Required Materials:

* Raspberry Pi 4 (recommended because it is 64 bit and has enough computational power to run vision)
* Micro SD card (>= 16GB)
* Pi Camera 2 (with the larger connector)
* Romi Chassis
* Battery Pack
* Battery Adapter
* Ethernet Cable (and the hardware to plug that ethernet cable into a computer)
* USB C cord and 5V power supply or computer port
* Mini HDMI to Full-size HDMI cord

Resources:

* Libcamera: <https://docs.arducam.com/Raspberry-Pi-Camera/Native-camera/Libcamera-User-Guide/>
* Github Repository: <https://github.com/r3dotstone/RomiVision>
* Raspberry Pi on Romi Tutorial: <https://www.pololu.com/blog/663/building-a-raspberry-pi-robot-with-the-romi-chassis>
* Romi Arduino Documentation: <https://www.pololu.com/docs/0J69>
* RPi camera and opencv: <https://pyimagesearch.com/2015/03/30/accessing-the-raspberry-pi-camera-with-opencv-and-python/>
* Opencv: <https://www.geeksforgeeks.org/opencv-python-tutorial/>, <https://learnopencv.com/getting-started-with-opencv/>
* NumPy: <https://www.w3schools.com/python/numpy/numpy_intro.asp>, <https://numpy.org/doc/stable/>
* Matplotlib: <https://www.w3schools.com/python/matplotlib_pyplot.asp>, <https://matplotlib.org/stable/users/index>

Step 1: Image the RPi

* Download RPi imager from this link: <https://www.raspberrypi.com/software/>
* Insert Micro SD card into computer

In the imager:

* Set device to Raspberry Pi 4 (or the model you have)
* Set operating system to “Raspberry Pi OS (64-bit)”
* Select SD card as storage
* Navigate to settings by clicking on the gear icon
* set hostname to something easy to type, **record it**
* set username to something easy to type, **record it**
* set password to something easy to type, **record it**
* Uncheck “Configure Network”
* Enable SSH **(very important)**
* Click Write
* Wait for a while
* Insert into RPi

Step 2: Get the RPi Wireless

* Plug RPi into power with nothing else plugged into it
* Let sit for plugged in for 5 minutes (configuring SD card)
* Unplug from power, then plug back in with nothing plugged in

There are many ways to interface with an RPi to get it initially connected to a network, and most of them are cumbersome. The simplest is to just plug it into a monitor, keyboard, and mouse, then connect it to the network you are on, and **record** its IP. Note that you will need to select \_\_\_

Others include:

* Configuring the network in the imager, but that will not not work with eduroam
* Connecting it directly to ethernet also won’t work on eduroam, because of the authentication required
* Place a wpa\_supplicant file in a specific directory on the micro SD card using this software: <https://www.diskinternals.com/linux-reader/>. Reference this article for connecting to eduroam: <https://www.diskinternals.com/linux-reader/>
* It is also possible to connect it to your computer with an ethernet cable, enable internet sharing, and VNC or SSH into it, this is the sleekest, but sometimes unreliable. To do this you will need TigerVNC (install from this link: <https://sourceforge.net/projects/tigervnc/files/stable/1.13.1/>)

In TigerVNC Viewer

* + Type “[hostname].local” in the VNC server field
  + Enter your username and password
  + Connect to the network the same way you would with a keyboard and mouse
* Once you have connected the RPi to a network, you can proceed to the following steps

In Command Prompt, filling in the info in [ ] with your information:

Note: lines preceded by “$” are commands

* Connect via SSH
  + $ ssh [username]@[hostname].local
  + Type “yes” and hit enter when prompted with the message about the ECDSA key fingerprint
  + Enter your password when prompted
  + $ sudo raspi-config
  + Navigate to the bottom and update the utility, reboot if necessary
  + Navigate to “Interface Settings” using the arrow-keys and enter
  + Navigate to VNC and enable the setting
  + Navigate to I2C and enable that as well
  + Make sure to select finish when you are done, and reboot if necessary

Step 3: Update and Install Packages, Clone the Git Repository

Note: When imaged, the RPi doesn’t come with updated packages. It is **very important** that all packages are updated. If something isn’t working, try updating the offending package.

