

# **ESTISIMbot**

## **Qu'est-ce que l'ESTISIM**

L'ESTISIM (Etudes Scientifiques et Techniques de l'ISIMs) est une ASBL gérant les contrats de recherche appliquée entre les entreprises ou les administrations et l'ISIMs.

Pour de plus amples informations : <https://www.heh.be/article-24-asbl-ESTISIM>

## **Caractéristiques de l'ESTISIMbot**

L'ESTISIMbot à un numéro d'identification unique qui est repris sur le châssis et sur son Raspberry. Son hardware se compose d'un Raspberry v3, d'un contrôleur moteurs PiconZero et d'un suiveur de ligne. L'installation de la bibliothèque du PiconZero a déjà été effectuée et se trouve dans le dossier **home** du compte utilisateur. Si la bibliothèque a été effacée, il suffit d'exécuter le script **piconz.sh** qui va recréer ce répertoire. L'interface graphique a été désactivée ainsi que la redirection X11 via le SSH. Le service SSH est disponible et fonctionnel. La carte WiFi a été configurée afin de créer un réseau WiFi local unique pour ce Raspberry et donc ce robot. La carte Ethernet n'est pas redirigée vers le WiFi mais permet tout de même de se connecter à un autre réseau.

## **Comment utiliser un ESTISIMbot ?**

### **Se connecter**

Grâce au numéro d'identification unique **x** du robot, vous allez pouvoir identifier son point d'accès WiFi :

- ESSID : ESTISIM-Rpi-**x**
- Channel : **x**
- Sécurité : WPA2/PSK (CCMP)
- Clef d'accès : Raspberry@HEH

Une fois connecté au point d'accès WiFi, vous allez obtenir une adresse 192.168.2.1/24 avec comme première adresse celle du Raspberry. Vous pouvez à présent utiliser FileZilla pour le transfert SFTP ou vous connecter en SSH à l'aide du compte :

- Username : student
- Password : HEH-Powa

### **Se connecter à Internet**

Si vous devez transférer des fichiers sur votre *Raspberry* depuis Internet, il est préférable d'envisager de les télécharger en local et de les transférer via SFTP. Dans le cas contraire, vous pouvez utiliser une clef USB.

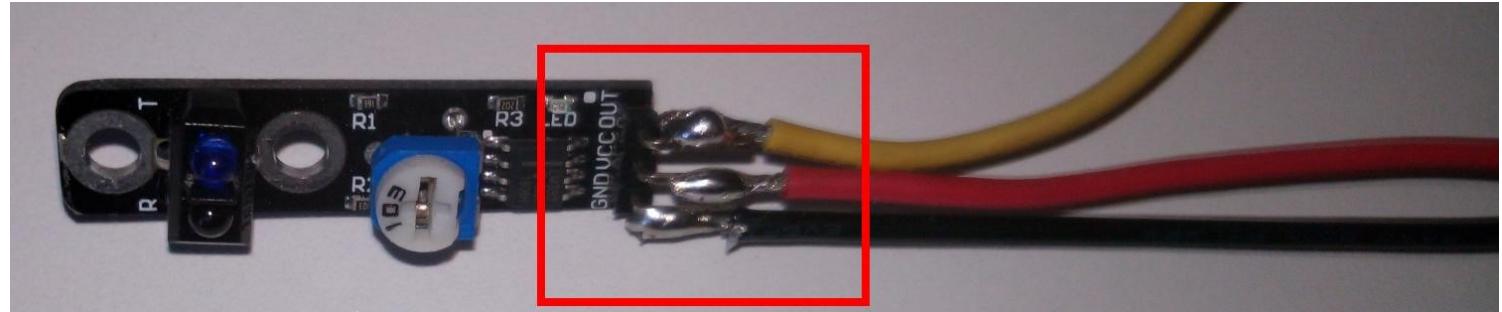
Si vous devez vraiment accéder à Internet pour une raison quelconque, il vous faudra connecter votre Raspberry au réseau Ethernet et vous connecter au Raspberry via son point d'accès WiFi.

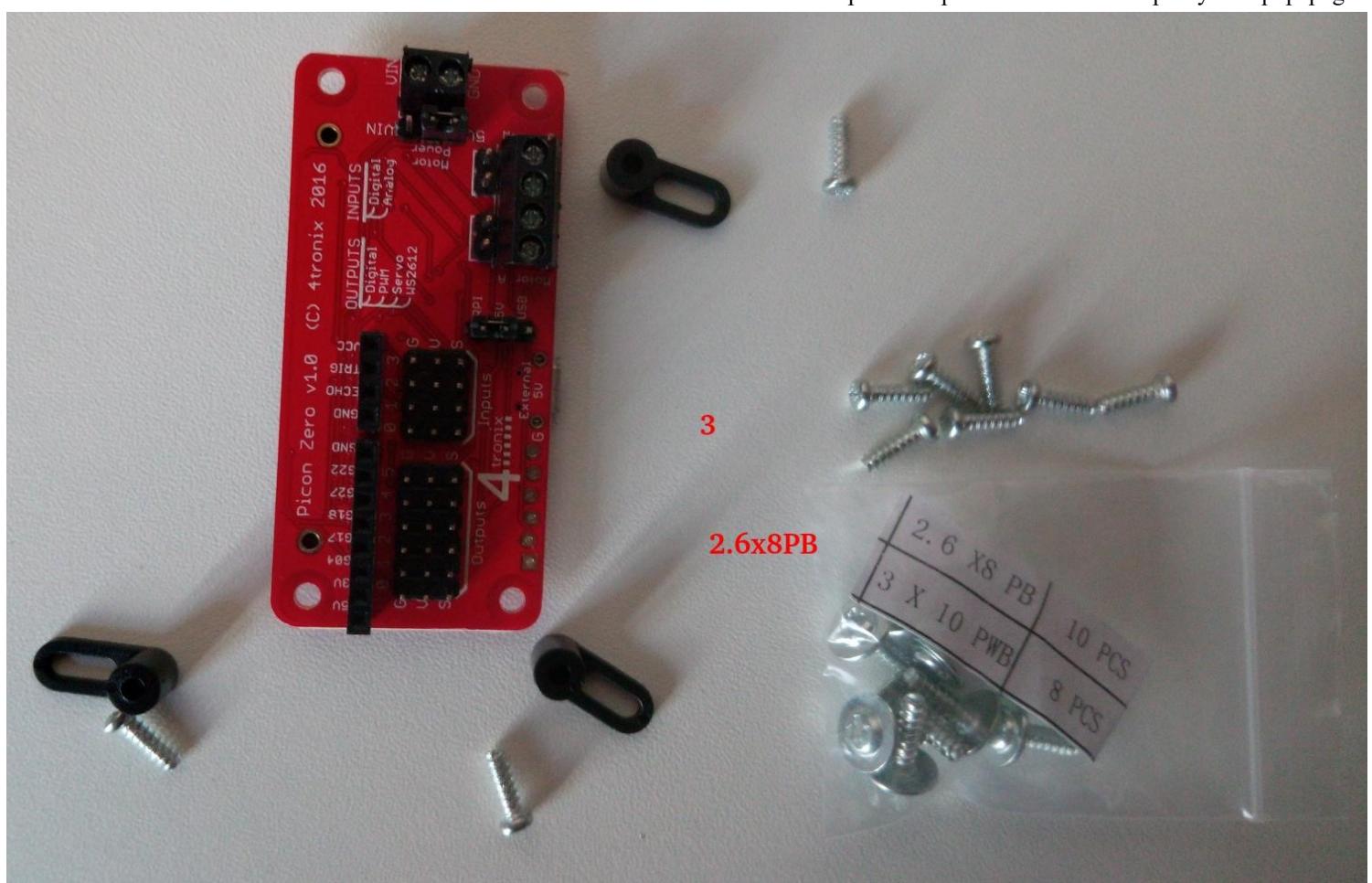
### **Bibliothèque Python**

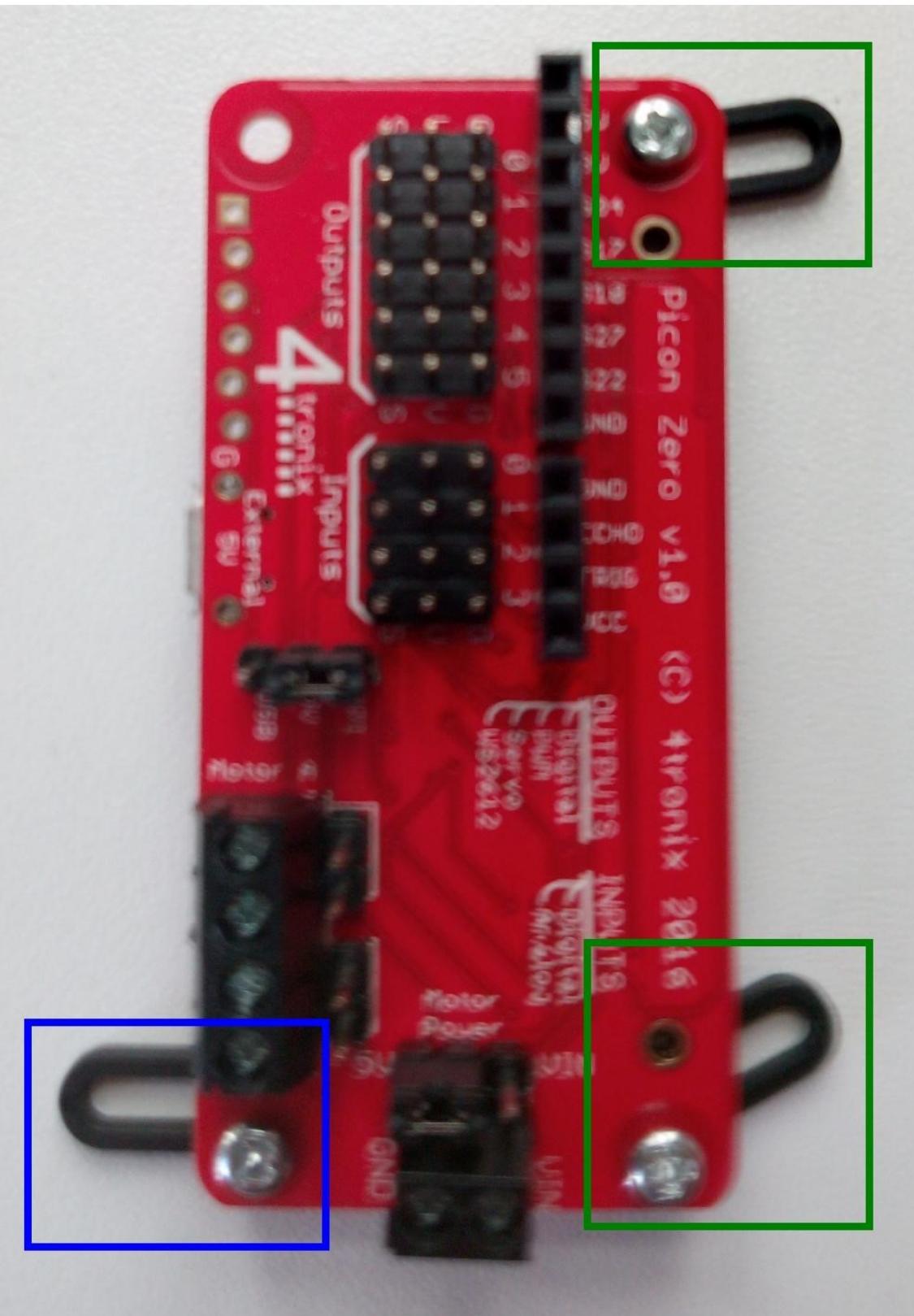
Une bibliothèque Python a été implémentée conformément à l'installation du robot ESTISIM. La classe Python permettant de contrôler ce robot s'appelle **ESTISIMBot**. Pour de plus amples informations, veuillez consulter la page HEHBot.

