1. Make a bootable USB thumb drive with file: Lubuntu-DonkeyCar_vX.Xx-17.04.iso
   Windows users can try Rufus(https://rufus.akeo.ie/)
   Mac users can try Etcher(https://tutorials.ubuntu.com/tutorial/tutorial-create-a-usb-stick-on-macos#0)
   Linux users can use unetbootin (sudo apt install unetbootin)

2. Boot up your computer from the USB thumb drive. The process varies from computer to computer so determine what method works for you. On Lenovo laptops it's typically holding down the Fn+F12 buttons as soon as the large "Lenovo" name shows up on bootup. Then selecting "UEFI USB" device option.

   hit enter when the "try Lubuntu without installing" option is presented and wait for the desktop to appear.

3. Connect your computer to your preferred WiFi network using the Networks icon found in the bottom right corner. A few default documentation web pages are presented when you open up the web browser by double clicking on the Firefox icon on the desktop.

4. Create your Raspberry Pi image.
   Also setup its hostname and WiFi by inserting a 8GB or more uSD card(cancel any dialogs wanting to display the uSD) and then opening up a Terminal window and typing:
   script-zipdd ~/Downloads/donkey_2.2.img.zip yourname-donkeycar
   Then enter the questions about your Wifi setup and wait until it finishes.

5. Remove the uSD card and put it into the Raspberry Pi on your DonkeyCar and boot it up. In about 1 minutes you can ssh into it using the terminal window like this:
   ssh pi@yourname-donkeycar.local
   BUT, before you do that, let’s upgrade and setup the donkeycar software on the rPi to the latest from the Master branch in github(https://github.com/wrosco/donkey). Do this with:
   script-rpi-donkeyUpdate NOTE: Just run **script-rpi-donkeySetup** to setup without github updates. You may be prompted to accept the authentication for the connection. Type yes and hit enter to allow your computer to communicate with the rPi remotely.

6. It's recommended you change the default login password using the raspi-config program by remotely logging into the rPi(using **ssh pi@yourname-donkeycar.local**) and then entering:
   sudo raspi-config
   Besides changing your password, check to be sure the Camera and I2C options have been enabled. You will find them in menu item 5, “Interfacing Options”. Under advanced options is the option to expand the filesystem to use all of the uSD and this too is recommended. **Do not reboot** if you will be adding Bluetooth and PS3 Controller pairing.

7. Customizing your setup: If you’d like to use the PS3 Bluetooth controller, now would be the time to edit the config.py file located in the /home/pi/d2(~/d2) directory. Set the following:
   ```
   USE_JOYSTICK_AS_DEFAULT = True
   ```
   Also, other things which might need customizing are also found in this file(~/d2/config.py). Things like: STEERING_LEFT_PWM = 460

```
STEERING_RIGHT_PWM = 290
THROTTLE_FORWARD_PWM = 500
THROTTLE_STOPPED_PWM = 350
THROTTLE_REVERSE_PWM = 220
```

To enable use of bluetooth on the rPi and start the pairing process with your Sony PS3 Controller, run this in a new terminal or after logging out of the ssh shell: **script-bluetoothPS3-setup**
It'll halt the system when done and hitting enter in the terminal window exposes what Bluetoothctl commands need to be run after removing power and rebooting. Use **script-bluetoothPS3-final** to remotely run the bluetoothctl on the rPi and finish the bluetooth setup.

8. Get training data(tubs) by driving your Donkey Car around the track to accumulate images and steering/throttle data. Start driving with:
   script-drive
   If using the webController, point your browser to yourname-donkey.local:8887 and pick the joystick option and drive.

9. Get your training tub data from your car to your local d2/data directory:
   First, create you local directory with **script-local-createcar** and then run:
   script-rsync-data

10. Train your pilot(myPilot) using this command:
    script-train your-tub-name or **script-train all** or **script-train your-tub-name,your-tub-name,...**
    Note: just running script-train will present you with a list of your tubs.

11. Copy your pilot(myPilot) back to your DonkeyCar using:
    script-rsync-pilot

12. Run your pilot with this:
    script-auto-drive

Scripts Listing of /home/lubuntu/bin

script-zipdd - script-zipdd <path to donkey_2.img.zip file> <yourname-donkeycar> [uSD device]
copies Donkey rPi image file to uSD and customizes it for your preferred WiFi router and sets the rPi hostname and sets the rPi hostname for all commands in script-config.

script-rpi-donkeyUpdate - script-rpi-donkeyUpdate [yourname-donkeycar.local]
Runs git pull, pip install, donkey createcar, etc to get the latest donkey software on the rPi and a new d2 directory(~/d2).

script-rpi-donkeySetup - script-rpi-donkeyUpdate [yourname-donkeycar.local]
Runs donkey command createcar to create a new d2 directory(~/d2).

script-local-createcar - script-local-createcar

Runs donkey command to create the d2 directory(~/d2)

script-remote-createcar - script-remote-createcar [yourname-donkeycar.local]

Runs donkey command to create the d2 directory(~/d2) on the rPi

script-rsync-data - script-rsync-data [yourname-donkeycar.local]