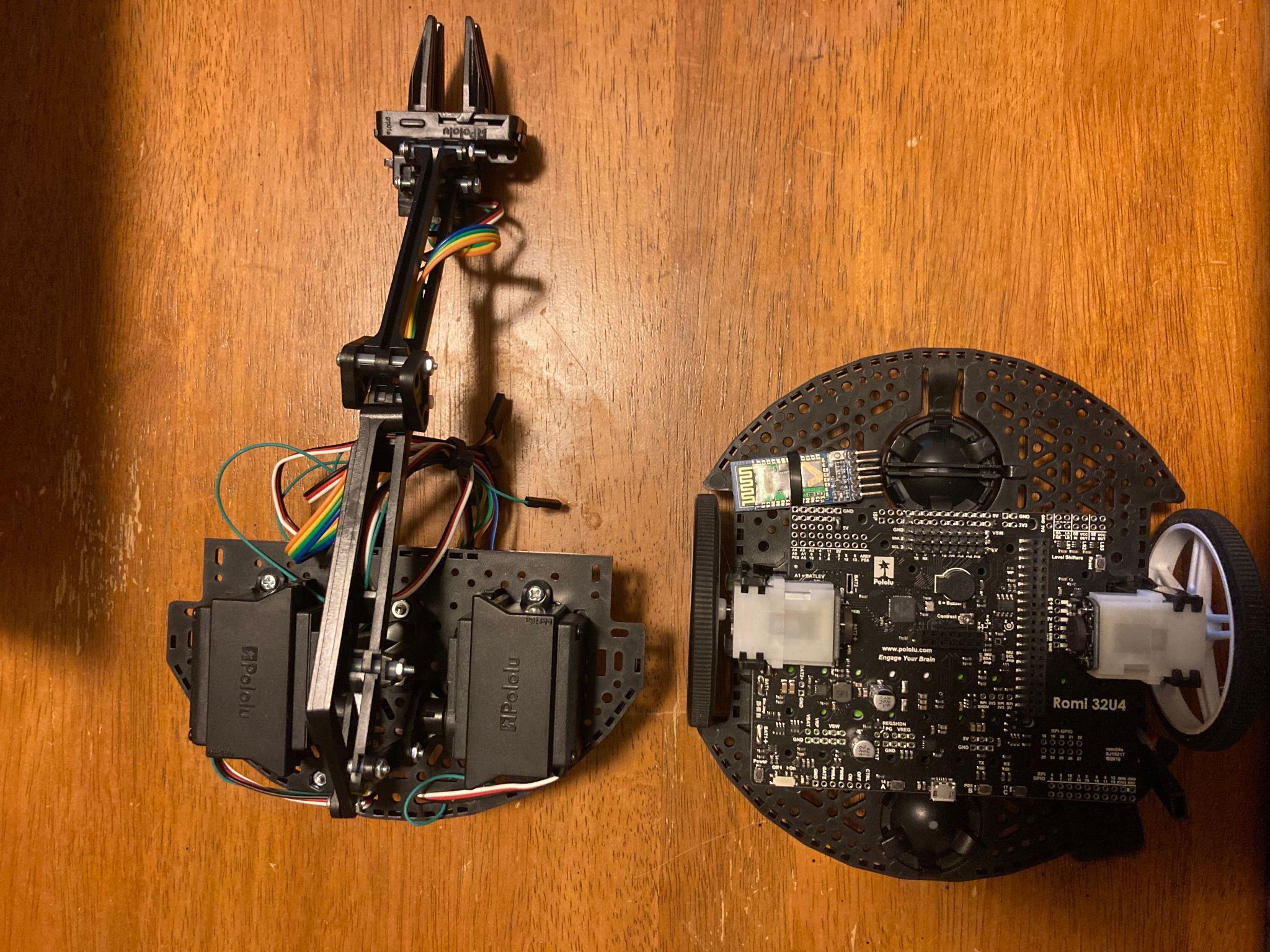
* $ sudo apt update
* $ sudo apt upgrade
* Do not unplug while updating packages
* $ pip install opencv-python --break-system-packages
* $ pip install picamera2 --break-system-packages
* $ git clone <https://github.com/r3dotstone/RomiVision.git>

Note: NumPy is installed by default under apt, but it is a dependency for almost everything. For visualization, you will want Matplotlib ($ pip install matplotlib --break-system-packages)