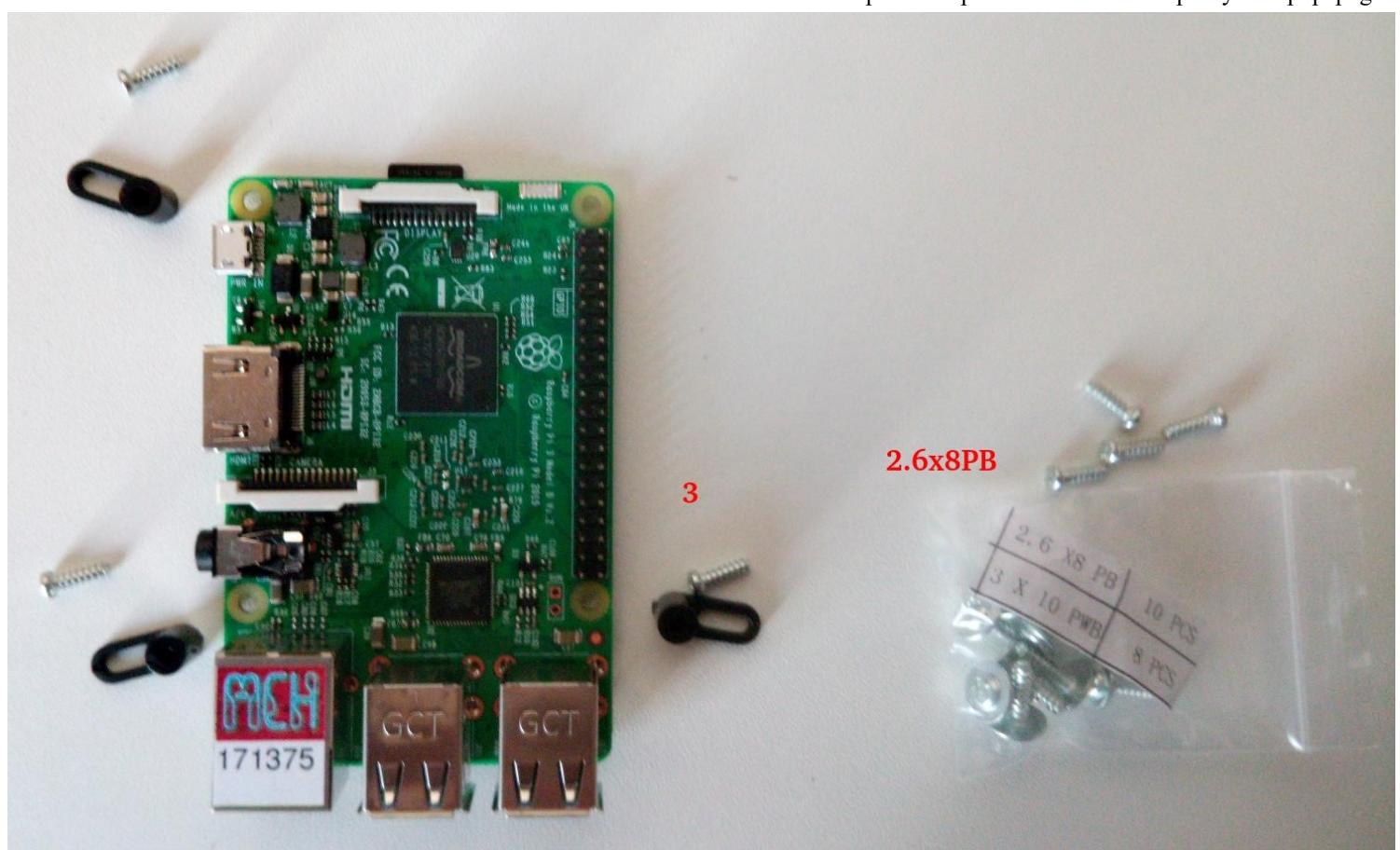
## **Installation**

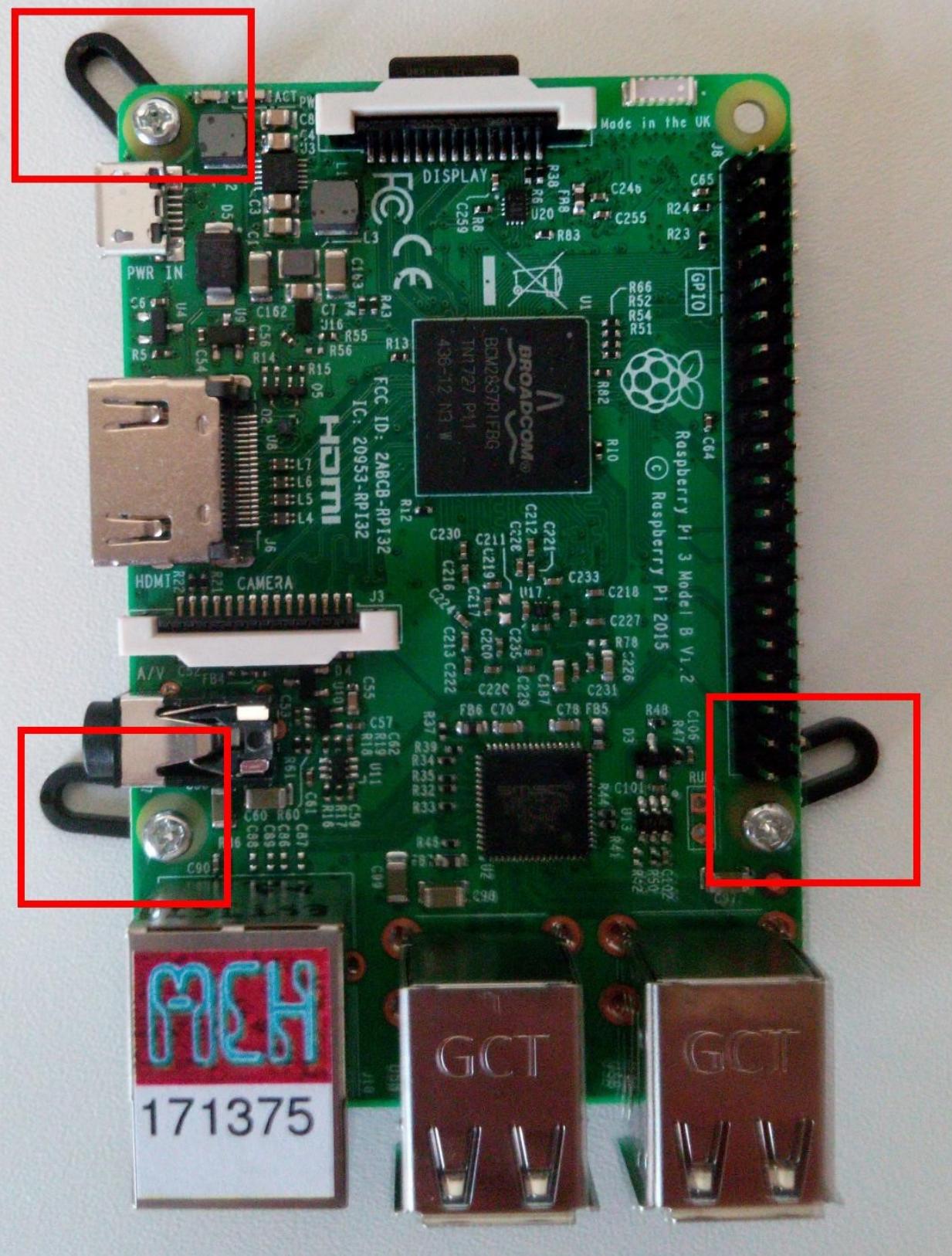
### **Matérielle**

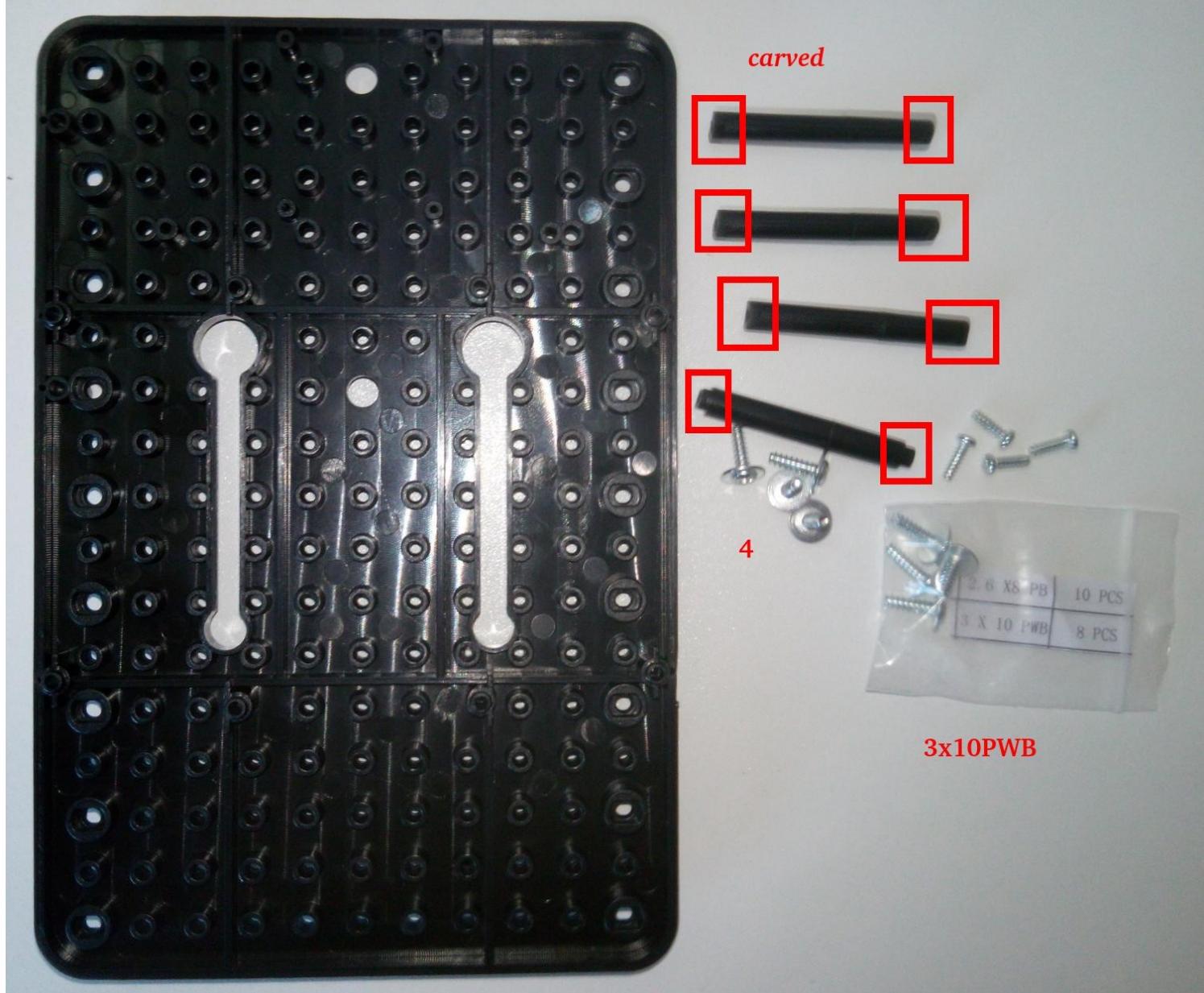


