Laser Tracker User's Manual

Developed By: Alex Oh, Ryan Taylor, Arianna Santiago

Table of Contents

Section I - Configuring the Raspberry Pi

Materials Required

- 1 SD card with capacity of at least 16 GB
- 1 Ethernet to USB adapter (if laptop does not have a ethernet port)
- 1 Raspberry Pi 4
- 1 Computer with Windows or MacOS installed
- 1 SD to USB adapter

A. Creating the Disk Image

1. Unplug the SD card from the Raspberry Pi and connect it to a PC either using an microSD to USB adapter if the computer does not have an SD card.
2. Download the Raspberry Pi Imaging Writer from https://www.raspberrypi.com/news/raspberry-pi-imager-imaging-utility/
3. (IMPORTANT!) After opening the Image Writer, click on the gear on the bottom right corner to change settings.
    a. Set the Hostname to a hostname of choice (we will use 'rpi4car')
    b. Enable ssh with password authentication
    c. Set a username and password
4. Create the disk image by following the steps in the Imaging Utility. The SD card should now be properly formatted.
5. Follow the instructions here (https://notenoughtech.com/raspberry-pi/connect-raspberrypi-pc/) to establish a static IP address between the computer and the Raspberry Pi.
    a. Note: a static IP of 192.168.137.2 will be used here
6. Safely eject the SD card from the computer and place it into the Raspberry Pi and power on the Raspberry Pi.

B. Connecting to the Raspberry Pi

7. From here, we can connect to the Raspberry Pi through SSH.
    a. Open any command prompt, powershell, etc. and type the command filling fields accordingly. An example of ours is given below.

        ```
        ssh [username]@[ip-address]
        ssh alexRPI@192.168.137.2
        ```

    b. You will likely be prompted about SHA1 Fingerprint. Type "yes" if prompted.
    c. You will be asked for the username and password. Type the information required and hit enter. (Note: you will not be able to see what you are typing for security reasons).
    d. Once you see the terminal starting with the username@ip-address from earlier, you are connected to your Raspberry Pi.
8. Run the command `sudo raspi-config`
    a. Navigate to the following settings and change the following settings
        i. Interfacing Options → Enable Camera
        ii. Interfacing Options → VNC

1. If you choose to connect to VNC, change the VNC resolution to a higher resolution.

9. Here, you can either use terminal commands or simply turn off the Raspberry Pi and plug the SD card back into a PC.
    a. Edit the file /boot/config.txt and uncomment the line `hdmi_force_hotplug = 1`

C. Connecting through VNC Viewer and Wi-Fi

10. Now, we can connect to the Raspberry Pi using VNC Viewer. Download VNC Viewer onto your PC using the following link: https://www.realvnc.com/en/connect/download/viewer/
11. After opening VNC viewer, go to File → New Connection.
    a. Enter the IP address (192.168.137.2) into the IP field.
    b. Give your connection a name. This is purely used for your VNC Viewer name.
    c. You will be prompted for username and password. Fill it out accordingly and you will come to the homescreen of the Raspberry Pi.
12. Connect the Raspberry Pi to Wi-Fi. Press the VNC button in the top right corner. An IP address should be written besides the given 192.168.137.2. Record this IP address. This is the Raspberry Pi's IP address over the wireless network.
13. Close the window of VNC viewer and create a new connection in the same way as you did the first time in VNC.
    a. This time, use the wireless IP address to connect to the Raspberry Pi. (NOTE: the laptop must be connected to the same wireless network)
    b. Again, enter the username and password.
14. Once you land on the desktop of the Raspberry Pi, feel free to unplug the ethernet cable from the Raspberry Pi, since we are now connected over Ethernet.

