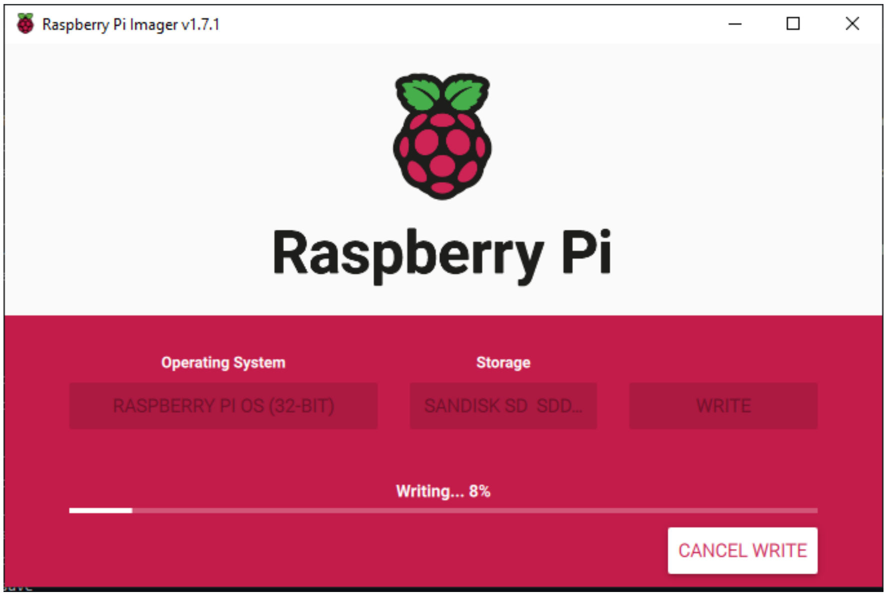
1.) Use the raspberry-pi imager to load full desktop version (erase the card first if necessary)



Operating System: RASPBERRY PI OS (32-BIT)

Storage: SANDISK SD SDD...

Writing... 8%

2.) plug in raspberry-pi to monitor, power, mouse, keyboard (try to avoid headless setup as it is easier to make a mistake during)

NOTE: Ethernet is also helpful for this if wifi networks are uncooperative

3.) Follow the raspberry-pi setup with the username of ece02

4.) Go into the apps menu (top left raspberry), go to preferences->raspberry pi configuration->display->set headless resolution to 1920x1080 or your preferred resolution (this helps make sure the machine boots even with any cable issues)

5.) preferences->raspberry pi configuration->interfaces set SSH and VNC to enable (only necessary if you intend to remote access)

6.) allow the system to restart -> type hostname -I to find the local IP (make sure the wifi is on and you get the wifi ip) ->SSH into from another computer to confirm access (do the same with VNC if you want access to the GUI, VNC can be installed on windows).

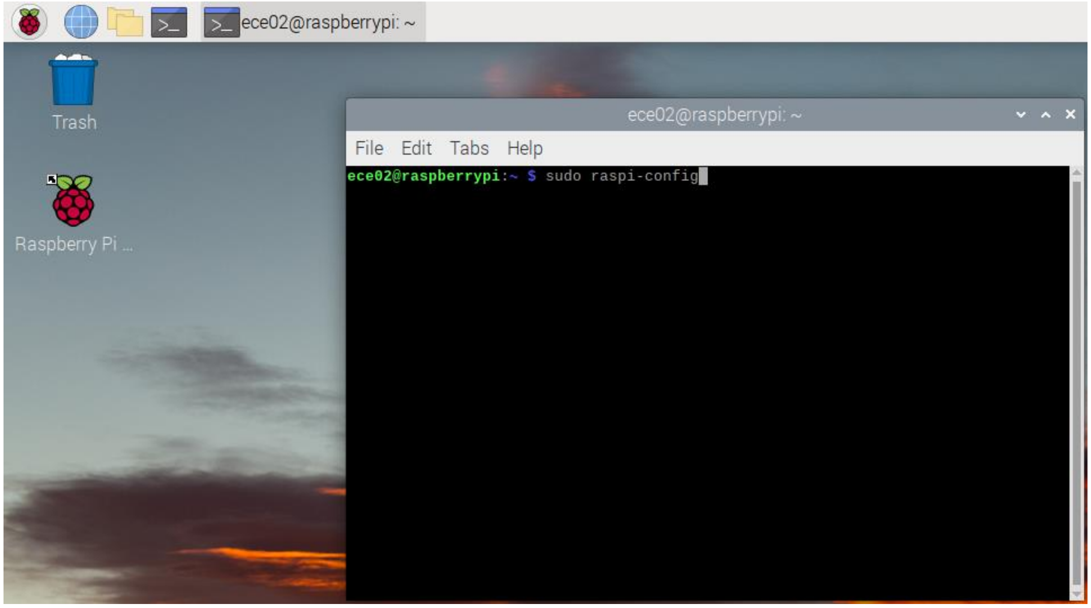
7.) reboot with ethernet disconnected to verify headless over wifi if ethernet is not possible at the location the model is to be stored

