

I. Problem 1 (Prepare your laptop):

- A. Problem: None
- B. Learning: Other groups had connected through Command Prompt solely, learned how to do that, but still used RealVNC since a graphic interface is good.

II. Problem 2 (Setup/Login to the Pis):

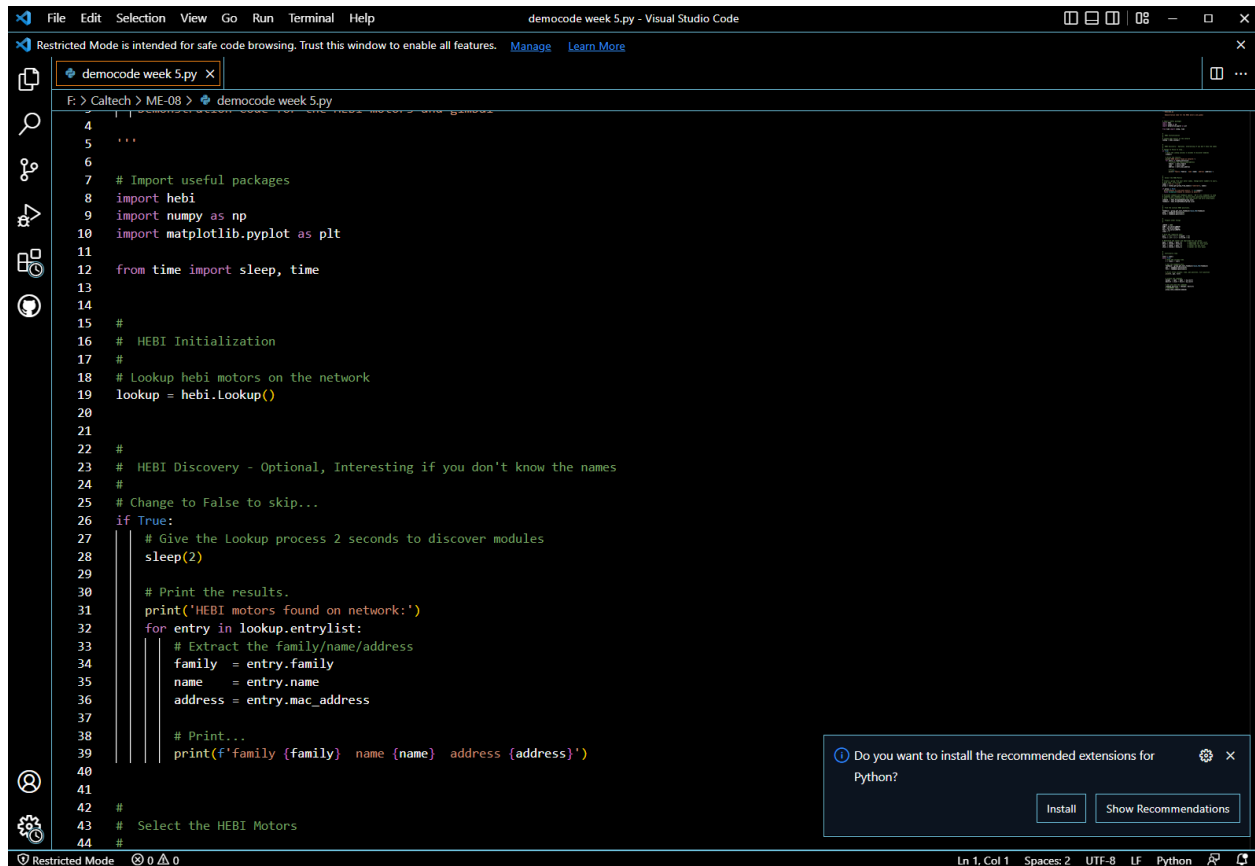
- A. Problem: None
- B. Learning: Sending files using RealVNC is enough to handle one file.

III. Problem 3 (Wire up the Gimbal):

- A. Problem: We were worrying about the wire entangled by the rotation of motors, we make it through the hole to the bottom motor and connect it that way. No entangling will occur even with a 360 rotation.
- B. Learning: It's also very simple, we just plug everything into the ports and turn on the switch and its lit and start working.

IV. Problem 4 (Transfer the demo code):

- A. Problem: Send the files using RealVNC and forgot to create a folder at first.
- B. Learning: scp command is a way I never used before, I usually use ftp to transfer files.



```
4
5
6
7 # Import useful packages
8 import hebi
9 import numpy as np
10 import matplotlib.pyplot as plt
11
12 from time import sleep, time
13
14
15 #
16 # HEBI Initialization
17 #
18 # Lookup hebi motors on the network
19 lookup = hebi.Lookup()
20
21
22 #
23 # HEBI Discovery - Optional, Interesting if you don't know the names
24 #
25 # Change to False to skip...
26 if True:
27     # Give the Lookup process 2 seconds to discover modules
28     sleep(2)
29
30     # Print the results.
31     print('HEBI motors found on network:')
32     for entry in lookup.entrylist():
33         # Extract the family/name/address
34         family = entry.family
35         name = entry.name
36         address = entry.mac_address
37
38         # Print...
39         print(f'family {family} name {name} address {address}')
40
41
42 #
43 # Select the HEBI Motors
44
```

## V. Problem 5 (Have the Code discover the HEBI motors):

- A. Problem: Initial run, didn't find motor, changed numbers of motor of the according pan tilt order, ran again and connected.
- B. Learning: No hotkey on pi for saving ( ctrl+s )

```
#
# Select the HEBI Motors
#
# Create a group from your motor names. Change motor numbers to yours,
# with [pan, tilt] order!
names = ['4.5', '7.4']
group = lookup.get_group_from_names(['robotlab'], names)
```

## VI. Problem 6 (Read the Motor Positions):

- A. Problem: When we manually move the joint and do not move it back, it will have a offset when running, and the motor will be stalled at the position, have to shut off and turn back up the motor to retain the despan and destilt.
- B. Learning: Fragile hardware

```
# Print three columns: time, pan position, tilt position
print('time: ' + t, 'pan: ' + pan, 'tilt: ' + tilt)
```

## VII. Problem 7 (Move the Motors!):

- A. Problem: As stated above, when manually moved the motor while it is on, it will create a physical offset. Our motor is noisy, but does not seems like a problem
- B. Learning: Sending code to the motor is different than the simulator

```
10
11 # Set and send the commands
12 command.position = [despan, destilt]
13 # UNCOMMENT THIS:
14 group.send_command(command)
15
```

## VIII. Problem 8 (This Week's Check-Off List):

- A. Problem: Motor is a bit noisy, but the TA say it's not a problem, so it's fine.
- B. Learning: Code ran, checked off.