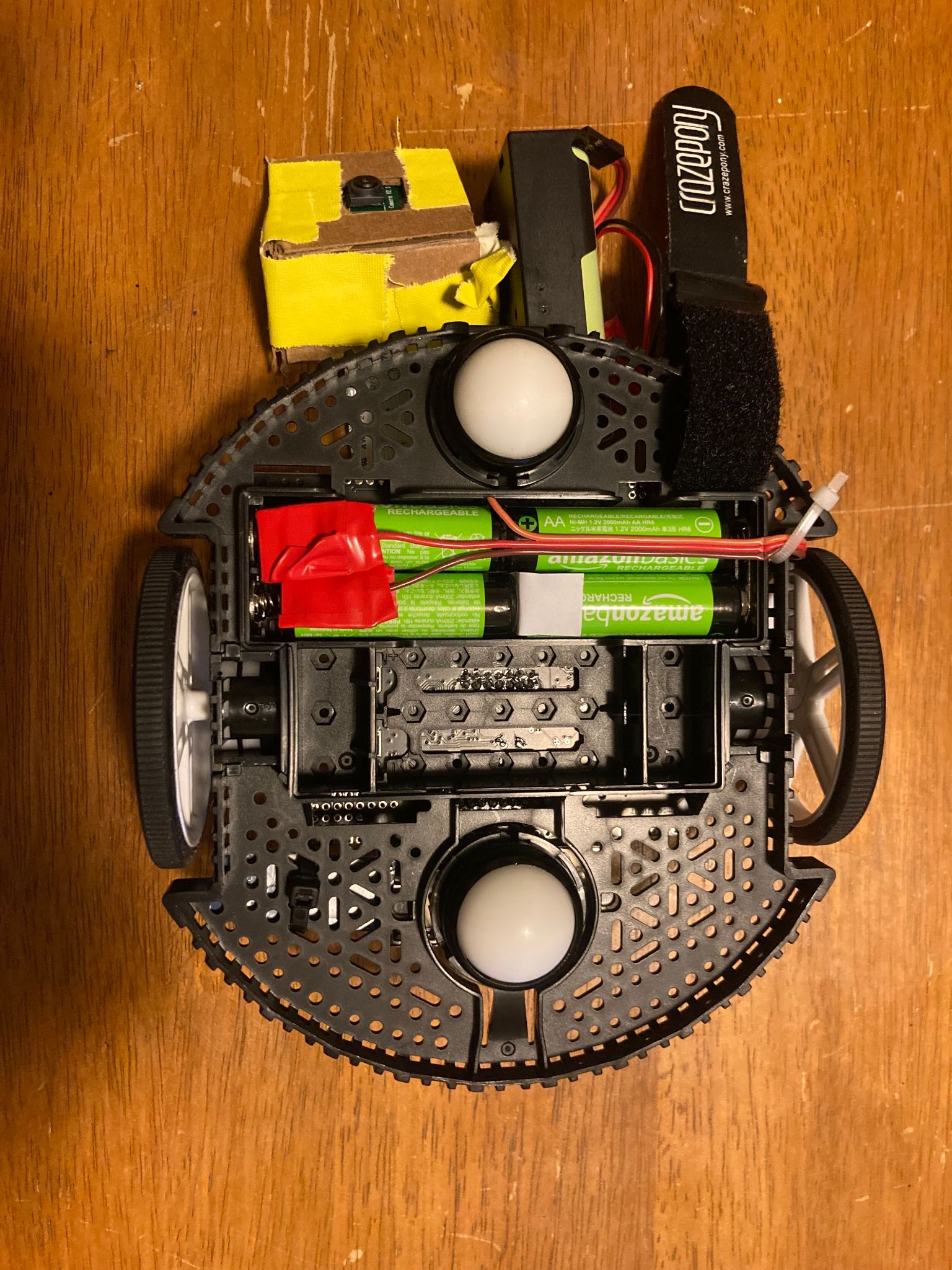
Step 4: Configure the Arduino (in the Romi)

* Download the Arduino IDE at this link: <https://docs.arduino.cc/software/ide-v1>
* Follow the directions at this link to add the board to the Arduino IDE: <https://www.pololu.com/docs/0J69/5.2>
* Go to <https://github.com/r3dotstone/RomiVision> and navigate to the arduino folder
* Download romi.ino and upload it to the Romi’s Arduino