SETUP CAMERA

Be aware, this method will be deprecated in the future and a solution using libcamera should be found or written

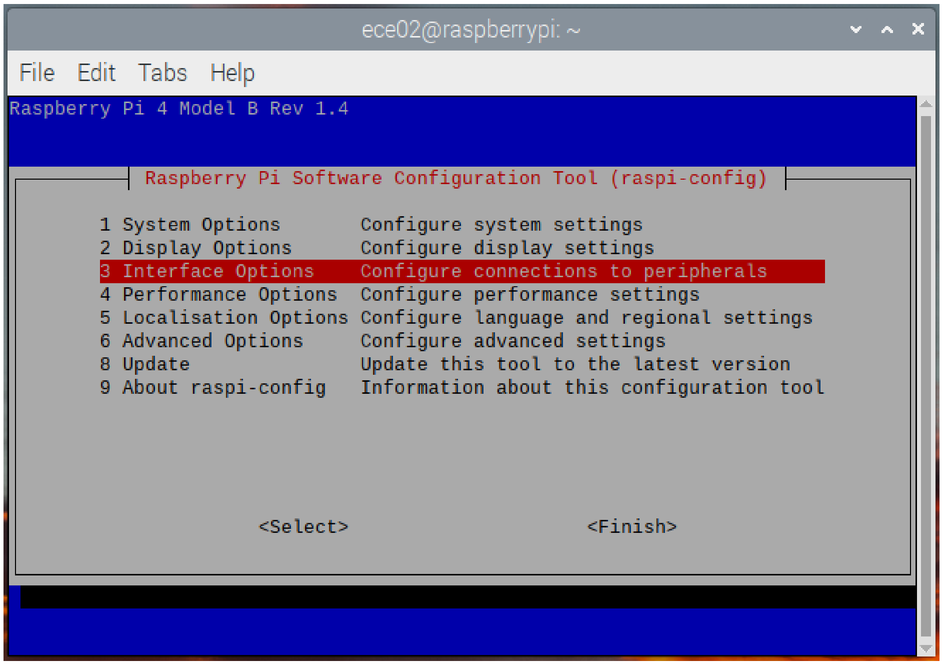
Also, this method disables VNC (when legacy camera is enabled)

1.) open a terminal and enter 'sudo raspi-config' (or ssh in if accessing remotely)



```
ece02@raspberrypi:~ $ sudo raspi-config
```

2.) Interface options -> legacy camera -> enable -> finish -> reboot

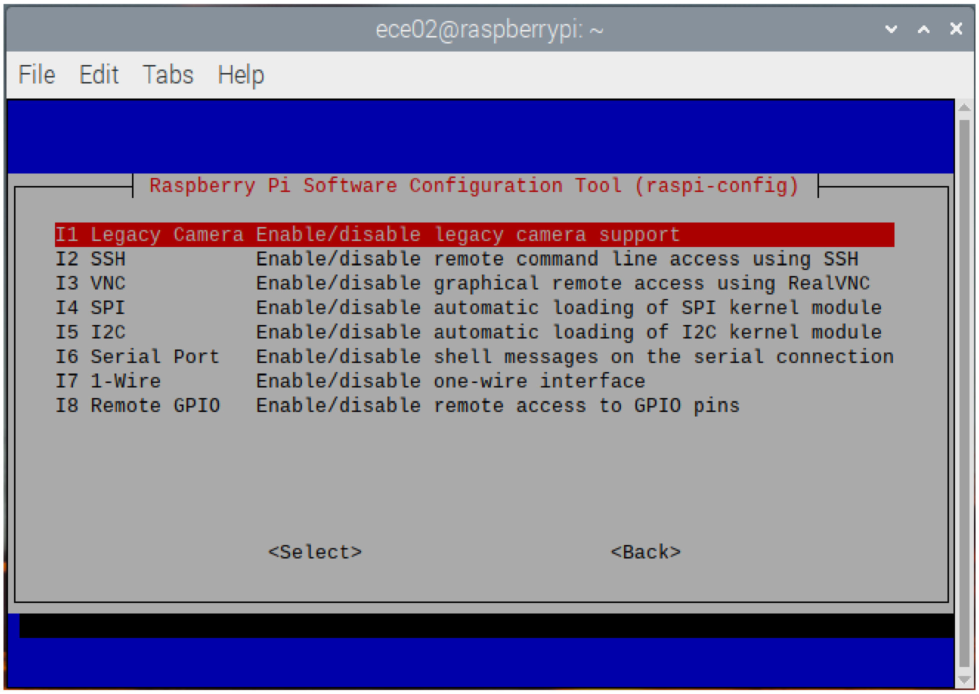


```
Raspberry Pi 4 Model B Rev 1.4

         Raspberry Pi Software Configuration Tool (raspi-config)

    1 System Options       Configure system settings
    2 Display Options      Configure display settings
    3 Interface Options    Configure connections to peripherals
    4 Performance Options  Configure performance settings
    5 Localisation Options Configure language and regional settings
    6 Advanced Options     Configure advanced settings
    8 Update               Update this tool to the latest version
    9 About raspi-config   Information about this configuration tool

              <Select>                      <Finish>
```



```
Raspberry Pi Software Configuration Tool (raspi-config)

I1 Legacy Camera Enable/disable legacy camera support
I2 SSH           Enable/disable remote command line access using SSH
I3 VNC           Enable/disable graphical remote access using RealVNC
I4 SPI           Enable/disable automatic loading of SPI kernel module
I5 I2C           Enable/disable automatic loading of I2C kernel module
I6 Serial Port   Enable/disable shell messages on the serial connection
I7 1-Wire        Enable/disable one-wire interface
I8 Remote GPIO   Enable/disable remote access to GPIO pins

                 <Select>                        <Back>
```

