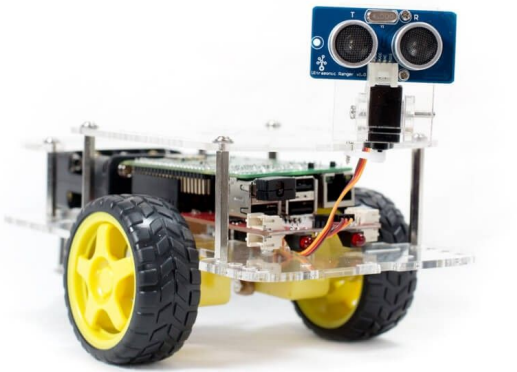
Introduction to GoPiGo

9/12/2017

What is a GoPiGo?

- The GoPiGo is a complete kit to build your own robot car.
- Easy to use API (with multiple language support)
- Raspberry Pi with Debian based OS



What you will receive

- A router
- A GoPiGo with:
  - Camera
  - Ultrasonic sensor
  - USB wifi adaptor
- Toolbox with tools
- USB connector
- Step 0 of HW1: Check that you have all of the supplies listed on the site

- Batteries not included! You can also buy a USB Battery Charger

How do I program it?

- Configure the WiFi on your GoPiGo
- SSH into your GoPiGo
- Run commands through ssh or VNC
- Moving the GoPiGo around
- Using the ultrasonic sensor
- Using the camera

Configure your wifi on the GoPiGo

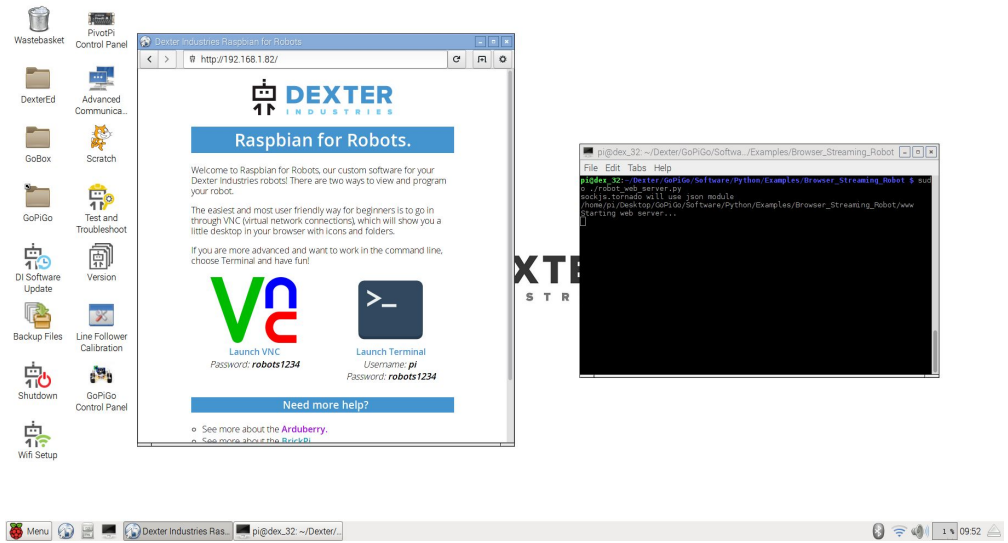
- Each of you will receive a router
- You'll need to make sure your GoPiGo connects to the router
- Make sure your batteries are charged!!!

Connecting to the router

- The wifi password is located on the bottom of the router
- Connect to the WiFi broadcasted by the router - it will likely be named either “team_X” or “TP-link”
- Connect to the router by going to tplinklogin.net after connecting
- Login to the console with username “admin” and password “admin”
- Go to DHCP->DHCP clients list and find the IP address of your GoPiGo
- Record this address, it should look like “192.168.0.101”