D. Updating the Software and Installing Needed Libraries

15. There are many required libraries and software required for this robot to operate correctly.
    a. Open the terminal (Ctrl+Alt+T) and run the following commands:
        i. `sudo apt-get update`
            1. This will update most of the important software including the Python Interpreter. Python 3.9 should be used (3.10 is not yet supported).
        ii. `sudo apt-get install -y build-essential tk-dev libncurses5-dev libncursesw5-dev libreadline6-dev libdb5.3-dev libgdbm-dev libsqlite3-dev libssl-dev libbz2-dev libexpat1-dev liblzma-dev zlib1g-dev libffi-dev`
            1. These are required for various Python libraries to behave as intended.
    b. Install the following libraries using pip
        i. `pip install opencv-contrib-python`
        ii. `pip install numpy`
        iii. `pip install pandas`

iv. `pip install scipy`
v. `pip install picamera2`

16. Clone the Freenove repository from the following Github link
    a. Enter Terminal, navigate to a suitable directory (using cd) to keep the repository, and enter the following command
        i. `git clone https://github.com/Freenove/Freenove_4WD_Smart_Car_Kit_for_Raspberry_Pi.git`
17. Clone the LaserTracker repository from the following link
    a. Enter Terminal, navigate to a suitable directory (using cd) to keep the repository, and enter the following command
        i. `git clone https://github.com/basicallyAlexOh/LaserTracker.git`
18. Power off the Raspberry Pi through VNC Viewer. Proceed to constructing the car.

Section II - Assembling the Robot

Materials Required

- Freenove 4WD Smart Car Kit for Raspberry Pi
- 1 Raspberry Pi 4
- 2 18650 3.7 V Lithium Ion Batteries
- 1 Battery Charger
- A Small Phillip's Head screwdriver

A. Physical Construction

1. Physically construct the robot by following the tutorial.pdf file at the following link (https://github.com/Freenove/Freenove_4WD_Smart_Car_Kit_for_Raspberry_Pi/blob/master/Tutorial.pdf).
2. Follow the steps starting at Page 41 (Chapter 2: Assemble Smart Car).
3. Refer to the following video if you would like a visual guide to constructing the car
    a. https://youtu.be/G3Q8xNatXgM

B. Batteries and Power

4. First, ensure that the batteries are charged. Charge the batteries using the included chargers if needed.
5. Once charged, place the batteries with the negative terminal of the batteries on the side with the spring. Adjust the battery if it is not in place.
6. Ensure that the LEDs turn on on the top face if the power button is pressed.
    a. Note: you may still power the Raspberry Pi off of the power supply (via the USB Type C port). This is highly recommended to avoid draining batteries when the car is not being used in motion.

C. Run the Test Program

7. Run the test program in the Freenove example code.
    a. Navigate to the directory in which you placed the repository.
    b. Use the command `ls` to find what files are within a directory.
    c. Use the following command to navigate to the subdirectory
        i. `cd ./Code/Server/`
    d. Ensure that the motor and servo are working by running the program as follows
        i. `python test.py Servo`
        ii. `python test.py Motor`
    e. The program will start and run. The servo should adjust a few times and move both the servos. Hitting any button on the keyboard will stop the program.
    f. The Motor test will test the car going forward, back, left, then right, then stopping.
8. Ensure that the intended behavior is observed.

Section III - Running the Laser Follower

1. Running the Laser Follower is relatively simple. Simply navigate to the directory of the Laser Tracker repository.
2. The `./FinalApplication/` folder contains all the needed code for the robot to behave as intended (including source code from the Freenove repository).
3. The general process of running the program will be as follows
    a. Place the car on the ground at the starting point
        i. Make sure to note how the robot is angled, as this will significantly alter the behavior of the robot on retracing its path.
        ii. In terminal (connected through VNC viewer), run the following command after navigating to the `FinalApplication` folder using `cd`
            1. `python main.py`
                a. If it does not work, try `sudo python main.py`
                b. Otherwise, install additional packages if they are missing.
        iii. Turn on the blue laser and point it at the ground in front of the robot. Slowly trace out a path approximately 20-30 cm in front of the robot. The robot will catch up to the position of the laser.
        iv. Turn off the laser at the end of the path.
        v. Within the next 10 seconds, place the robot at its initial position (making sure that the angle is approximately correct.
        vi. The robot should now retrace the original path that was pointed to with relatively precise accuracy.