NOTE: This will give a deprecation warning. For the time being, the method remains intact. Furthermore the camera is not neccessary for in-person use.

3.) open a terminal and enter 'nano rpi_camera_surveillance_system.py'

4.) copy/paste the code from this link "https://randomnerdtutorials.com/video-streaming-with-raspberry-pi-camera/" (alternatively a copy is provided in the repository which node red is set to use by default)

Note: the code example is from the picamera library so is provided with the Raspbian OS.

5.) ctrl-x -> y -> enter   to save

6.) python3 rpi_camera_surveillance_system.py

7.) test by entering the following IP address into browser (in the case of localhosting the ip is 127.0.0.1) http://{raspberryPi'sIpGoesHere}:8000

8.) Complete any necessary port forwarding on your router if you want the simulator to be accessible across the internet

Setup NODERED

1.) https://nodered.org/docs/getting-started/raspberrypi follow the instructions here

2.) run 'node-red init' (you'll need the settings file for later) (~/.node-red)

NOTE: also enable autostart on boot

3.) security (This is only necessary if the automatic setup does not work)

a.) ‘node-red admin hash-pw "typePasswordHere"’

b.) copy and paste the hash into the password section specified in the tutorial in the settings file

c.) save

d.) reboot

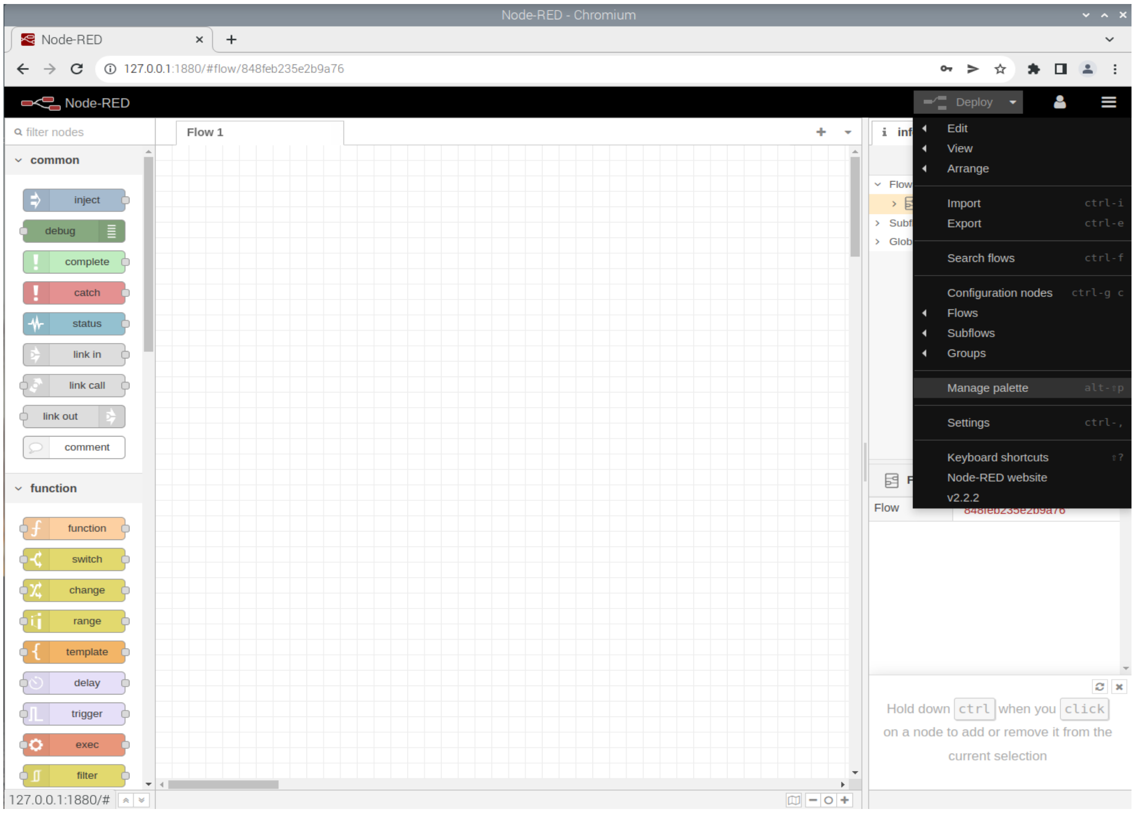
Steps 4 and 5 are only necessary for across internet access