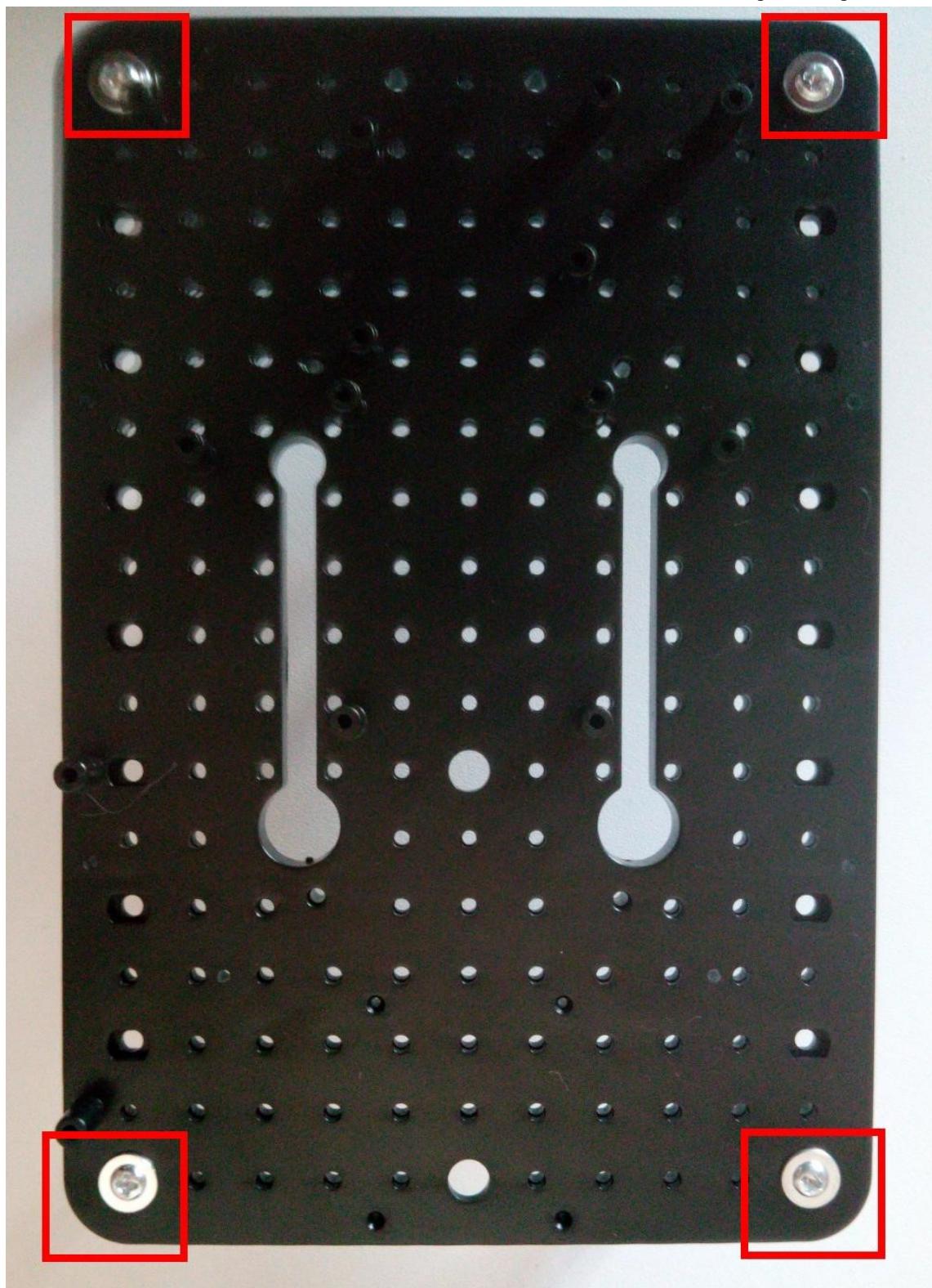


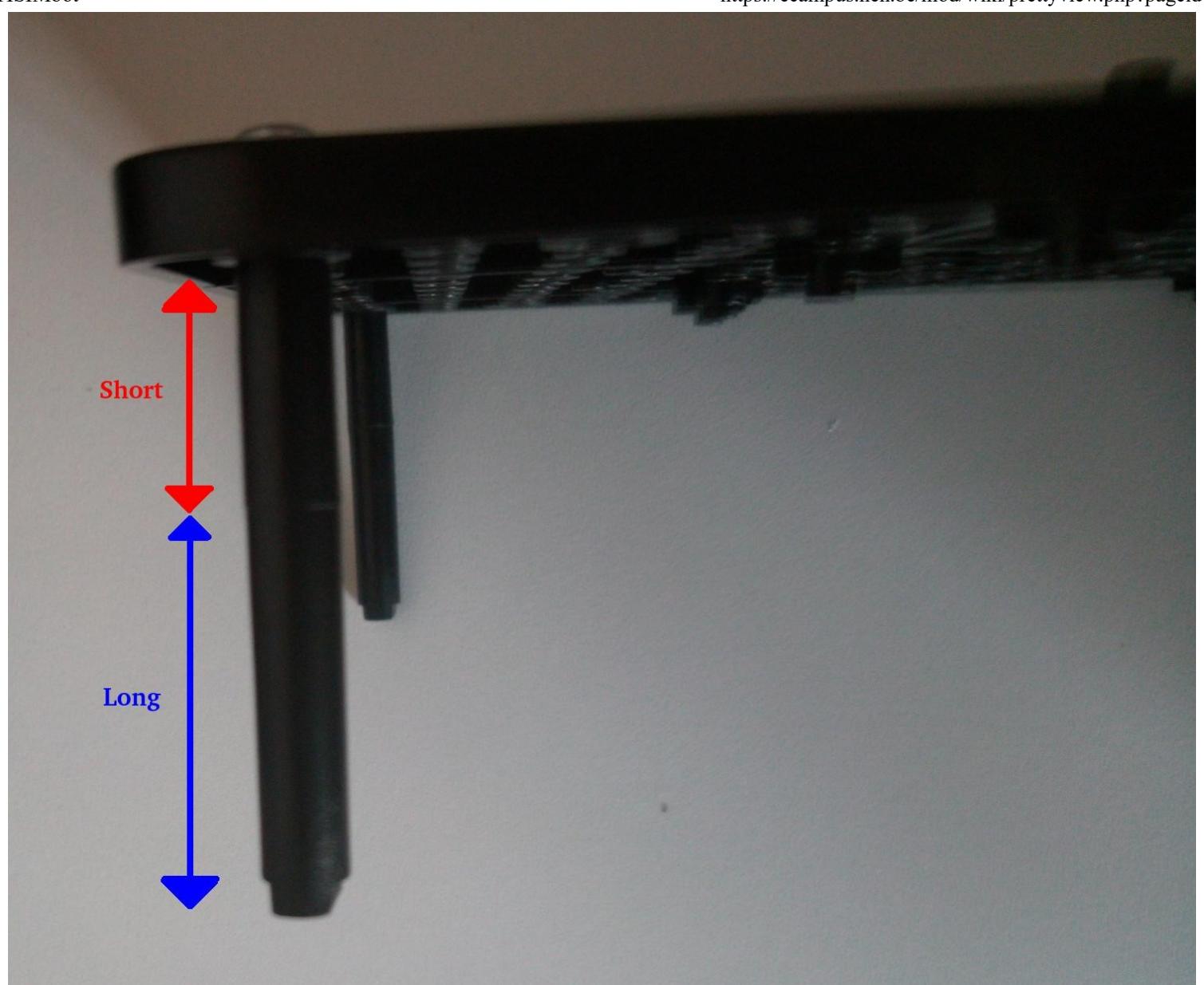




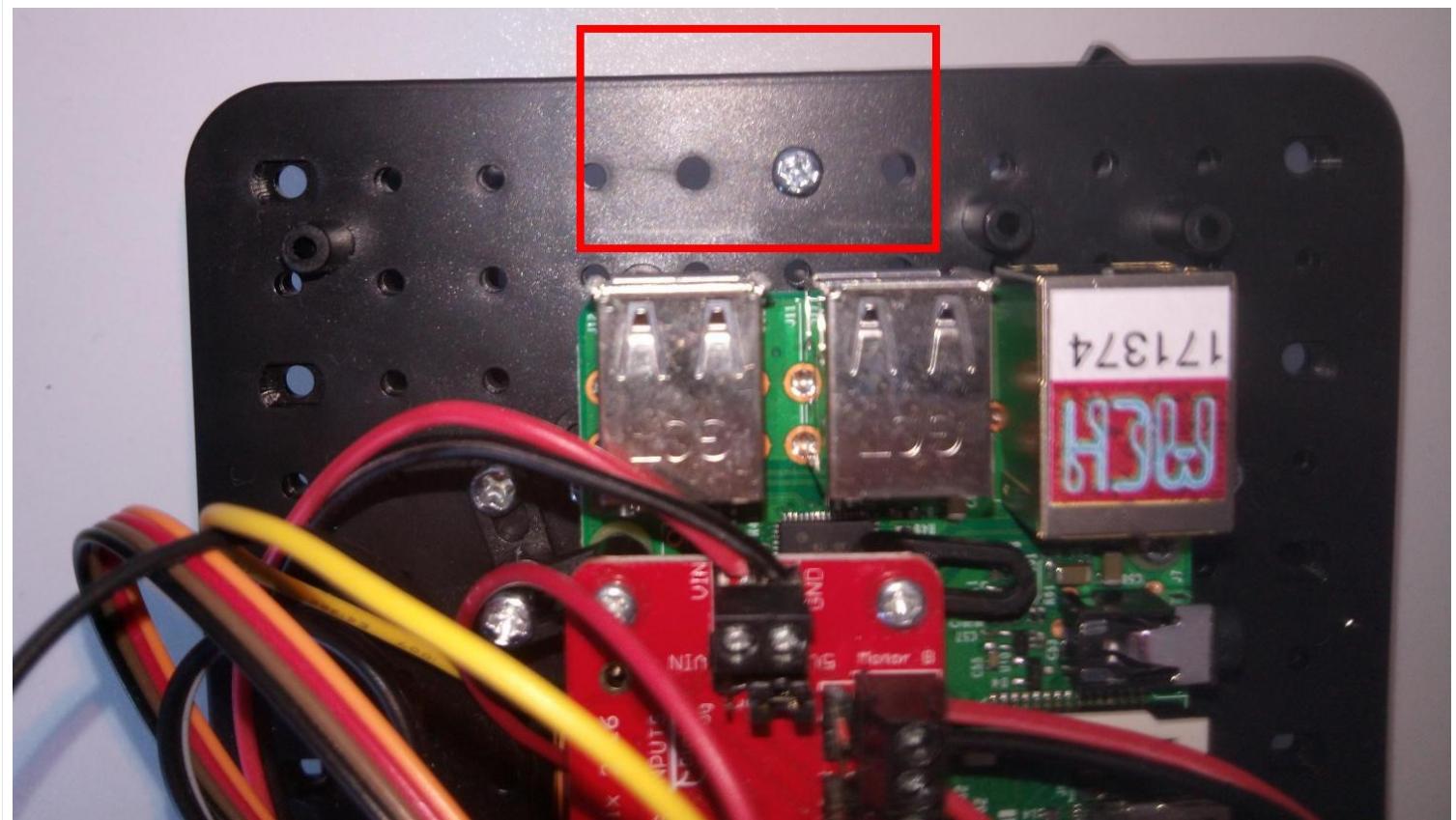
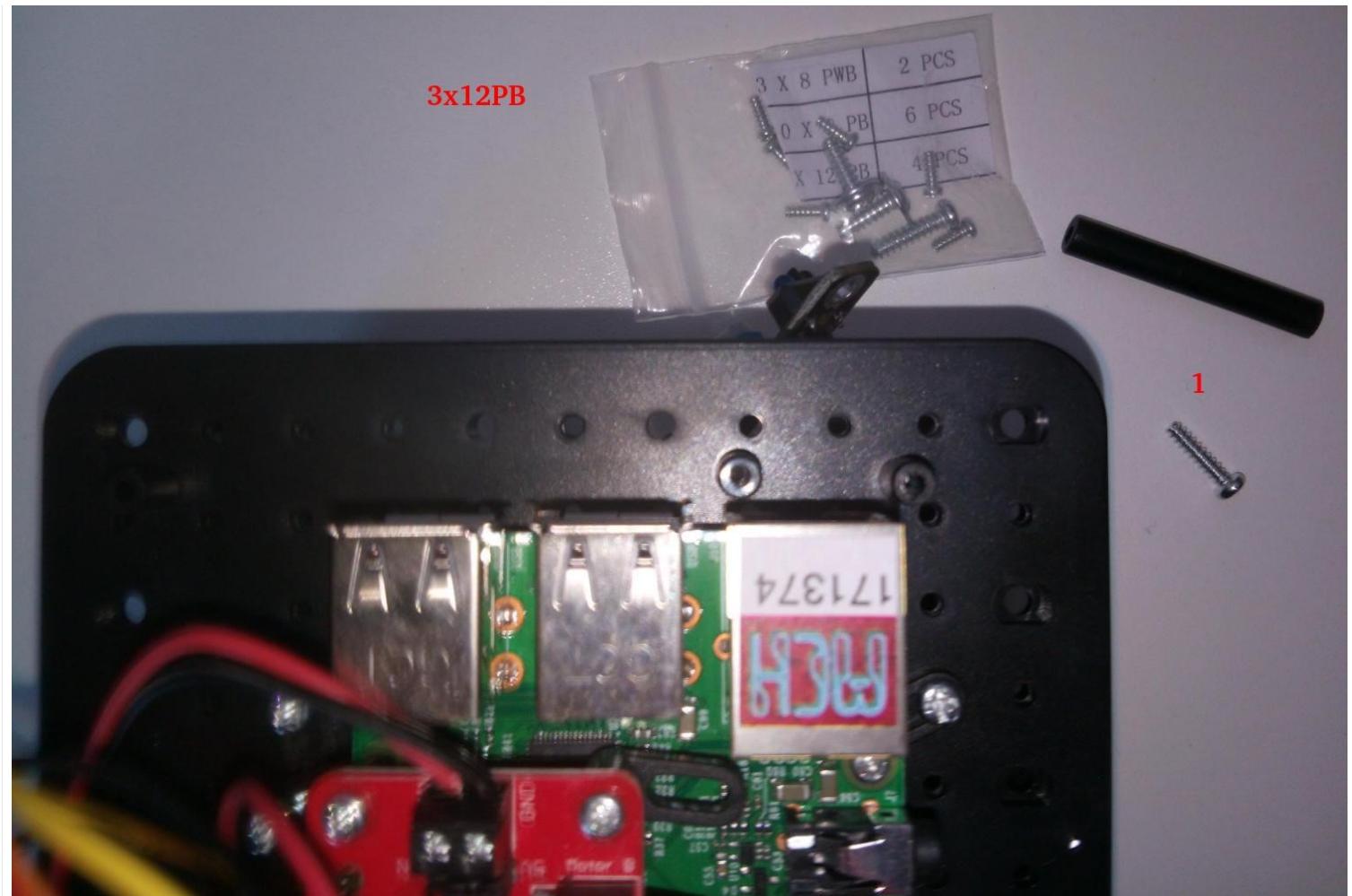