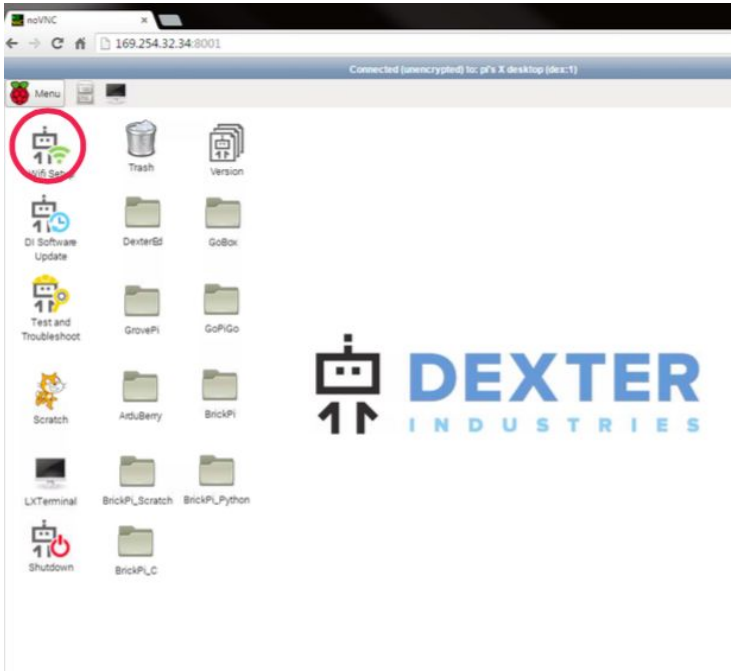
Login to the GoPiGo via VNC

- Open a browser and enter “192.168.0.101” or the IP address you found in the previous step. You should see:
- Use password robots1234



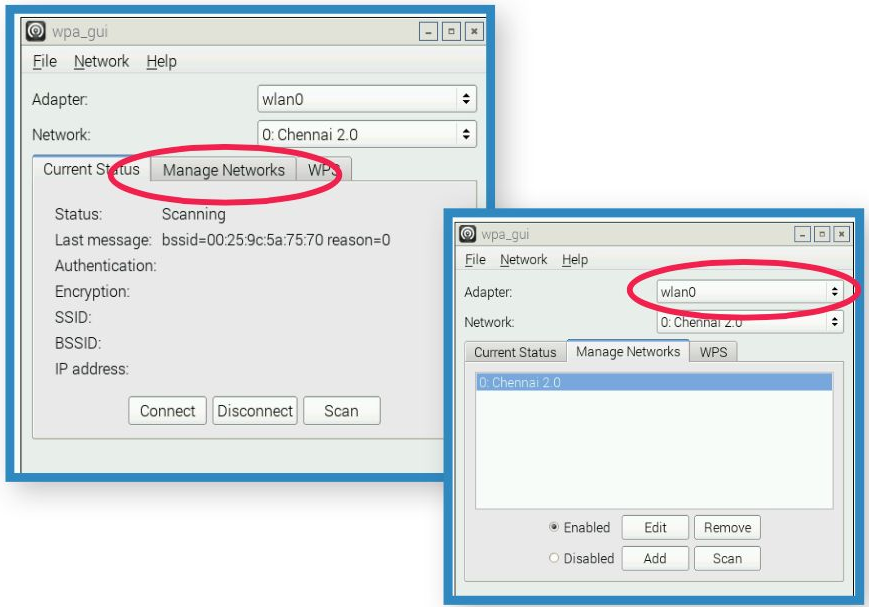
Open VNC

- The VNC view will give you a classic desktop like interface
- Open the wifi setup on the desktop



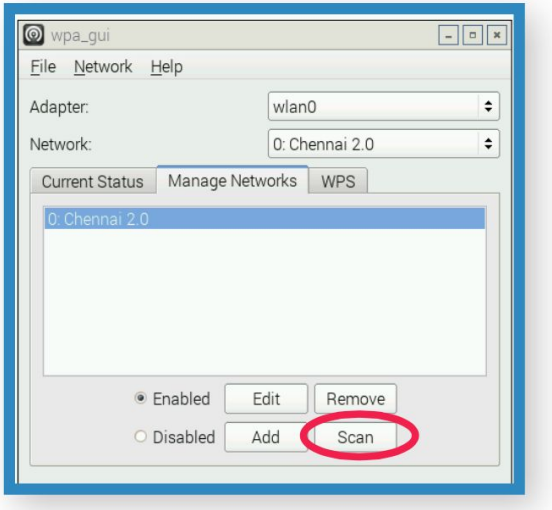
Select Manage Networks

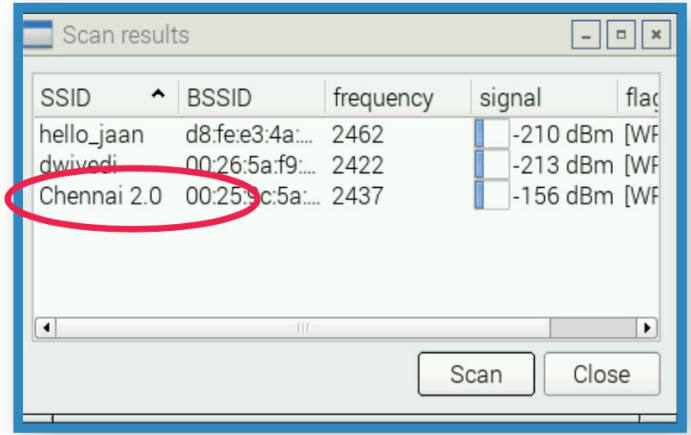
Make sure the first field of Adapter is set to “wlan0”



