DOIT ESP32 DEVKIT V1)
   * If you don’t see the COM Port in your Arduino IDE, you need to install the CP210x USB to UART Bridge VCP Drivers)
4. If you are planning on using **Tufts\_Wireless** to connect this board to the internet, [find it’s MAC Address](https://randomnerdtutorials.com/get-change-esp32-esp8266-mac-address-arduino/) and register it with Tufts Technology Services.
5. There are a ton of great example codes to get you started programming in **File > Examples**

### Running the ESP32

1. Connecting to WiFi from
   * **File > Examples > WiFi (ESP32) > WiFiScan**
   * **WiFi.macAddress()** returns the MAC Address of the device
2. Using the [ESP32 Dual Cores](https://randomnerdtutorials.com/esp32-dual-core-arduino-ide/)
   * The ESP32 comes with 2 Xtensa 32-bit LX6 microprocessors, so it’s dual core.
   * You can use the **xTaskCreatePinnedToCore()** function to pin a specific task to a specific core
3. Using the Servo Motors
   * To run the [Servo Motors](https://dronebotworkshop.com/esp32-servo/) use ServoESP32 made by RoboticsBrno (Jaroslav Paral).
   * Find it in the Library Manager **Sketch > Include Libraries > Manage Libraries**
   * Refer to the wiring diagram (Bottom Center). The orange wire **Control** can connect to any digital IO pin.
4. Getting Readings from the [Distance Sensor](https://www.sparkfun.com/products/15569)
   * No library or demo code for this device, but examples can be found online!
   * Wiring Diagram (Bottom Right)
5. Getting Readings from the Lidar Sensor
   * To get readings from the Lidar Sensor use the [Adafruit\_VL53L0X Library](https://github.com/adafruit/Adafruit_VL53L0X)
   * To test if it is working open **File > Examples > Adafruit\_VL53L0X > vl53l0x** and upload to your Arduino wired up to the sensor
   * This sensor uses I2C communication. Refer to the Wiring Diagram (Bottom Left)

**