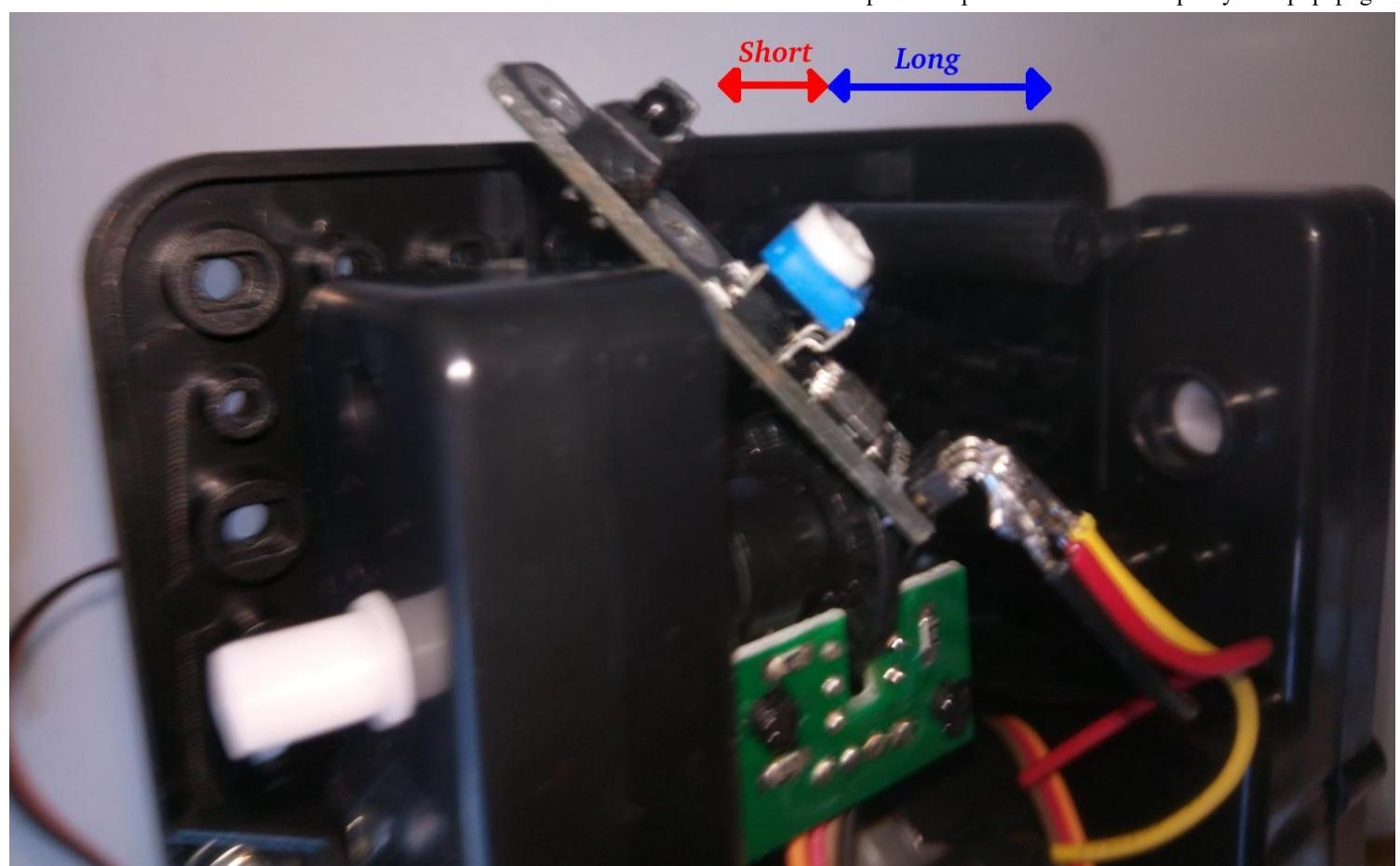








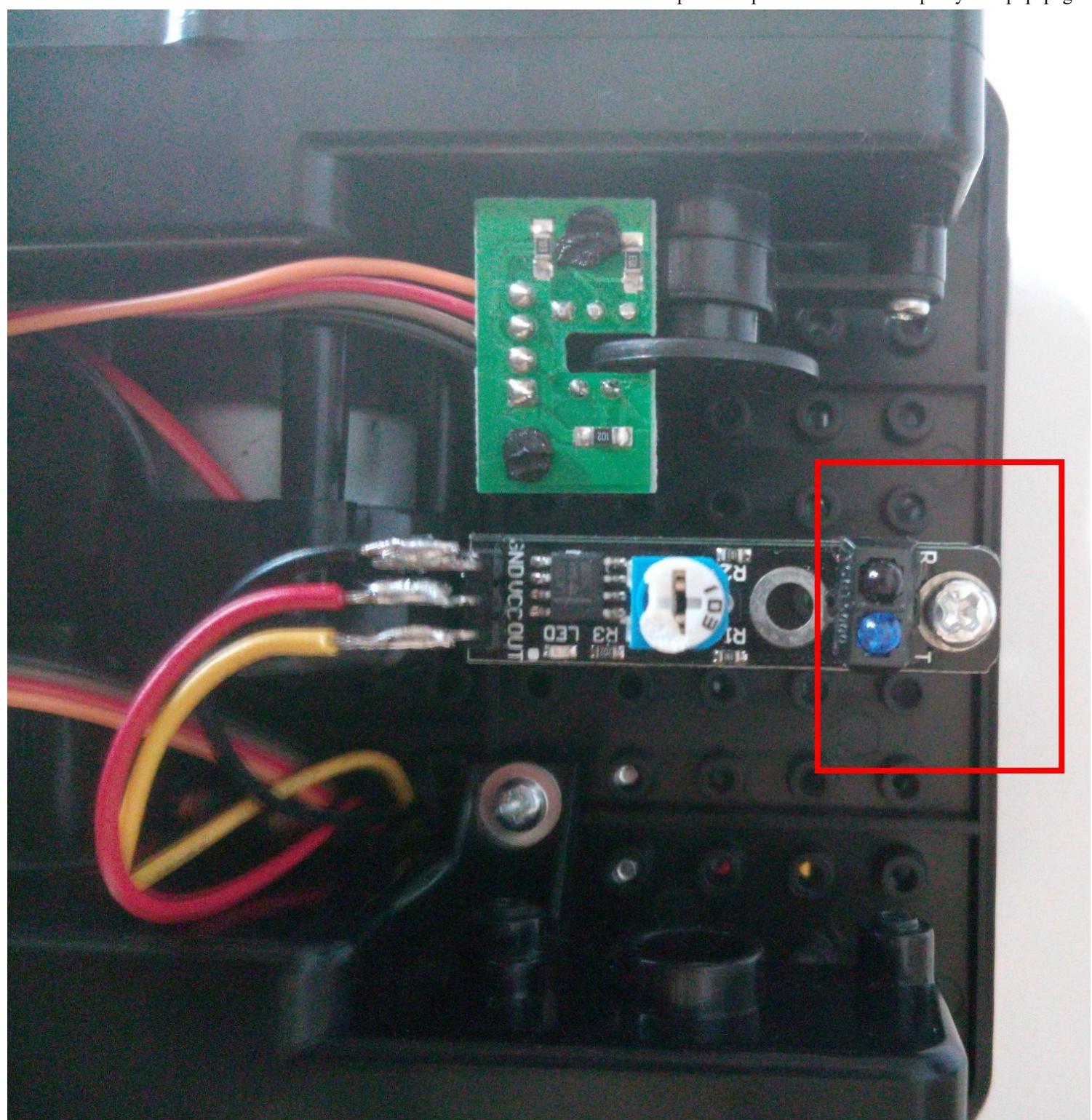


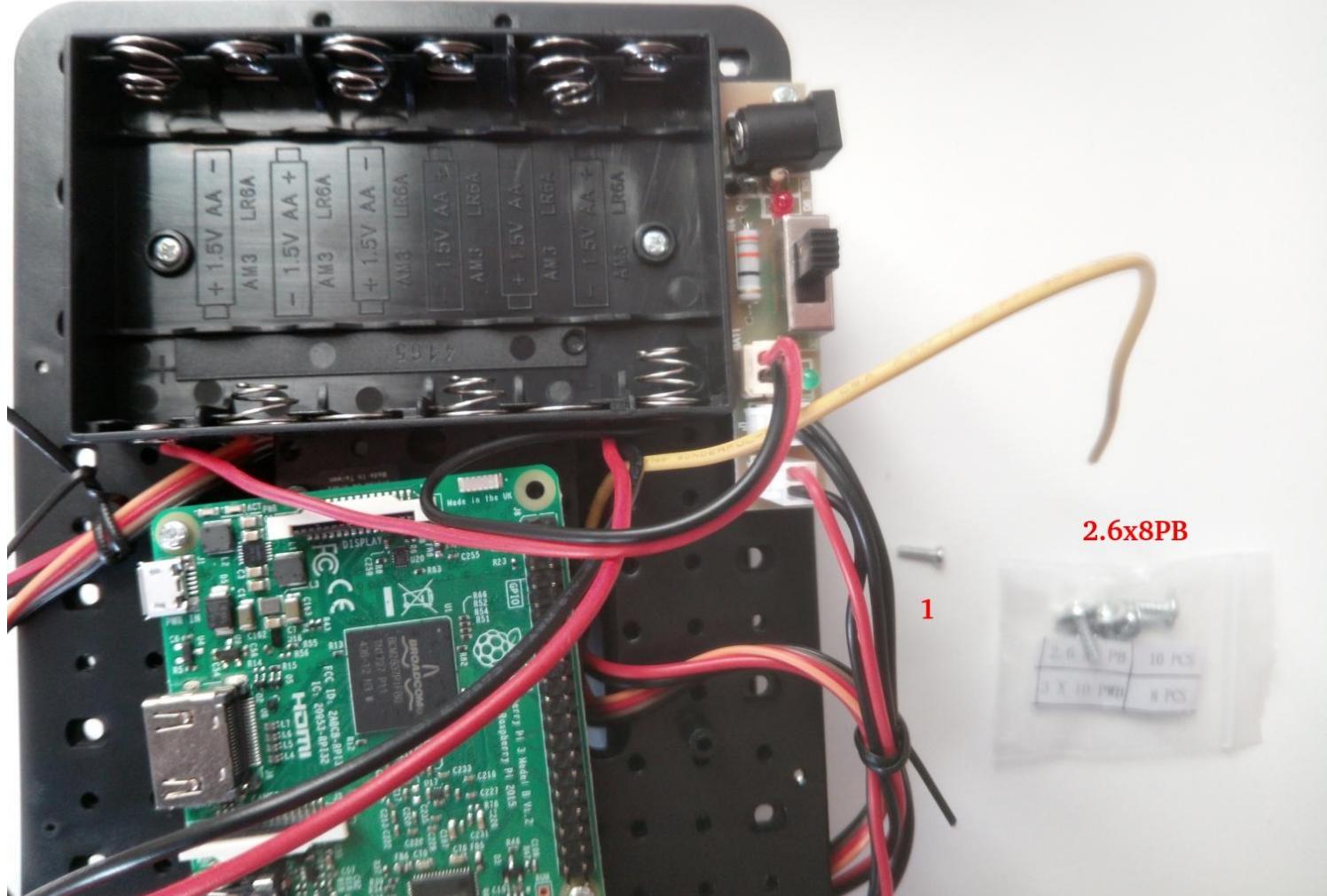


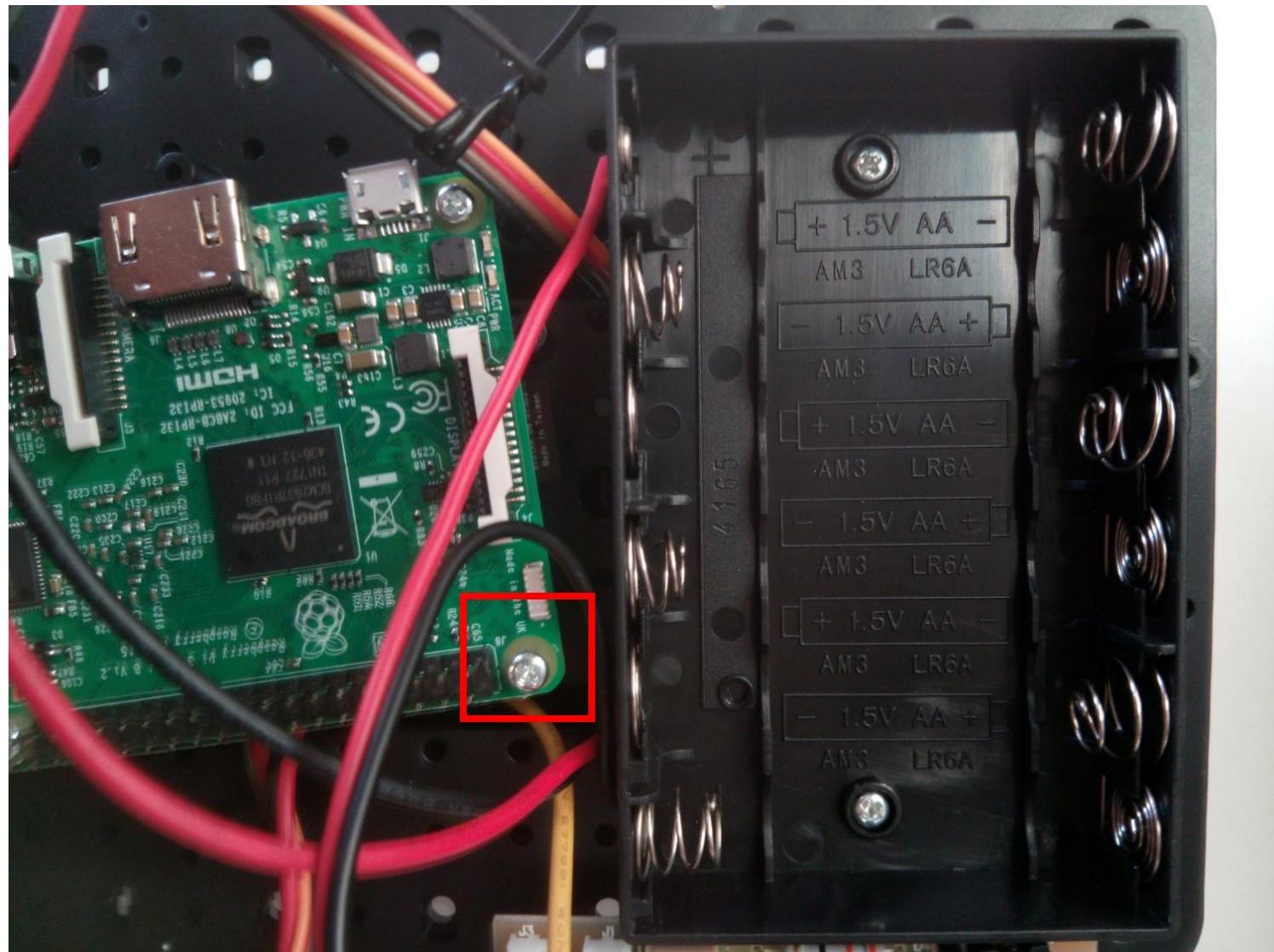