Scan for WiFi

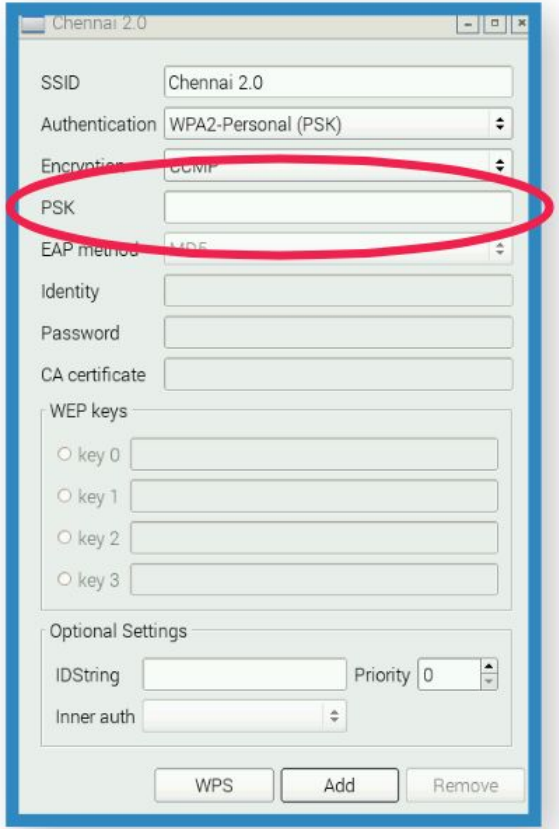
Press the “scan” button on the bottom right of the window. Then select the wifi network found in the router in a previous step





Enter WiFi password

Double check that you have entered the password that is located on the router



More comprehensive tutorial:

https://www.dexterindustries.com/GoPiGo/getting-started-with-your-gopigo-raspberry-pi-robot-kit-2/1-assemble-the-gopigo-2/assemble-gopigo-raspberry-pi-robot/1-assemble-the-gopigo2/2-connect-the-gopigo2/

Now connect via SSH!

- If you are on Windows - use Mobaxterm or Putty
- If you are on Linux/Mac you should have ssh installed already
- Once your GoPiGo and computer are on the same wifi network - enter the following in a terminal window:

  $ ssh pi@dex.local

  password: robots1234

SSH Interface

- We can now run programs
- cd ~/Desktop/GoPiGo/Software/Python/Examples/Basic_Robot_Control/basic_robot.py
- cd ~/Desktop/GoPiGo/Software/Python/Examples/Ultrasonic_Basic_Obstacle_Avoider/basic_obstacle_avoid.py

Components of basic_robot.py

- Movement functions
- Speed augmentation
- This tutorial does not cover Python - see meeting on Thursday 9/14 at 7pm for a tutorial in Python

- All Python functions are within the ‘gopigo’ module

Movement

- fwd() - Move the robot forward
- left() - turn the robot left
- right() - turn the robot right
- bwd() - move the robot backward
- stop() - stop all motion

- All movement actions will continue until a stop is called

Speed Augmentation

- increase_speed()
- decrease_speed()

- The speed of the GoPiGo can be between 0-255. The default speed is 200. increase_speed() increases the speed by 10 and decrease_speed() decreases the speed 10. Returns 1 on success, -1 on error

Putting basic_robot.py together

```
from gopigo import *
import sys
while True:
        print "Enter the Command:",
        a=raw_input()  # Fetch the input from the terminal
        if a=='w':
                fwd()   # Move forward
        elif a=='a':
                left()   # Turn left
        elif a=='d':
                right()  # Turn Right
        elif a=='s':
                bwd()   # Move back
        elif a=='x':
                stop()   # Stop
        elif a=='t':
                increase_speed()        # Increase speed
        elif a=='g':
                decrease_speed()        # Decrease speed
        elif a=='z':
                sys.exit()
        else:
                print "Wrong Command, Please Enter Again"
        time.sleep(.1)
```

Reading from the ultrasonic sensor

- us_dist(pin)
- Pin is the connection where the ultrasonic sensor is connected (in our case it is pin 15)

- This will return the distance in cms from the nearest object detected by the ultrasonic sensor.
- The sensor tends to be erroneous when pointed at an angle from an object
- Reads values up to 200cm semi-accurately

Using that in a movement paradigm

```
from gopigo import *
import time
distance_to_stop=20             #Distance from obstacle where the GoPiGo should stop
print "Press ENTER to start"
raw_input()                             #Wait for input to start
fwd()                                   #Start moving
while True:
        dist=us_dist(15)                        #Find the distance of the object in front
        print "Dist:",dist,'cm'
        if dist<distance_to_stop:       #If the object is closer than the "distance_to_stop" distance, stop the GoPiGo
                print "Stopping"
                stop()                                  #Stop the GoPiGo
                break
        time.sleep(.1)
```

More documentation

- List of all potentially useful Python functions here
- GoPiGo Github repo
- If you have questions please contact your TAs