Step 5: Assemble the Romi



* Remove the Romi’s arm, held on by four bolts on the bottom

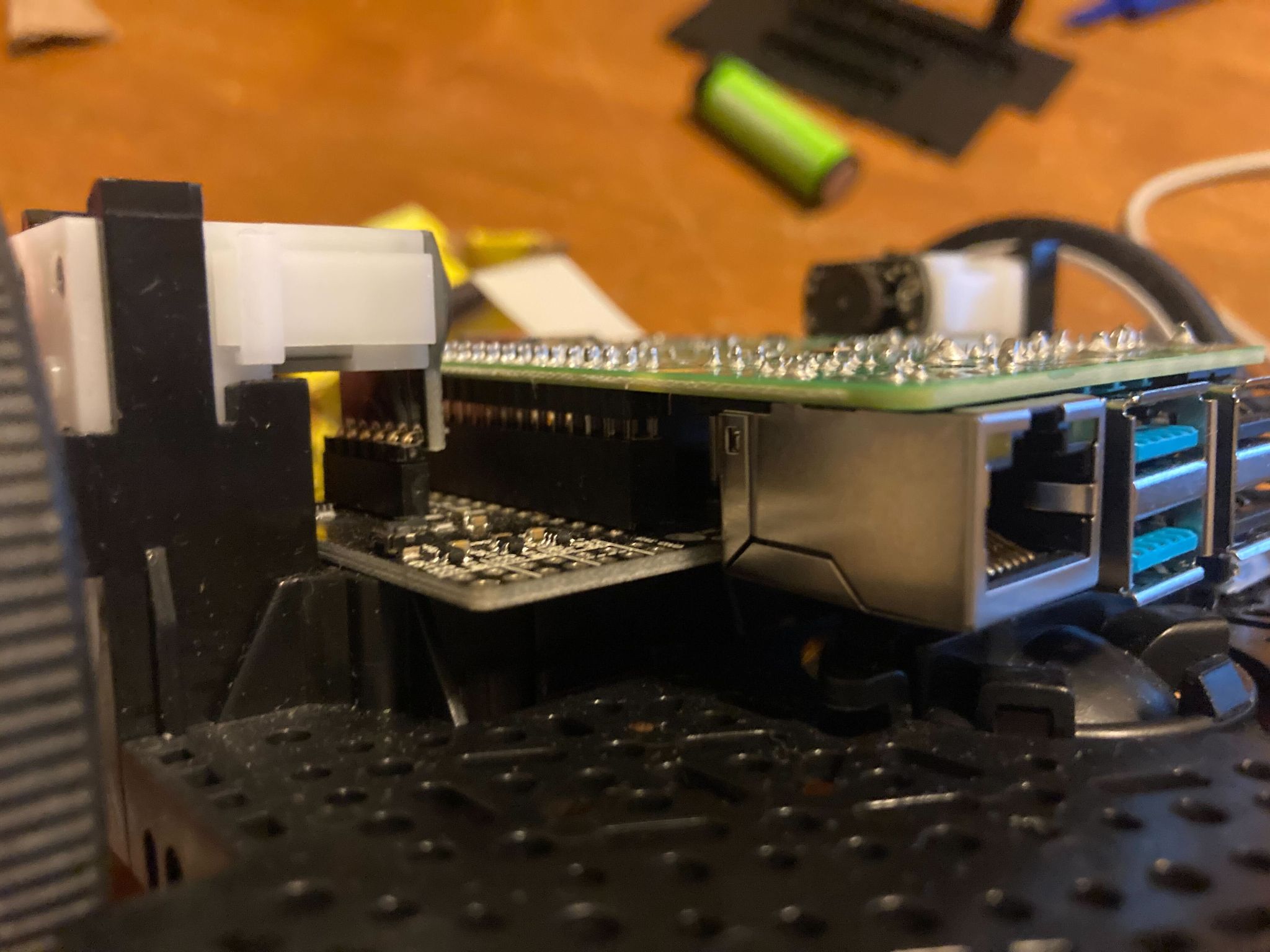


* Supply additional batteries to the Romi (without it, the RPi will overdraw the 4.8V of Ni M-H cells in the Romi to begin with)