1

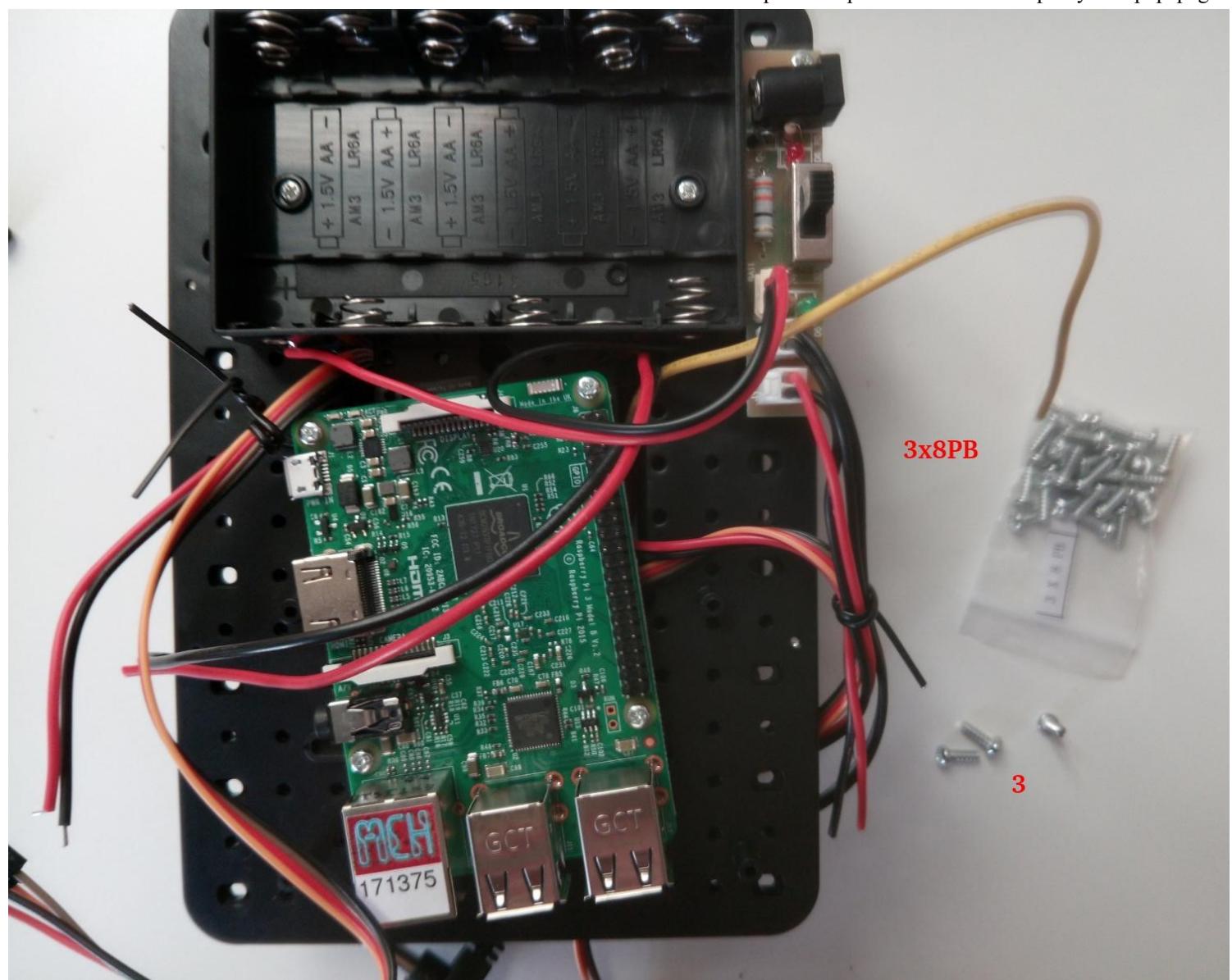
3x8PB

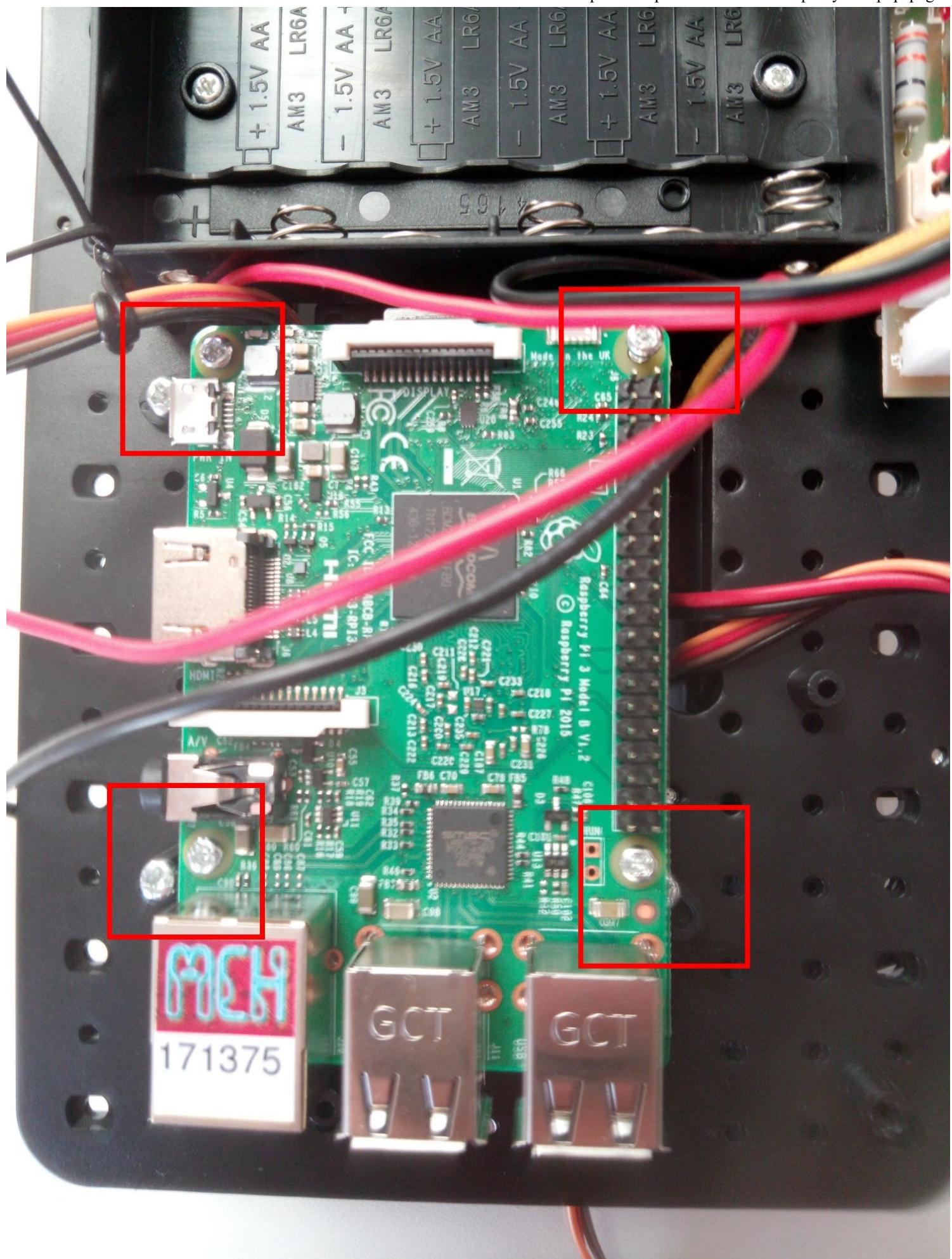


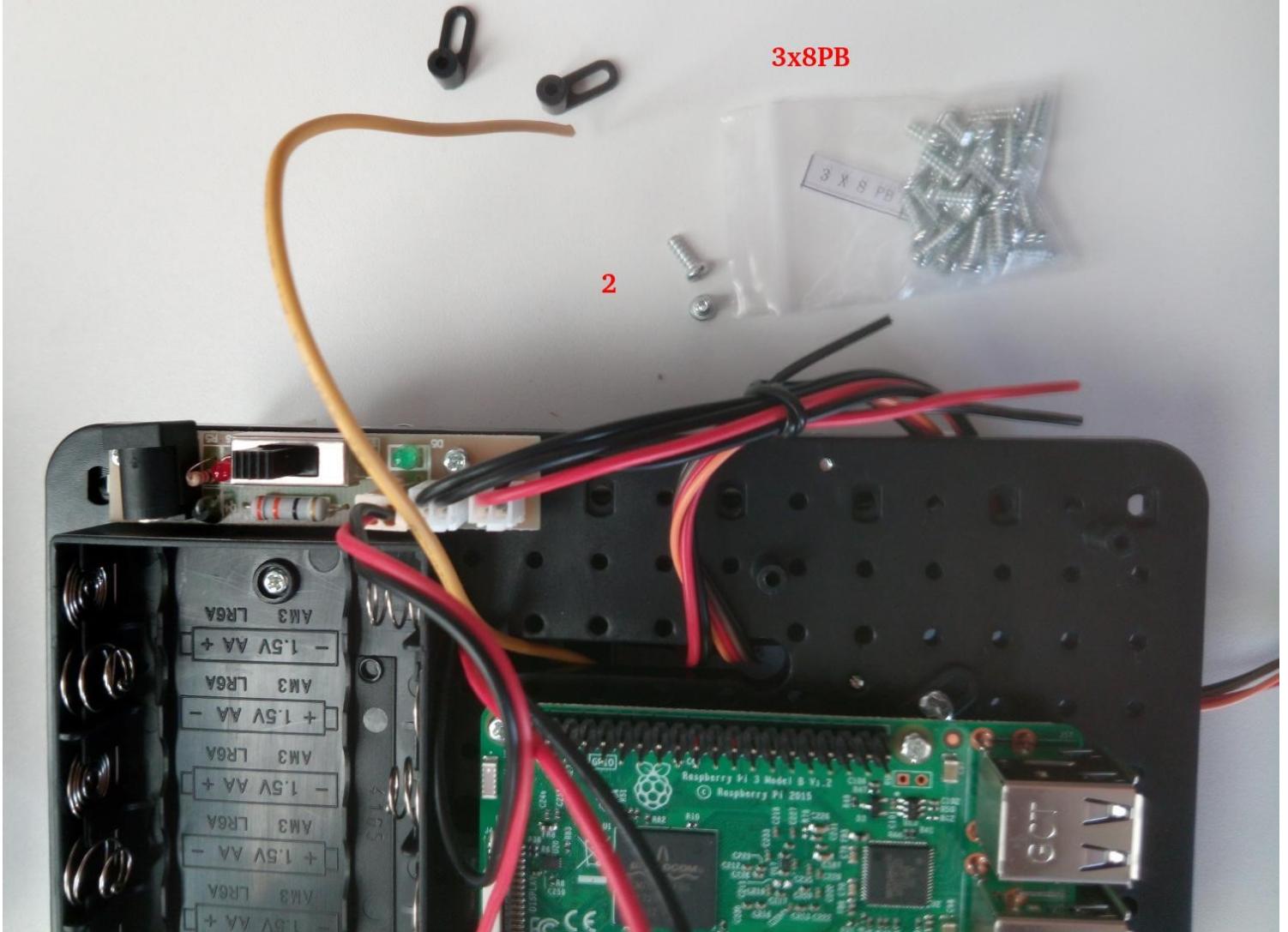


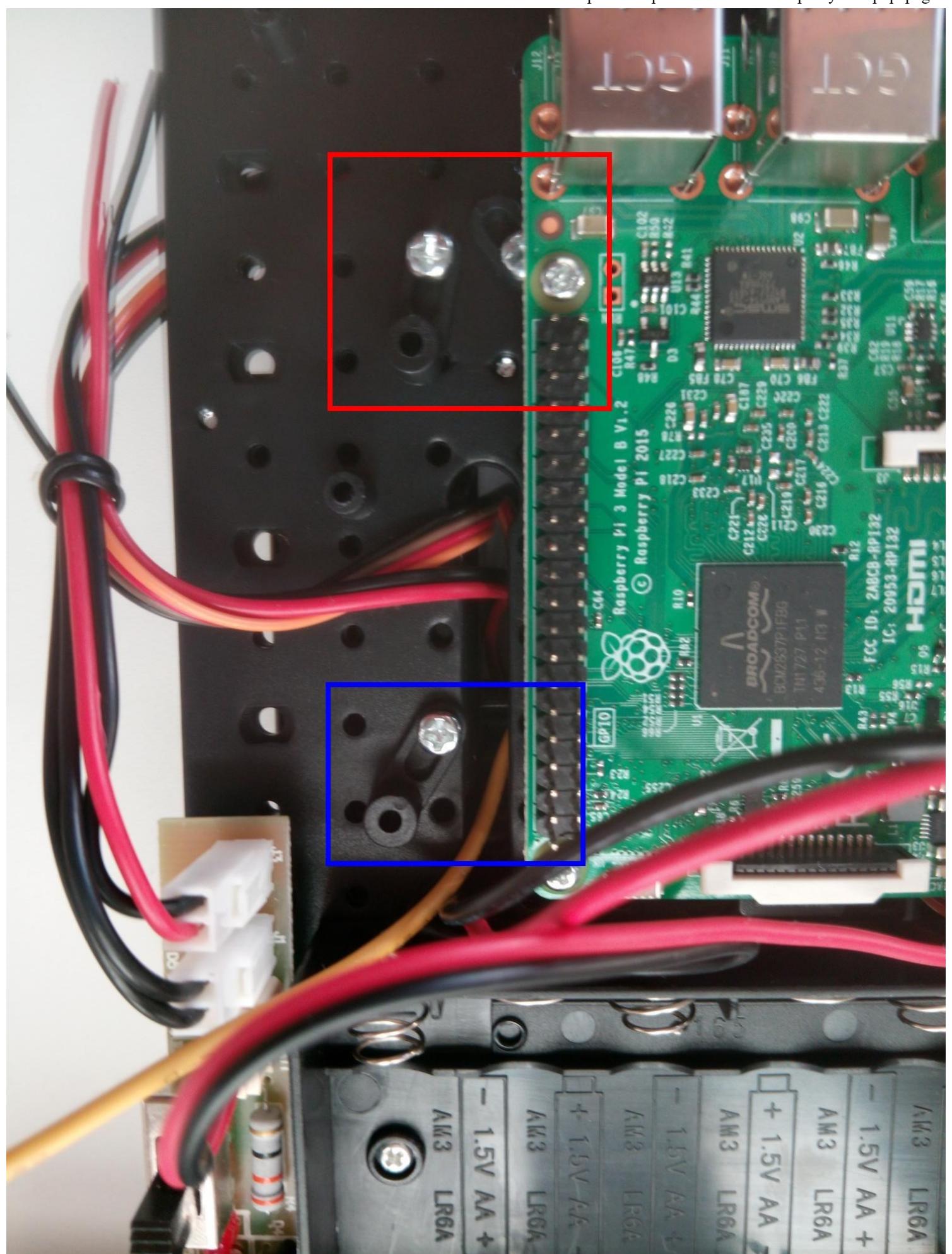