Copies your training data tubs(~/d2/data/*) from your Donkey Car to your computer(~/d2/data/*)

script-train - script-train <tub directory name> [tub name,tub name,... | all]

Runs the Tensorflow training on your tub data

script-rsync-pilot - script-rsync-pilot [yourname-donkeycar.local]

Copies your trained pilot(~/d2/models/myPilot) from your computer to your Donkey Car(~/d2/models/myPilot)

script-drive - script-drive [yourname-donkeycar.local]

Runs ssh to your Donkey Car and runs the drive command.

script-auto-drive - script-auto-drive [yourname-donkeycar.local]

Runs ssh to your Donkey Car and runs the drive command with model, myPilot.

script-bluetooth-setup - script-bluetooth-setup [yourname-donkeycar.local]

Loads needed software and enables auto start of the Bluetooth service. It automatically halts/shuts down the rPi and power must be removed and replaced for the Bluetooth hardware to be started properly

script-bluetooth-final - script-bluetooth-final [yourname-donkeycar.local]

Finishes the Bluetooth pairing process between the rPi and PS3 Controller. The user is required to connect the controller to the rPi over USB cable and to enter information listed prior to prompts. When finished you should see “js0” listed with other input devices.

script-reauthorize-ssh - script-reauthorize-ssh [yourname-donkeycar.local]

Reauthorizes automatic remote login which is lost with each reboot of this bootable ISO.
Enabling persistent storage would eliminate the need for reauthorization.
Look for information on enabling Persistent Storage in your favorite Bootable USB tool.

VERSION HISTORY

0.1c Adds updated script-zipdd with WiFi and hostname setup. (/home/lubuntu/bin)

Trimmed wpa_supplicant.conf file (/home/lubuntu/Downloads)
Added HowToDonkey.txt file to desktop. (/home/lubuntu/Desktop)
Added “Drive your car” setup page to default Firefox home pages.

0.1d Change to Lubuntu 17.04 base for working persistent storage boot options.
Add disabling wifi sleep in wlan0.conf
Fixed missing wlan0.conf file in Downloads directory
Update HowToDonkeyCar.txt changed to HowToDonkeyCar.pdf
-Added document listing of scripts
Added script-drive to the list of usable scripts
Added local and remote versions of createcar script
Added script-auto-drive to start driving using generated myPilot

0.1e Change to default wpa_supplicant.conf to contain default networks
Added script-rpi-donkeyUpdate to get the rPi to the latest version of donkey software.
Added script-setup-bluetooth placeholder script for update to the joystick bluetooth
Updated this HowToDonkeyCar.pdf to include the new script and what might be customized in the ~/d2/config.py file.
Fixed script-train due to changes in how the passed arguments worked.

0.1e1 Added script-setup-bluetooth script for update to the joystick bluetooth
Updated this HowToDonkeyCar.pdf to include reference to the bluetooth setup script

0.2 Added script-bluetoothPS3-setup and script-bluetoothPS3-final scripts for updates to use the PS3 SixAxis bluetooth controller.
Added Broadcom WiFi drivers.
Changed wpa_supplicant network priorities(10,20,30).
Fixed script-zipdd to use the WiFi already connected to and use optional USB devices for uSD image copying..
Updated this HowToDonkeyCar.pdf to include reference to the bluetooth scripts and above noted changes.

0.2b Add script-halt to remotely halt the rPi
Add script-config to hold customizations like defined SSID and/or DonkeyCar hostname
Change all scripts to source script-config and use myHostNameLocal if no hostname is specified.
Add ssh authentication which eliminates requireing a password when logging in from the Live ISO when used with persistent boot option.
Add script-reauthorize-ssh to help with reauthorizing remote login after rebooting the computer.

0.2c Updated to rPi image file donkey_2.2.img.zip
Updated scripts to adjust for the changes required for the new rPi software
Added the DonkeyCar Simulator for Linux software. To be found in the DonkeySimLinux directory

0.2c1 Fixed incorrect names of the two Bluetooth scripts: script-bluetoothPS3-setup / script-bluetoothPS3-final.
Added file script-rpi-donkeySetup which sets up the rPi donkey(d2) directory without a github update.

0.2c2 Updated to donkey v2.2.1 and fixed bug in USB uSD adapter code of script-zipdd.

0.2c3 Updated script-train to use “all” and also multiple tubs via specifying tub dirs like “tub1,tub2,tubx”
Added net-tools/ifconfig to the preinstalled packages.

0.2c4 Added Donkey-tools-notes.txt to Documents folder to help use available tools. Fixed zip filename error.
Added Python development IDE (Eric Python IDE - http://eric-ide.python-projects.org/) and runnable using eric6 command. Updated the desktop background image with Donkey operational life cycle.

Proposed

0.3 Add a Python development IDE (Eric Python IDE - http://eric-ide.python-projects.org/)
Initial Service Daemon with WebSocket I/F for switching from driving to auto-pilot
Initial Bluetooth controller option for turning WiFi On/Off