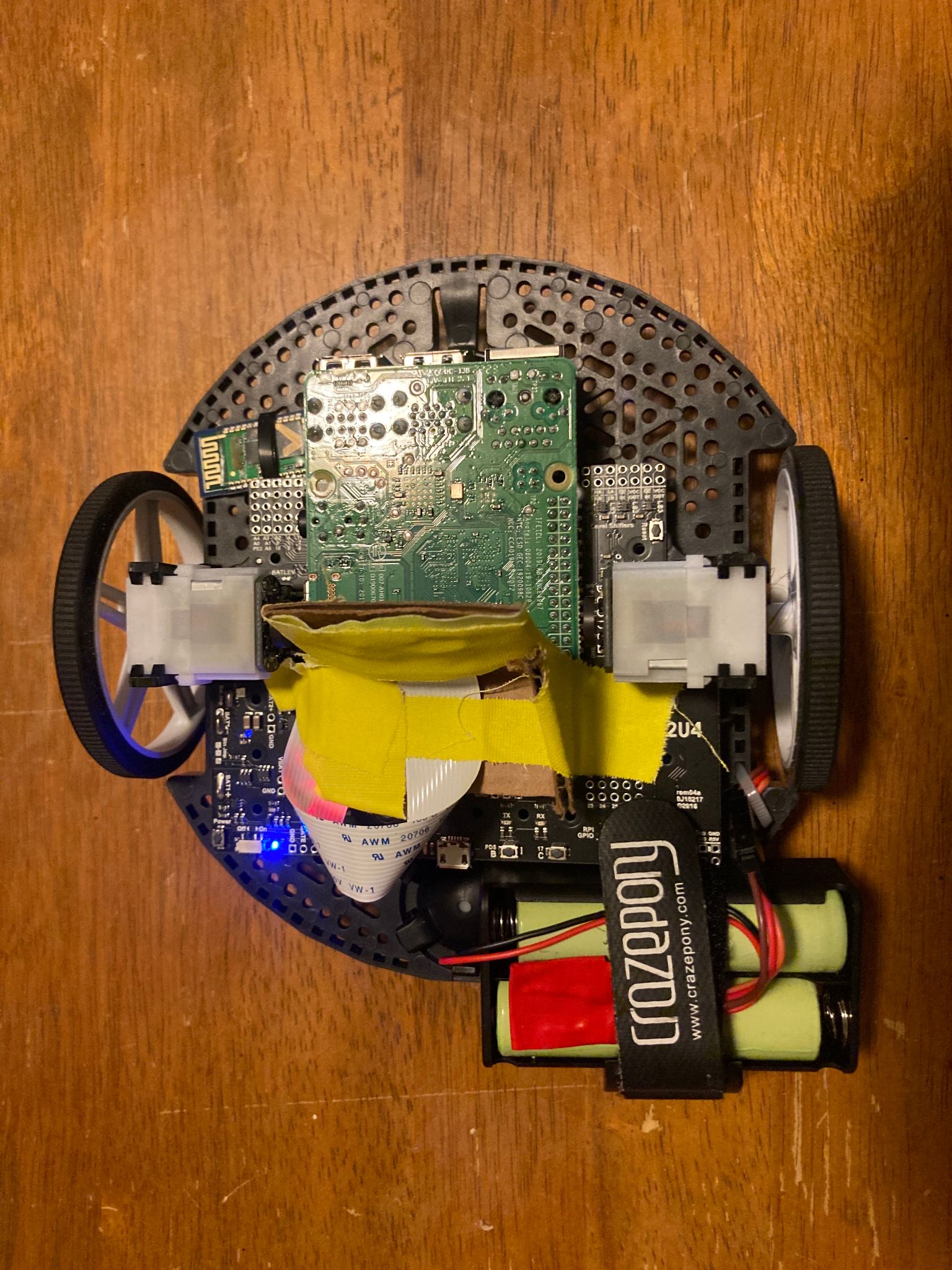
Note: The Romi’s control board can handle up to 10.8V. 2 3.7V 18650 LiIon cells in series worked well, but it is possible to install more Ni M-H or alkaline cells. Pictured are connectors that interface with the existing battery terminals and a piece of paper that breaks the original battery circuit.



* Plug the camera in to the RPi using the connector in the middle of the board (the other is for display)
* Lift the black plastic rail and insert ribbon cable, making sure to face the pins on cable towards the pins on the connector



* Plug the RPi into the Romi using the 2X20 pin connector on the Romi’s board that matches the GPIO pins on the RPi, reference [this](https://www.pololu.com/blog/663/building-a-raspberry-pi-robot-with-the-romi-chassis) guide for more details



* Mount the camera to the Romi directly between the wheels, ensuring not to crease the ribbon cable
* Secure the batteries and it’s ready to go

Step 6: Make and Run the Scripts

* To run any scripts, navigate to the folder with the script and run it with python
  + $ cd [folder]
  + $ python [scriptname].py
* All scripts will need to start with:

import cv2

import numpy as np

from picamera2 import Picamera2

import time

from romi\_interface.romi import Romi

romi = Romi()

cHeight = 1080

cWidth = 1920

picam2 = Picamera2()

config = picam2.create\_preview\_configuration(main={'size': (cWidth, cHeight), 'format': 'XRGB8888'})

picam2.configure(config)

picam2.start()

* If you need visualization, you will need:

import matplotlib.pyplot as plt