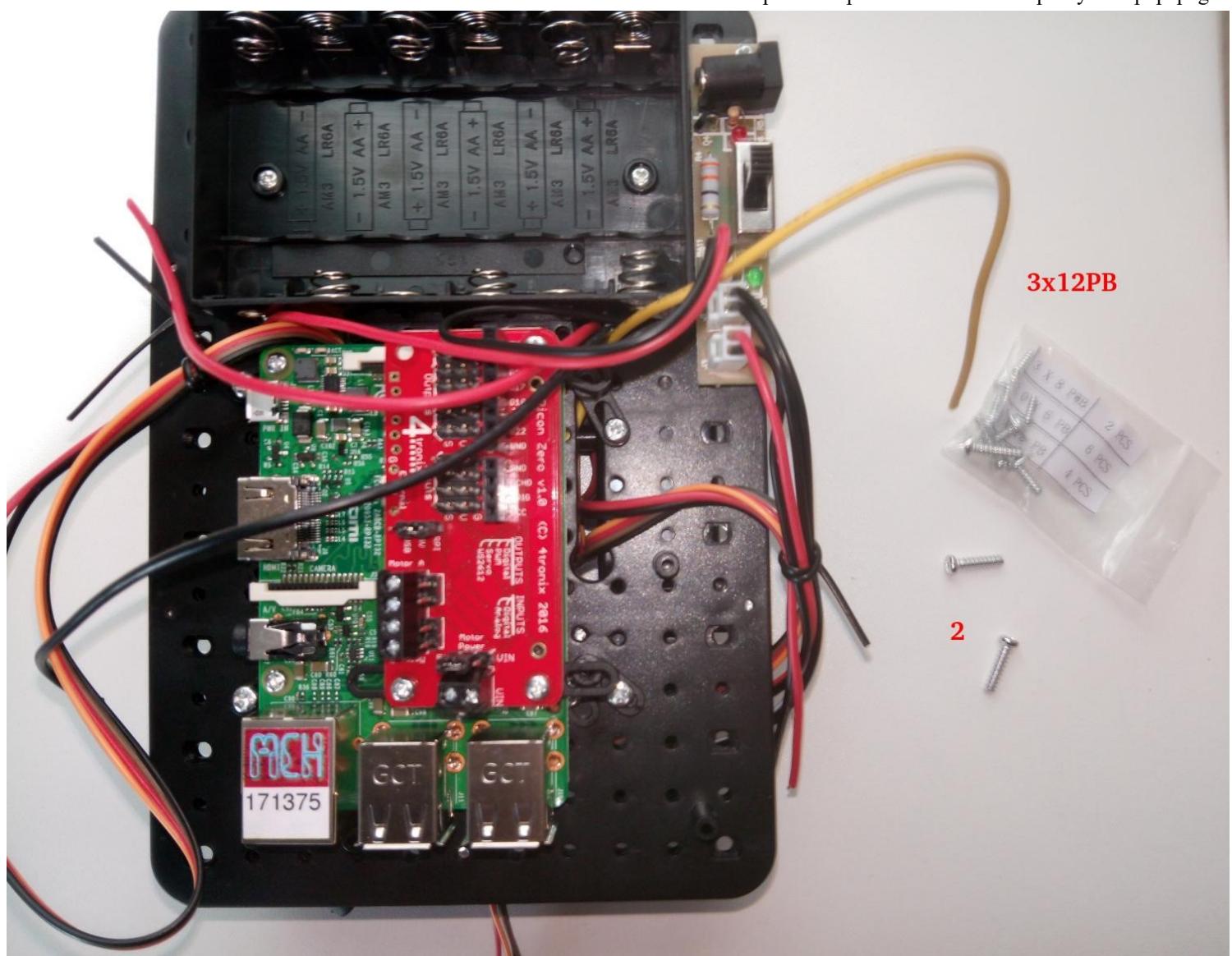


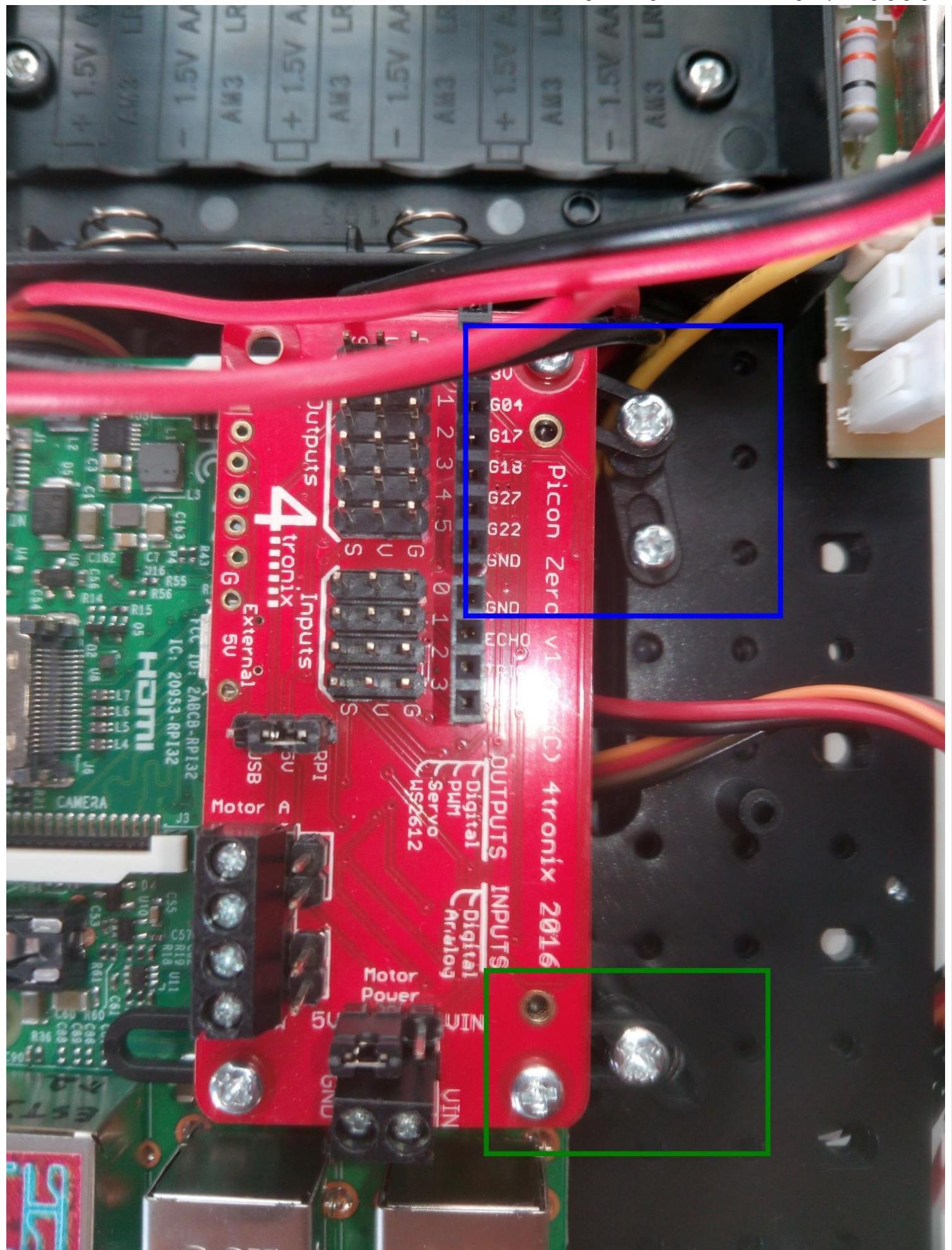


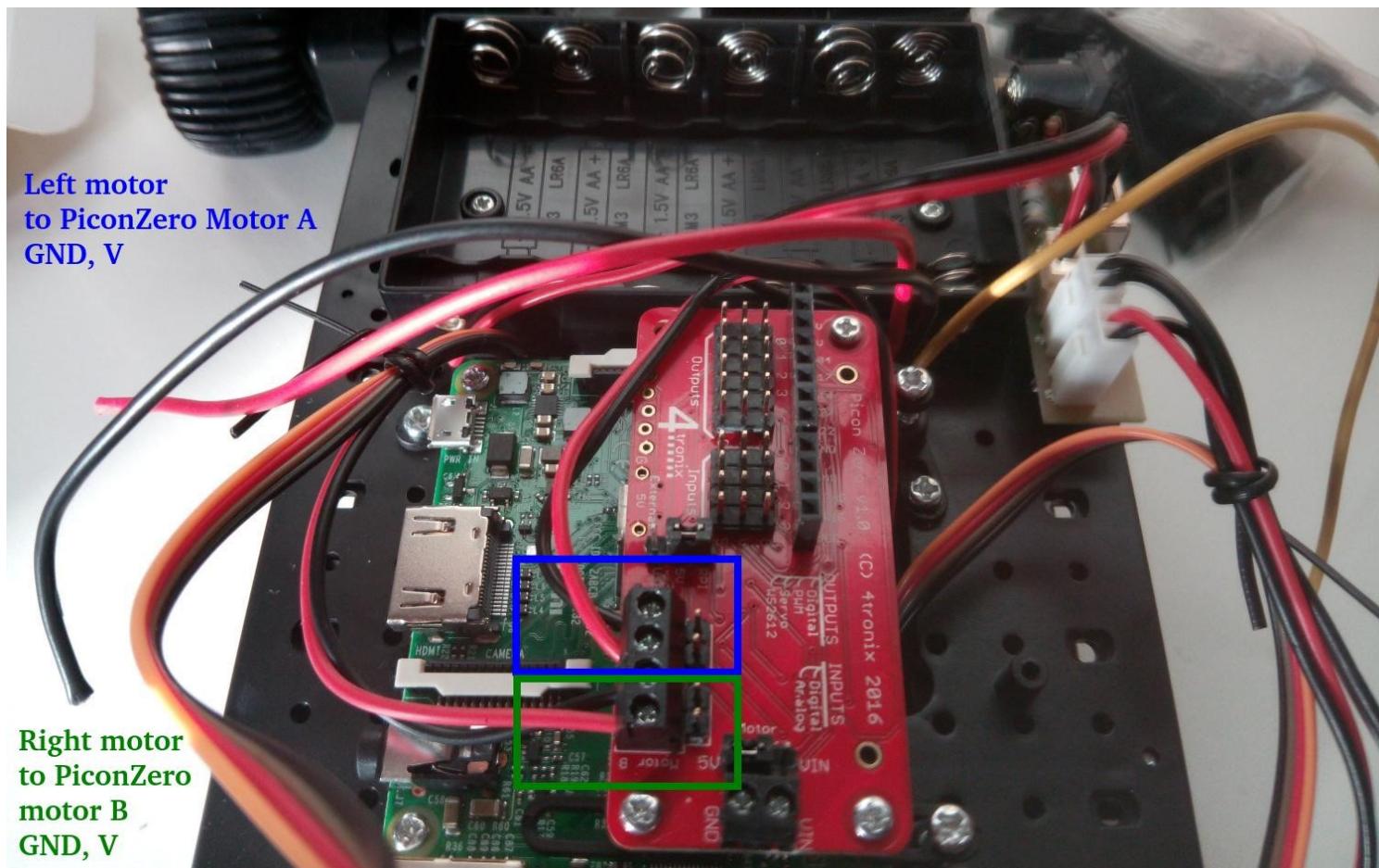
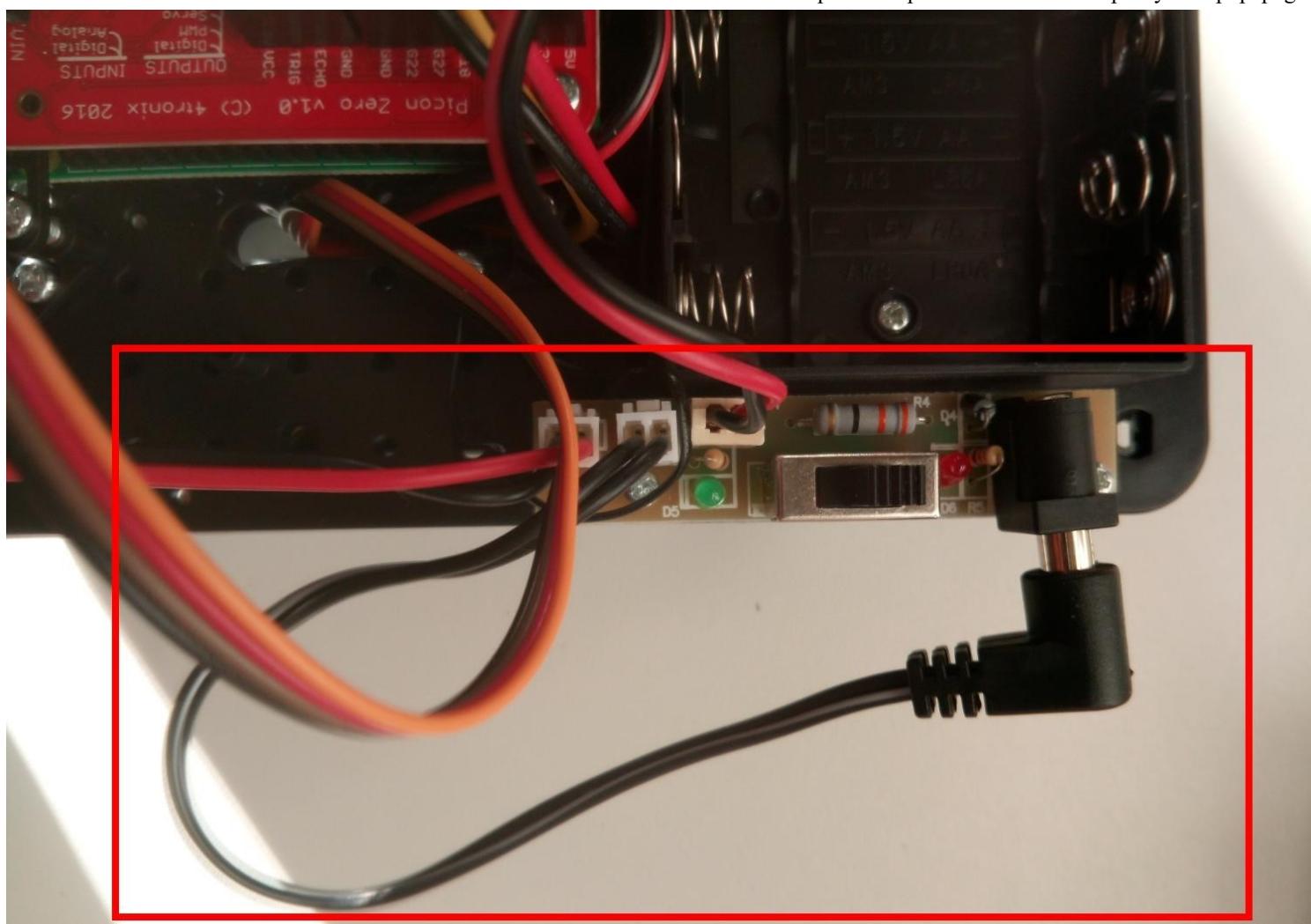


