

MANTA 3 Tuto : Quick start and basic configurations
-- Mars 2017--
-- sahbi

Connexion à la carte RPI 2

le mot de passe de la carte: "teammanta"

reboot la RPi 2 après chaque modification des fichiers de connexion: \$sudo reboot

1- ssh:

- attribuer adresse statique à carte ethernet ordi : ifconfig xxxx 192.168.50.1 (xxxx le nom de la carte donné par ifconfig)
- connaitre son adresse ip : hostname -I
- attribuer une adresse statique à la carte RPI 2: 192.168.50.42
 - en rajoutant les addr: inet, netmask, Bcast, gateway dans le fichier **/etc/network/interfaces**
- les adresses sont obtenues par la commande **sudo ifconfig**
- il faut tjrs faire de **sudo reboot**
- Pour changer le mot de passe: sudo passwd (reboot nécessaire ensuite)

more :

sous le fichier : **/etc/network/interfaces (modification en mode super user: sudo!)**

#Configure loopback interface

auto lo

iface lo inet loopback

Configure eth0 interface with static IP

auto eth0

iface eth0 inet static

address 192.168.50.42

netmask 255.255.255.0

broadcast 192.168.50.255

gateway 192.168.50.1

2. reboot from the prompt command

3. ssh-keygen -f "/home/mallouli/.ssh/known_hosts" -R 192.168.50.42: to

remove the previous hosts (from your machine)

4. ssh pi@192.168.50.42 then "yes"

5. pwd = teammanta

NB: uncomment in server (RPi2)

/etc/ssh/sshd_config/ **#PasswordAuthentication yes**

2- internet:

commenter les lignes de codes ci-dessus et rajouter :

#Configure loopback interface

auto lo

iface lo inet loopback

Configure eth0 interface

auto eth0

address 147.250.35.250

masque 255.255.255.0

gateway : 147.250.35.245

Tuto connect depuis internet :

<http://stackoverflow.com/questions/16040128/hook-up-raspberry-pi-via-ethernet-to-laptop-without-router>

ROS:

après la création d'un workspace ROS exemple : catkin_ws

do:

source ~/catkin_ws/devel/setup.bash (à près chque lancement d'un terminal)

à chque changement de CMakeLists.txt il faut faire un catkin_make dans le dossier

Ros_workspace (ex. catkin_ws/)

pour lancer un noeud:

terminal 1 : roscore

terminal 2: rosrun nom_pkg ros_node_name

NB: - pour les noeuds python il faut changer le mode d'accès aux fichier source avec la commande chmod a+x python_ros_node

- pour les noeuds cpp: après un catkin_make, copier l'exécutable produit dans le dossier où il y a le code source du noeud ros

Télécommande HK62

(réf :

https://hobbyking.com/en_us/hobbyking-hk6s-2-4ghz-fhss-6ch-tx-rx-mode-2.html?__store=en_us)

HobbyKing HK6S 2.4Ghz FHSS 6Ch Tx & Rx (Mode 2)

- 1) plug bind plug(in "id set" position)
- 2) turn on the reciever(plug esc/ubec/battery)
- 3) turn on the transmitter
- 4) wait a few seconds(10 must be enough)
- 5) unplug the bind plug
- 6) turn of the reciever
- 7) turn on the

reciever again and put a servo to test it If it works then you are done. If it doesn't work repeat the steps.

Imu MP-9150

web:

<http://ozzmaker.com/berryimu/>

<http://blog.bitify.co.uk/2013/11/interfacing-raspberry-pi-and-mpu-6050.html>

shell commands:

Once you have the board connected you can test to see if the Pi has detected it. This is done with the following command to install the i2c tools

```
sudo apt-get install i2c-tools
```

and then either

```
sudo i2cdetect -y 0 (for a Revision 1 board like mine)
```

or

```
sudo i2cdetect -y 1 (for a Revision 2 board)
```

then you should see output showing any I²C devices that are attached and their addresses

test sur chip: sur les matos qu'on a

imu HMC5883L n'est pas fonctionnelle: addr 0x1e: wrong information

imu pololu imu01c non plus

ce qui marche

IMU ITG/MPU 6 broche du labo : MPU-9150, addr: 0x68

#preuve script imu_test.py

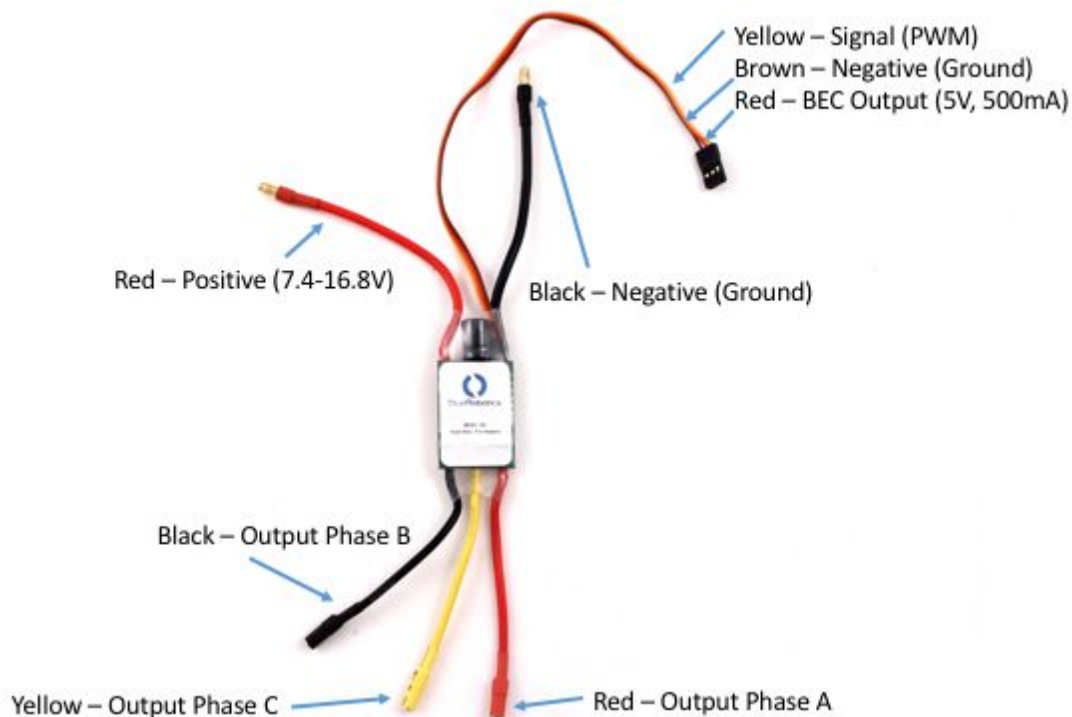
Blue ESC (Speed controllers)

Quick Start

(réf: <http://docs.bluerobotics.com/besc/>)

1. Connect the three motor wires to the motor. The order of connections does not matter, however, switching any two wires will change the direction of the motor. The output phases A, B, and C are completely interchangeable

2. Connect the red power wire and black ground wire to a power source like a battery. You will hear a few beeps from the ESC.
3. Connect the signal cable to your signal source like an RC radio receiver or microcontroller board. The yellow wire is the signal wire. The red wire is the battery eliminator circuit (BEC) output, which supplies 5V at 500mA to power a control system. The brown wire is ground.
4. Send a stopped signal (1500 microseconds) for a few seconds to initialize the ESC. You will hear a long tone.



GPS phidgets

http://www.phidgets.com/products.php?product_id=1040

penser à installer la librairie libphidgets

Installing

réf(http://www.phidgets.com/docs/OS_-_Linux)

To install the libraries, follow these steps:

1. Install libusb-1.0 development libraries - libusb-1.0-0-dev.

- Note that libusb-1.0 may be already on your system, but the development libraries probably aren't.
 - Search for libusb-1.0-0-dev or similar in your distribution package directory.
 - Or install [from source](#).
2. Unpack and install the [Phidget Libraries](#)
- From the main unpacked libraries directory, run:
 - `./configure`
 - `make`
 - `sudo make install`
 - This will compile `phidget21.h` and place the library into your gcc path

Note: Although these libraries are written in C, the additional libraries for Python, Java, and most other Phidget-supported languages depend on them.