4.) make sure to update port forwarding if across internet use is desired

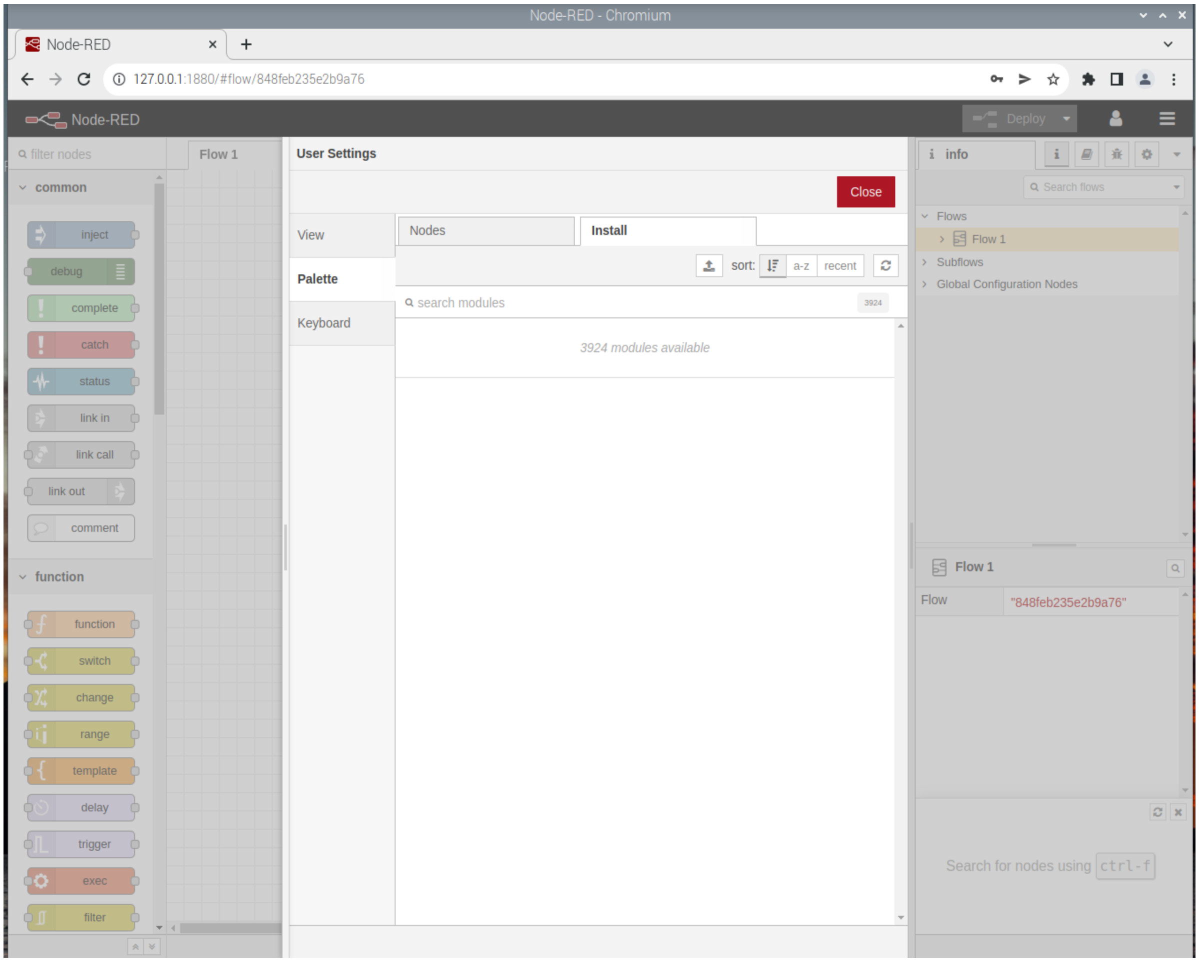
5.) ‘node-red-log’ to check port, then ‘ifconfig’ for ip