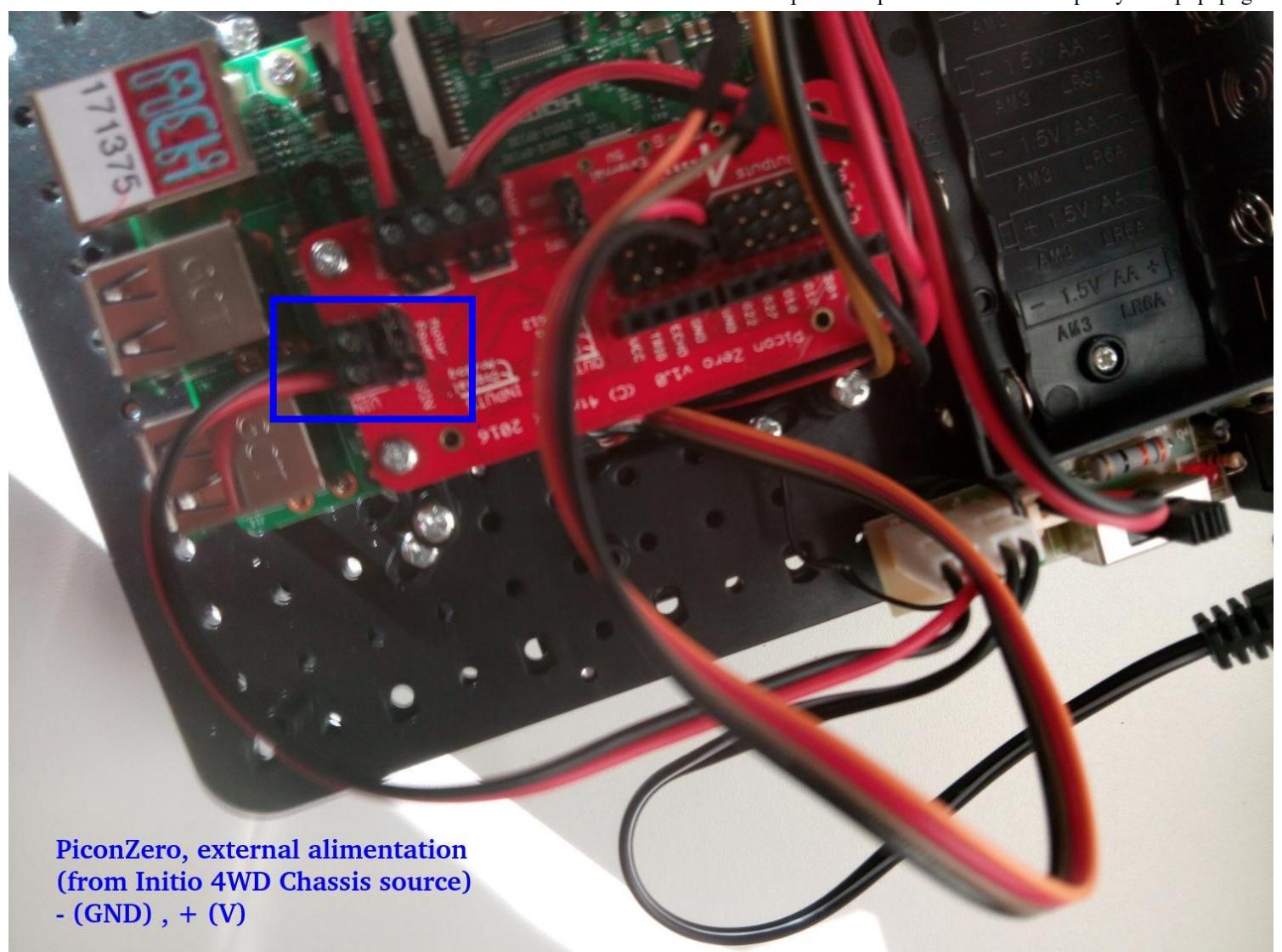


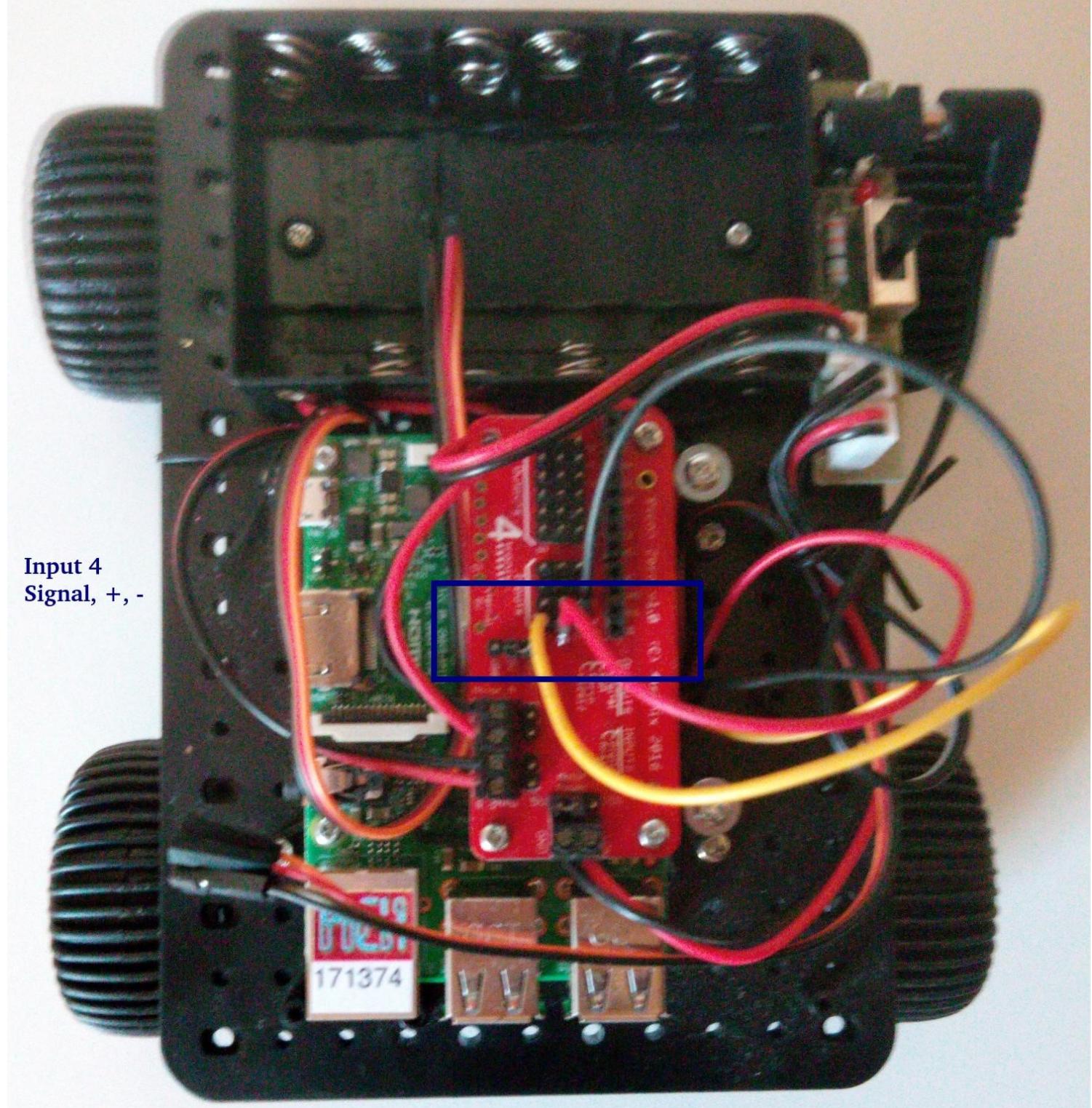


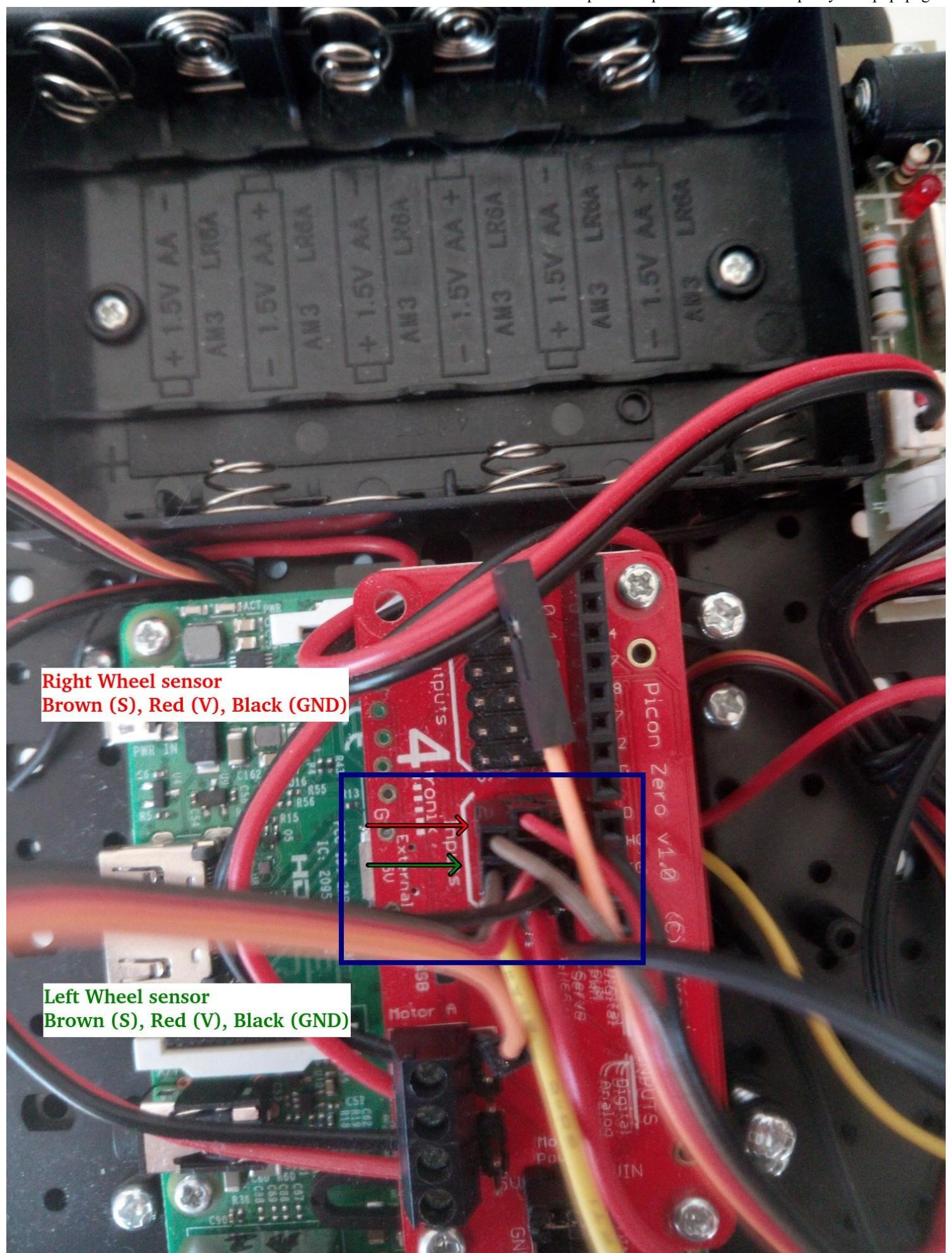






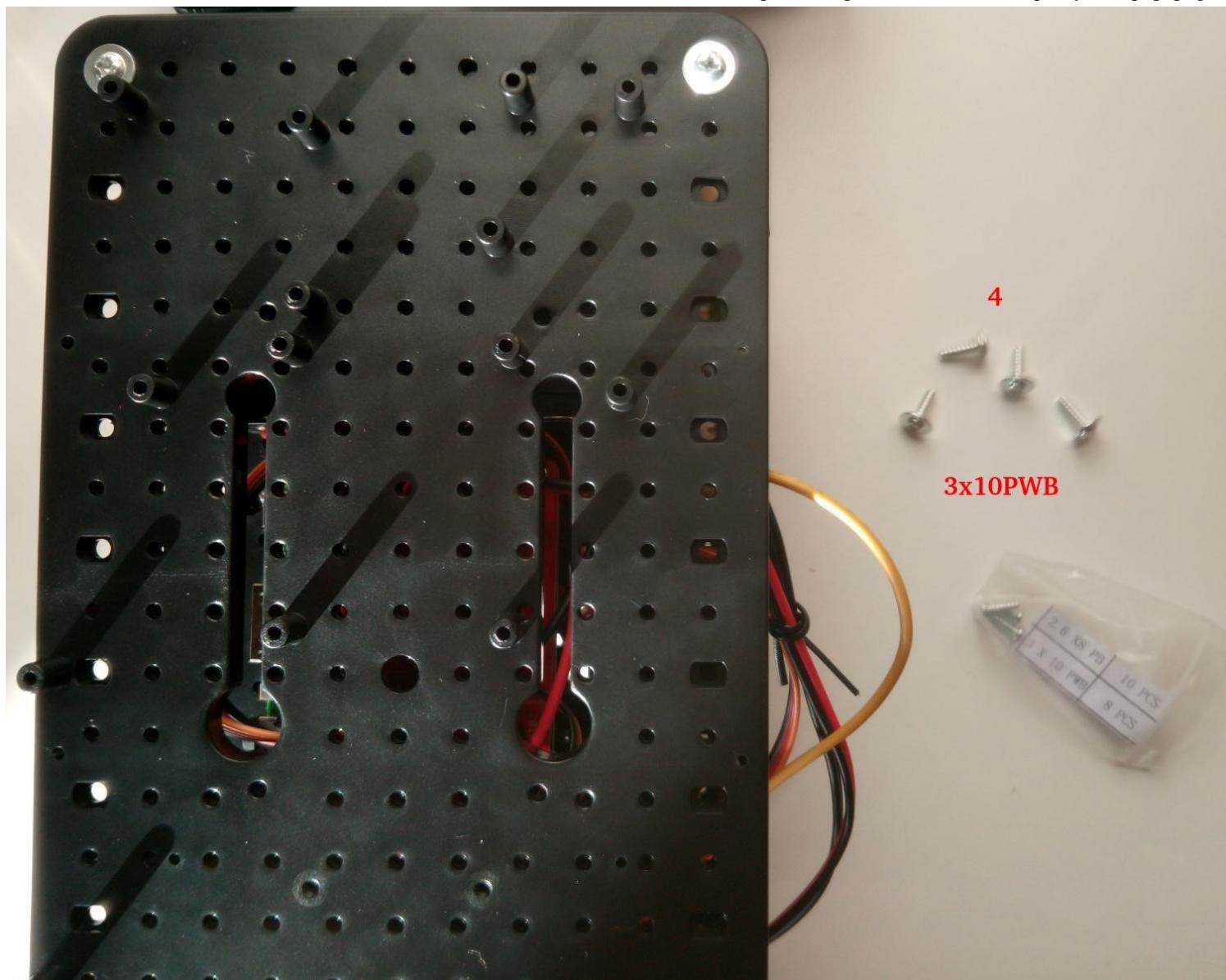




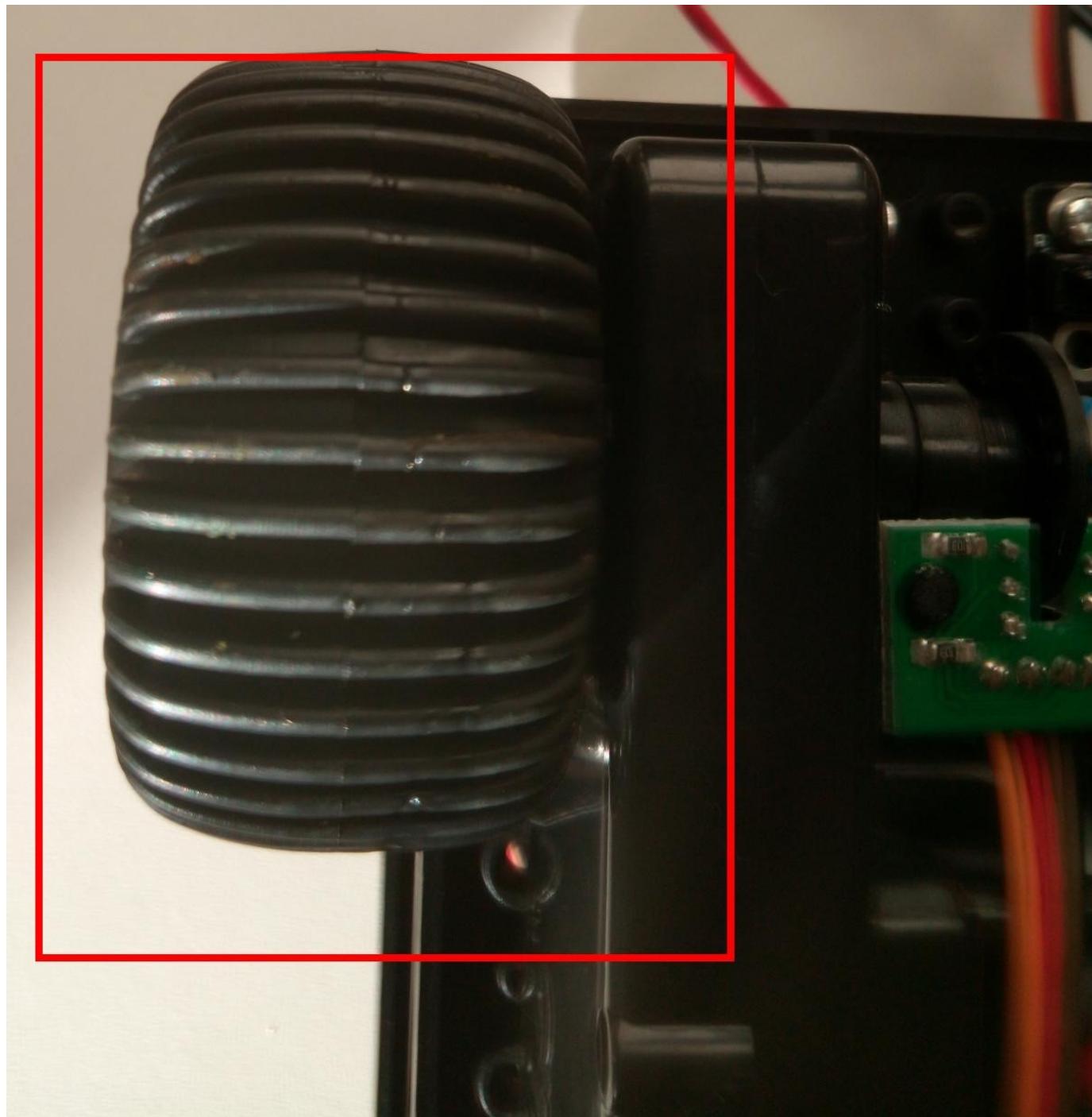


**Right Wheel sensor**  
Brown (S), Red (V), Black (GND)

**Left Wheel sensor**  
Brown (S), Red (V), Black (GND)







## Logicielle

- Insérer la carte micro SD dans le PC, la monter et créer une fichier à la racine nommé ssh :

```
touch ssh
```

Ensuite connecter en Ethernet, un clavier USB, une souris USB et installer Raspbian. Après redémarrage un message d'alerte sécurité doit être affiché pour dire que le mot de passe n'a pas été changé et que le ssh est actif. Si ce n'est pas le cas, ouvrir une console et exécuter :

```
sudo raspi-config
```

Choisissez dans l'interface la liste des services et activer le ssh.

- Exécuter

```
sudo raspi-config
```

Aller dans l'interface pour changer le mode Bureau/console et ne choisir que la console (enable boot to desktop/scratch => text mode).

- Redémarrer

```
sudo reboot
```

- Se connecter en ssh sur le raspberry à l'aide du login **pi** et du mot de passe **raspberry**.
- Modifier le mot de passe de l'utilisateur pi

```
passwd pi
```

- Obtenir les priviléges administrateurs :

```
sudo -s
```

- Créer le compte étudiant

```
adduser student
```

et attribuer le mot de passe **HEH-Powa**

- Restreindre l'accès X11

### Ajouter un groupe

```
groupadd X11Login
```

Ajouter l'utilisateur pi dans le groupe X11Login

```
usermod -a -G X11Login pi
```

Créer un fichier de règle pour l'environnement lightdm

```
echo "X11Login" > /usr/local/etc/lightdm.allow
```

Ensuite éditez le fichier de configuration de lightdm

```
nano /etc/lightdm/lightdm.conf
```

et modifier dans section [seat:\*] les valeurs :

- greeter-hide-users=true
- greeter-allow-guest=false
- allow-guest=false
- autologin-guest=false
- commenter #autologin-user=pi

Il faut maintenant éditer le fichier du module d'authentification pam :

```
nano /etc/pam.d/lightdm
```

Et ajouter la ligne

```
auth required pam_listfile.so onerr=fail item=group sense=allow file=/usr/local/etc/lightdm.allow
```

avant

```
@include common-account
```

- 1.Modifier le nom d'hôte (ex avec "ESTISIM-Rpi-8"):

```
hostname "ESTISIM-Rpi-8"; hostname > /etc/hostname
```

Ajouter l'adresse du réseau dans le fichier hosts

```
echo "127.0.1.1 `hostname`" >> /etc/hosts
```

Dans ce même fichier (/etc/hosts) supprimer la ligne

```
127.0.1.1 raspberry
```

- Mettre à jour :

```
apt-get update
```

```
apt-get upgrade
```

Et s'assurer que les bibliothèques python de base soient disponibles :

```
apt-get install python-smbus python3-smbus python-dev python3-dev
```

- Désactiver le forwarding SSH, en particulier X11

```
echo "" >> "/etc/ssh/sshd_config"
```

```
echo "# Allow group to use X11" >> "/etc/ssh/sshd_config"
```

```
echo "Match Group X11Login" >> "/etc/ssh/sshd_config"
```

```
echo " X11Forwarding yes" >> "/etc/ssh/sshd_config"
```

```
echo " AllowTcpForwarding yes" >> "/etc/ssh/sshd_config"
```

Ouvrir le fichier

```
nano /etc/ssh/sshd_config
```

et modifier les valeurs suivantes :

- X11Forwarding no

- AllowTcpForwarding no
  - Préparer l'installation du PiconZero

### Télécharger l'installateur

```
wget http://4tronix.co.uk/piconz.sh -O piconz.sh
```

Ajouter les droits d'utilisation du bus I2C à l'utilisateur student (pi l'a déjà)

```
usermod -a -G i2c student
```

Ajouter la directive du shell à exécuter

```
sed -i '1i #!/bin/bash'
```

```
mv piconz.sh /usr/local/bin/
```

```
chmod +x /usr/local/bin/piconz.sh
```

```
echo "dtparam=i2c1=on" >> /boot/config.txt
```

```
echo "dtparam=i2c_arm=on" >> /boot/config.txt
```

Ajouter la bibliothèque personnelle

```
wget http://heh.belegkarnil.be/Python/HEHBot.py
```

```
mkdir -p `` python -m site
```

```
grep "/usr/local" "s/[,]*$\" sed "\s/^$["`"
```

```
mkdir -p `` python3 -m site
```

```
grep "/usr/local" "s/[,]*$\" sed "\s/^$["`"
```

```
cp *.py `` python -m site
```

```
grep "/usr/local" "s/[,]*$\" sed "\s/^$["`"
```

```
cp *.py `` python3 -m site
```

```
grep "/usr/local" "s/[,]*$\" sed "\s/^$["`"
```

```
rm *.py
```

- Autoriser l'utilisateur student à éteindre et rallumer le Raspberry

