How to download Raspberry pi’s Operating Systems

Basic setup

- Plugin keyboard to the USB port (preferably USB 2.0 port to reserve USB 3.0 of higher bandwidth to other better-demanded devices such as hard drive)
- Plugin micro HDMI to the port 0 (the one closer to the power port)
- Optional: plugin the network cable to Ethernet port (You can use a wireless connection but Ethernet is typically faster)

Software setup

- Insert your microSD card to your computer. If it’s used, first format it before further actions. If it’s new, ignore it.
- Downloads DOOBS (an operating system for Raspberry pi). If you bought a microSD card with preinstalled DOOBS, ignore this step
- Unzip the downloaded file and copy all files into the microSD card. This will transplant the while NOOBS OS to microSD card
- Take out the microSD card from your computer and plug it in Raspberry pi
- Plug in the power cable to run your pi

Config setup

- Once pi is powered on, you will see a window on the screen with several choices of OS (only in this case when you use Ethernet). Click Wifi network to connect to the wireless if you don’t use Ethernet.
- Choose Raspbian Full as recommended and click install
- Once the installation is completed, the Pi will reboot and you will see a “welcome to raspberry pi” window
- Follow instruction on that window and do the config setting
- When a “update software” window pops, click next. This might take a while
- Click restart

How to open bluetooth

- After you have successfully installed the Raspbian operating system on each raspberry pi, open a terminal, and type “service bluetooth status”
- If you see a green “active (running)”, then it means the bluetooth service is up and running. If not, run the command “service bluetooth start” to start the bluetooth service
- Now run “bluetoothctl” and you will enter the bluetooth control line
- Run “power on”
- Run “agent on”
- Run “discoverable on”
- Run “pairable on”
- These series of commands make sure that the pi is discoverable and pairable

How to remote login you Pi from your own computer

- Click raspberry icon on the top left corner of the screen → select “Preferences” → select “Interfaces ” → enable SSH
- Open your terminal (mac) or command prompt (windows) of your personal computer and type ssh pi@ip_address, where ip_address is the ip address of your Pi, which you can get by typing ifconfig on your pi’s command prompt

How to download codes into pis

- Once you have the code ready on your own computer, simply securely copy it into your pi by typing scp code_path pi@ip_address:targeted_path_on_your_pi

Meaning of user interface input

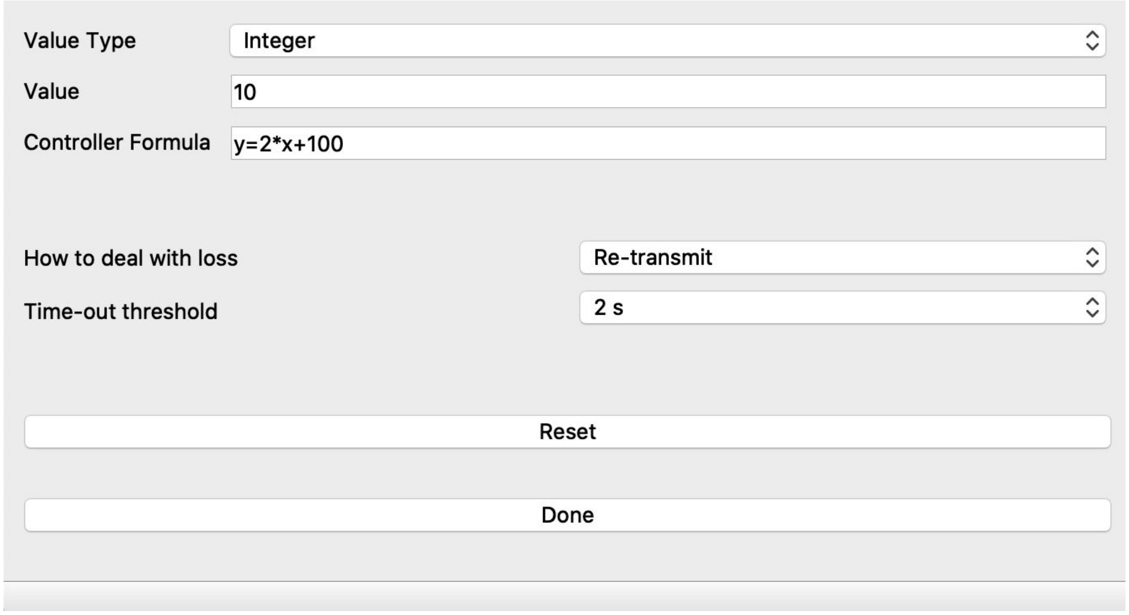
- Controller
    - Enter the mac address of controller
    - Controller is the pi that controls the behavior of the plant
- Plant
    - Enter the mac address of plant
    - Plant is the pi that operates and sends real-time information to controller
- Intermediate Node
    - Enter the mac address of intermediate node
    - Intermediate node is the pi that serves as the middleman between controller and plant
- Value Type
    - The types of value that you want to transmit
    - E.g. matrix, integer, string, etc
- Value
    - The initial value for your system
    - Depends on your value type
- Controller Formula
    - The state transition formula for control
    - Controller receives data from plant and after computing, it will send new data back to the plant
- How to deal with loss
    - If packet loss happens during data transmission, how to deal with it
    - E.g. retransmission, switch transmission routes, controller guessing
- Time-out threshold
    - To detect loss, we implement a certain timeout threshold

