Project RoboArm

Documentation

# Iteration 1:

The first iteration will allow the raspberry pi to use its Wi-Fi module as an access point, allowing a user to connect to it using a smartphone or laptop. On this device, the user should then be able to move the robot arm using the install python script showing scroll wheels for the different servos.

# Setting up the Raspberry Pi

1. Download Raspian Buster Lite from: <https://downloads.raspberrypi.org/raspbian_lite_latest>
2. Download Etcher on a laptop/computer with a SD card reader: <https://www.balena.io/etcher/>
3. Flash the Raspian Buster Lite zip to the SD card using Etcher.
4. Enable SSH by downloading the linked file and placing it in the root of the SD card used, another method is to do this using the command line later: <https://drive.google.com/file/d/0B_BzxRImIcd-SWE2NVNPT1FycFE/view>
5. Place the SD card in the raspberry pi, plug in the power and the pi should boot into raspian lite.
6. Plug in an ethernet cable to allow the raspberry pi to connect to the internet
7. Type and run: sudo apt-get update
8. Type and run: sudo reboot
9. The raspberry is now updated and ready to be used.

# Setting up the Raspberry as a hotspot

Hostapd and Dnsmasq will be used to enable us to use the raspberry pi as a wireless access point. Hostapd is the package that lets us create a wireless hotspot using a Raspberry Pi, and dnsmasq is an easy-to-use DHCP and DNS server.

1. Type and run: sudo apt install dnsmasq hostapd (hit y to continue)
2. As these services need to be configured, type and run: sudo systemctl stop hostapd and sudo systemctl stop dnsmasq
3. To configure a static ip for the pi, type: sudo nano /etc/dhcpcd.conf
4. Go to the end of this conf and edit it so it looks like:

interface wlan0

static ip\_address=192.168.4.1/24

nohook wpa\_supplicant

1. Now run: sudo service dhcpcd restart
2. By default, the dnsmasq config contains unnecessary information, it is therefore better to start from blank using the following commands:

sudo mv /etc/dnsmasq.conf /etc/dnsmasq.conf.orig

sudo nano /etc/dnsmasq.conf

1. Type the following information into this new config file:

interface=wlan0 # Use the require wireless interface - usually wlan0

dhcp-range=192.168.4.2,192.168.4.20,255.255.255.0,24h

1. Reload the dns using:

sudo systemctl reload dnsmasq

1. Next, edit the hostapd config file (this will be an empty file): sudo nano /etc/hostapd/hostapd.conf
2. Add the following information by typing it out:

interface=wlan0

driver=nl80211

ssid=NameOfNetwork

hw\_mode=g //(or a for 5ghz)

channel=7

wmm\_enabled=0

macaddr\_acl=0

auth\_algs=1

ignore\_broadcast\_ssid=0

wpa=2

wpa\_passphrase=PassPhrase

wpa\_key\_mgmt=WPA-PSK

wpa\_pairwise=TKIP

rsn\_pairwise=CCMP

1. Now the pi has to be instructed as to where this file is located, using the following command:

sudo nano /etc/default/hostapd

1. Locate the line with #DAEMON\_CONF and replace it with:

DAEMON\_CONF="/etc/hostapd/hostapd.conf"

1. Now we can enable hostapd and start it using the following commands:

sudo systemctl unmask hostapd

sudo systemctl enable hostapd

sudo systemctl start hostapd

1. Lastly, run the following few commands to add routing and set up a masquerade for outbound traffic using the ethernet port:

* Edit /etc/sysctl.conf and uncomment this line:

net.ipv4.ip\_forward=1

* Add a masquerade for outbound traffic on eth0:

sudo iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE

* Save the iptables rule:

sudo sh -c "iptables-save > /etc/iptables.ipv4.nat"

* Edit /etc/rc.local and add this just above "exit 0" to install these rules on boot.

iptables-restore < /etc/iptables.ipv4.nat

1. Reboot the pi.
2. Using an external device, like a laptop or smartphone, a wifi connection should now appear with the name and password specified in step 10.

# Running the python script to move the arm

1. Enable I2C using the following:

sudo raspi-config

Go to Interfacing options/advanced, then I2C and enable it.

Reboot using: sudo reboot

1. As the RoboArm uses an Adafruit module to control the servos, run the following command to add support for it:

sudo pip install adafruit-pca9685

1. As the python script is on github, use the following commands to install git support and linking your account:

sudo apt-get install git

git config --global user.name "John Doe"

git config --global user.email johndoe@example.com

git clone <https://github.com/TomiEckert/ProjectRoboArm>

1. To run the robotarm python app:

python app.py

1. Connect to the created hotspot and move the scrollers for individual servos to move the robot arm.