```
echo "student `hostname` =NOPASSWD: /bin/systemctl poweroff,/bin/systemctl halt,/bin/systemctl reboot" >> /etc/sudoers
```

```
echo 'alias reboot="sudo systemctl reboot"' >> /home/student/.bash_aliases
```

```
echo 'alias poweroff="sudo systemctl poweroff"' >> /home/student/.bash_aliases
```

```
echo 'alias halt="sudo systemctl halt"' >> /home/student/.bash_aliases
```

```
chown student:student /home/student/.bash_aliases
```

- Éditer le fichier de configuration nano afin qu'il soit plus convivial.

```
nano /etc/nanorc
```

Et dé-commenter et attribuer les valeurs suivantes :

- set linenumber
- tabsize 3
  - Créer un point d'accès WiFi

```
sudo apt-get install dnsmasq hostapd
```

```
sudo systemctl stop dnsmasq
```

```
sudo systemctl stop hostapd
```

Modifier le fichier de configuration de hostapd

```
sudo nano /etc/default/hostapd
```

en dé-commentant et en attribuant la valeur suivante

```
DAEMON_CONF="/etc/hostapd/hostapd.conf"
```

Désactiver association serveur DHCP à l'interface réseau ethernet

```
echo "denyinterfaces wlan0" >> /etc/dhcpd.conf
```

Créer une configuration spécifique pour l'interface WiFi

```
echo "allow-hotplug wlan0" > /etc/network/interfaces.d/wlan0
```

```
echo "iface wlan0 inet static" >> /etc/network/interfaces.d/wlan0
```

```
echo "address 192.168.2.1" >> /etc/network/interfaces.d/wlan0
```

```
echo "netmask 255.255.255.0" >> /etc/network/interfaces.d/wlan0
```

```
echo "network 192.168.2.0" >> /etc/network/interfaces.d/wlan0
```

```
echo "broadcast 192.168.2.255" >> /etc/network/interfaces.d/wlan0
```

Configurer le point d'accès WiFi avec le channel en reprenant l'exemple de l'hôte "ESTISIM-Rpi-8" associé à un channel 8 et un ESSID équivalent à l'identifiant du Raspberry :

```
echo "interface=wlan0" > /etc/hostapd/hostapd.conf
```

```
echo "driver=nl80211" >> /etc/hostapd/hostapd.conf
```

```
echo "hw_mode=g" >> /etc/hostapd/hostapd.conf
```

```
perl -e 'print "channel=$1" if <STDIN> =~ /-(\d+)$/;' < /etc/hostname >> /etc/hostapd/hostapd.conf
```

```
echo "ieee80211n=1" >> /etc/hostapd/hostapd.conf
```

```
echo "" >> /etc/hostapd/hostapd.conf
```

```
echo "wmm_enabled=1" >> /etc/hostapd/hostapd.conf
```

```
echo "ht_capab=[HT40][SHORT-GI-20][DSSS_CCK-40]" >> /etc/hostapd/hostapd.conf
```

```
echo "macaddr_acl=0" >> /etc/hostapd/hostapd.conf
```

```
echo "ignore_broadcast_ssid=0" >> /etc/hostapd/hostapd.conf
```

```
echo "" >> /etc/hostapd/hostapd.conf
```

```
echo "# Use WPA2" >> /etc/hostapd/hostapd.conf
```

```
echo "auth_algs=1" >> /etc/hostapd/hostapd.conf
```

```
echo "wpa=2" >> /etc/hostapd/hostapd.conf
```

```
echo "wpa_key_mgmt=WPA-PSK" >> /etc/hostapd/hostapd.conf
```

```
echo "rsn_pairwise=CCMP" >> /etc/hostapd/hostapd.conf
```

```
echo "" >> /etc/hostapd/hostapd.conf
```

```
echo "# Change these to your choice" >> /etc/hostapd/hostapd.conf
```

```
echo "# This is the name of the network" >> /etc/hostapd/hostapd.conf
```

```
perl -e 'print "ssid=ESTISIM-Rpi-$1" if <STDIN> =~ /-(\d+)$/;' < /etc/hostname >> /etc/hostapd/hostapd.conf
```

```
echo "# The network passphrase" >> /etc/hostapd/hostapd.conf
```

```
echo "wpa_passphrase=Raspberry@HEH" >> /etc/hostapd/hostapd.conf
```

- Configurer et associer le service DHCP à l'interface WiFi :

```
mv /etc/dnsmasq.conf /etc/dnsmasq.conf.bck
```

```
echo "interface=wlan0 # Use interface wlan0" > /etc/dnsmasq.conf
```

```
echo "listen-address=192.168.2.1 # Specify the address to listen on" >> /etc/dnsmasq.conf
```

```
echo "bind-interfaces # Bind to the interface" >> /etc/dnsmasq.conf
```

```
echo "server=8.8.8.8 # Use Google DNS" >> /etc/dnsmasq.conf
```

```
echo "domain-needed # Don't forward short names" >> /etc/dnsmasq.conf
```

```
mv /etc/dnsmasq.conf /etc/dnsmasq.conf.bck
```

```
echo " bogus-priv # Drop the non-routed address spaces." >> /etc/dnsmasq.conf
```

```
echo " dhcp-range=192.168.2.2,192.168.2.20,255.255.255.0,24h" >> /etc/dnsmasq.conf
```

- Redémarrer

```
reboot
```

- Se connecter au nouveau point d'accès WiFi avec le mot de passe configuré (Raspberry@HEH). Se connecter en ssh avec le compte étudiant :

```
ssh student@192.168.2.1
```

Installer la bibliothèque du PiconZero

```
piconz.sh
```