How to connect two pi’s via bluetooth

- After you have finished the input in the user interface, it will generate two argument files that the controller pi and the plant pi needs. Scp the controllerArgs file to the controller pi, and the plantArgs file to the plant pi
- Scp the respective python files to the respective pi's
- Be sure to keep the respective python files in the same directory with their arguments. For the intermediate hop pi, there's no argument file so the python file can be put anywhere
- In each pi's terminal, run the python files. Run them in order of hop -> plant -> controller. If you don't run them in this order, it may cause communication errors

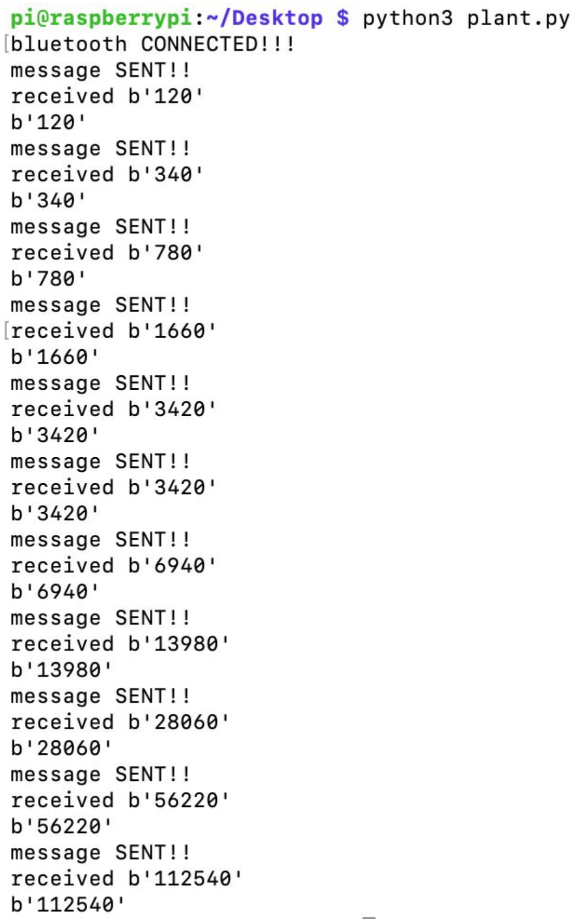
Manual test

- Try entering these parameters into the user interface in the fields other than the mac addresses:



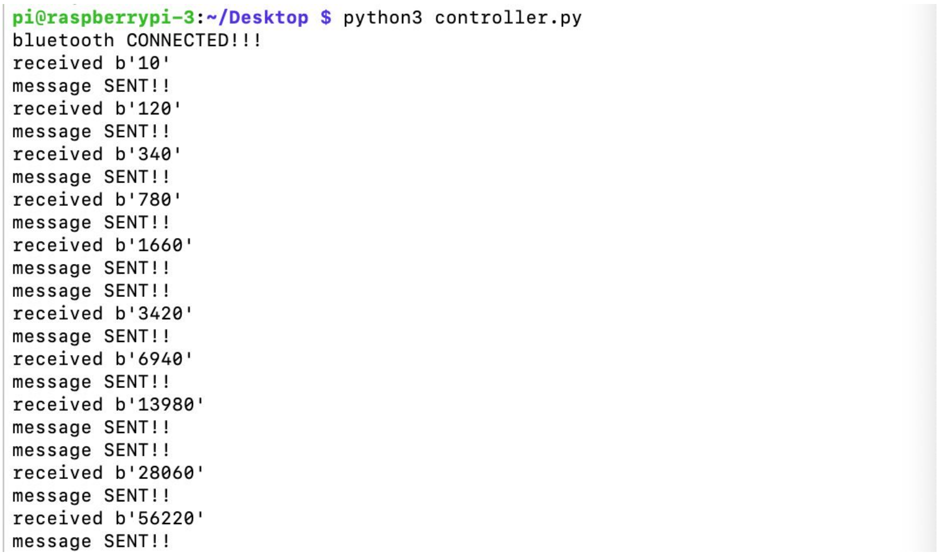
Value Type: Integer
Value: 10
Controller Formula: y=2*x+100
How to deal with loss: Re-transmit
Time-out threshold: 2 s

- After hitting done and copying all the relevant files into the respective raspberry pi's, run the python programs in the order explained above. You should get the following results:
    - Plant:



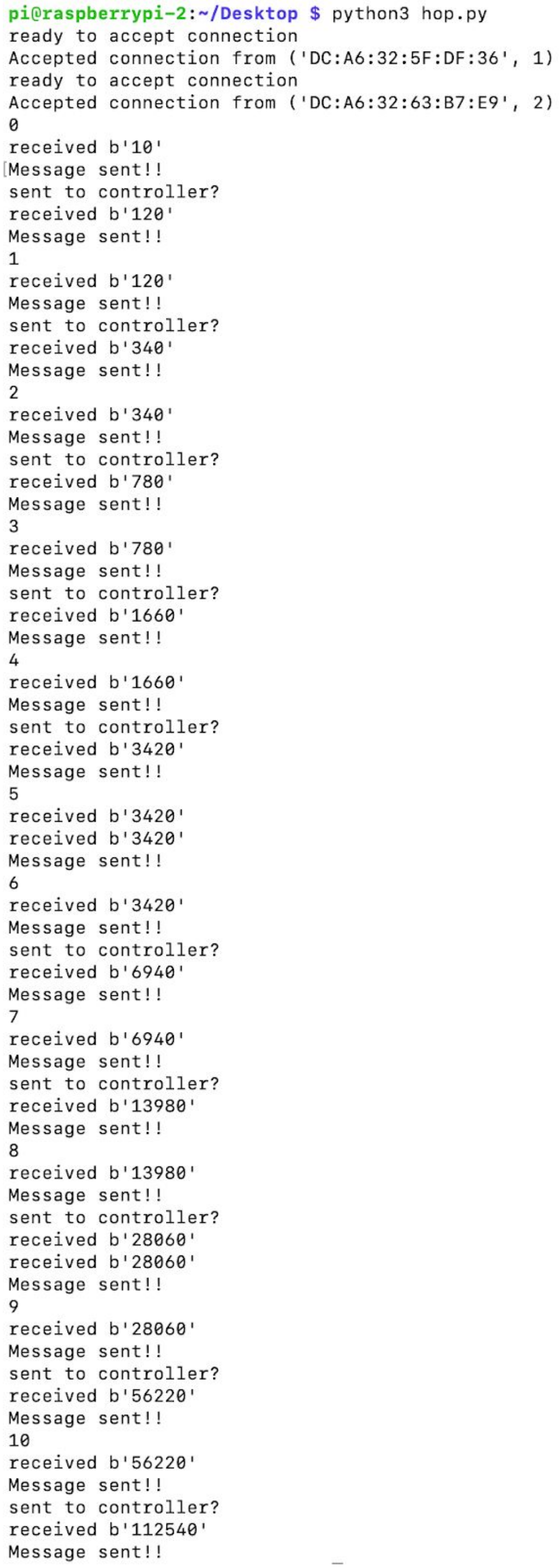
```
pi@raspberrypi:~/Desktop $ python3 plant.py
bluetooth CONNECTED!!!
message SENT!!
received b'120'
b'120'
message SENT!!
received b'340'
b'340'
message SENT!!
received b'780'
b'780'
message SENT!!
received b'1660'
b'1660'
message SENT!!
received b'3420'
b'3420'
message SENT!!
received b'3420'
b'3420'
message SENT!!
received b'6940'
b'6940'
message SENT!!
received b'13980'
b'13980'
message SENT!!
received b'28060'
b'28060'
message SENT!!
received b'56220'
b'56220'
message SENT!!
received b'112540'
b'112540'
```

- Controller:



```
pi@raspberrypi-3:~/Desktop $ python3 controller.py
bluetooth CONNECTED!!!
received b'10'
message SENT!!
received b'120'
message SENT!!
received b'340'
message SENT!!
received b'780'
message SENT!!
received b'1660'
message SENT!!
message SENT!!
received b'3420'
message SENT!!
received b'6940'
message SENT!!
received b'13980'
message SENT!!
message SENT!!
received b'28060'
message SENT!!
received b'56220'
message SENT!!
```

- Hop:



```
pi@raspberrypi-2:~/Desktop $ python3 hop.py
ready to accept connection
Accepted connection from ('DC:A6:32:5F:DF:36', 1)
ready to accept connection
Accepted connection from ('DC:A6:32:63:B7:E9', 2)
0
received b'10'
Message sent!!
sent to controller?
received b'120'
Message sent!!
1
received b'120'
Message sent!!
sent to controller?
received b'340'
Message sent!!
2
received b'340'
Message sent!!
sent to controller?
received b'780'
Message sent!!
3
received b'780'
Message sent!!
sent to controller?
received b'1660'
Message sent!!
4
received b'1660'
Message sent!!
sent to controller?
received b'3420'
Message sent!!
5
received b'3420'
received b'3420'
Message sent!!
6
received b'3420'
Message sent!!
sent to controller?
received b'6940'
Message sent!!
7
received b'6940'
Message sent!!
sent to controller?
received b'13980'
Message sent!!
8
received b'13980'
Message sent!!
sent to controller?
received b'28060'
received b'28060'
Message sent!!
9
received b'28060'
Message sent!!
sent to controller?
received b'56220'
Message sent!!
10
received b'56220'
Message sent!!
sent to controller?
received b'112540'
Message sent!!
```

- If you don’t see the exact same output on the respective pi’s, check the steps for errors and try to redo the process.