6.) add the following item to the pallet via "triple-bar-menu-in-top-right"->manage pallet

-node-red-dashboard



Hold down ctrl when you click on a node to add or remove it from the current selection



7.) pip3 install the following modules:

pyXSteam

scipy

pip3 install --upgrade numpy

and the following apt installs:

sudo apt-get install libatlas-base-dev

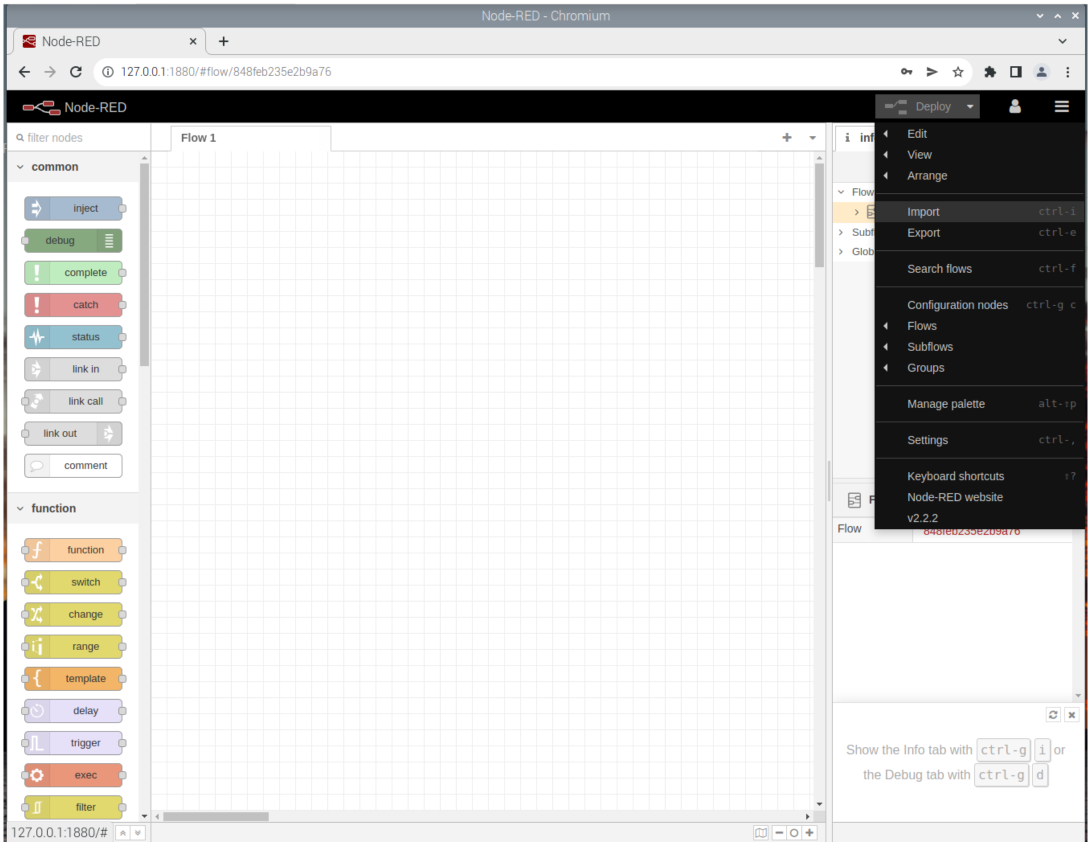
8.) REBOOT

9.) Import the nodered flows on your pc from the github (either the test bench or the simulator)

a.) ‘git clone https://github.com/TnMarshall/ECE02_2022_Project_Documentation.git’

b.) in a browser go to 127.0.0.1:1880 (this is the node-red editor)

c.) from the menu, import the desired json from the repository



d.) deploy

10.) verify by logging in to the node-red dashboard http://127.0.0.1:1880/ui/#/1