

# Communiquer avec Poppy

## Table des matières

Communication filaire	2
Etablir la communication avec le serveur Web interne	2
Premiers tests	4
Monitor and control	5
Snap ! build your Own blocks	5
Jupiter Python terminal	5
Communication en ssh	6
Configuration du wifi	7
Vérifications préalables	7
Configuration	8

## Communication filaire

### Etablir la communication avec le serveur Web interne

Le robot peut être directement branché avec un câble. A sa mise sous tension il reçoit une adresse du serveur DHCP disponible sur le réseau, par

exemple :

poppy	192.168.1.110	00:1e:06:30:58:9c
-------	---------------	-------------------

Cette adresse répond à la commande ping.

```
C:\Users\Luc>ping 192.168.1.110

Envoi d'une requête 'Ping' 192.168.1.110 avec 32 octets de données :
Réponse de 192.168.1.110 : octets=32 temps<1ms TTL=64

Statistiques Ping pour 192.168.1.110:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
    Durée approximative des boucles en millisecondes :
        Minimum = 0ms, Maximum = 0ms, Moyenne = 0ms

C:\Users\Luc>
```

Pendant le nom **poppy** n'est pas résolu par le DNS :

```
C:\Users\Luc>ping poppy
La requête Ping n'a pas pu trouver l'hôte poppy. Vérifiez le nom et essayez à nouveau.
```

C'est le nom **poppy.local** qui est résolu

```
C:\Users\Luc>ping poppy.local

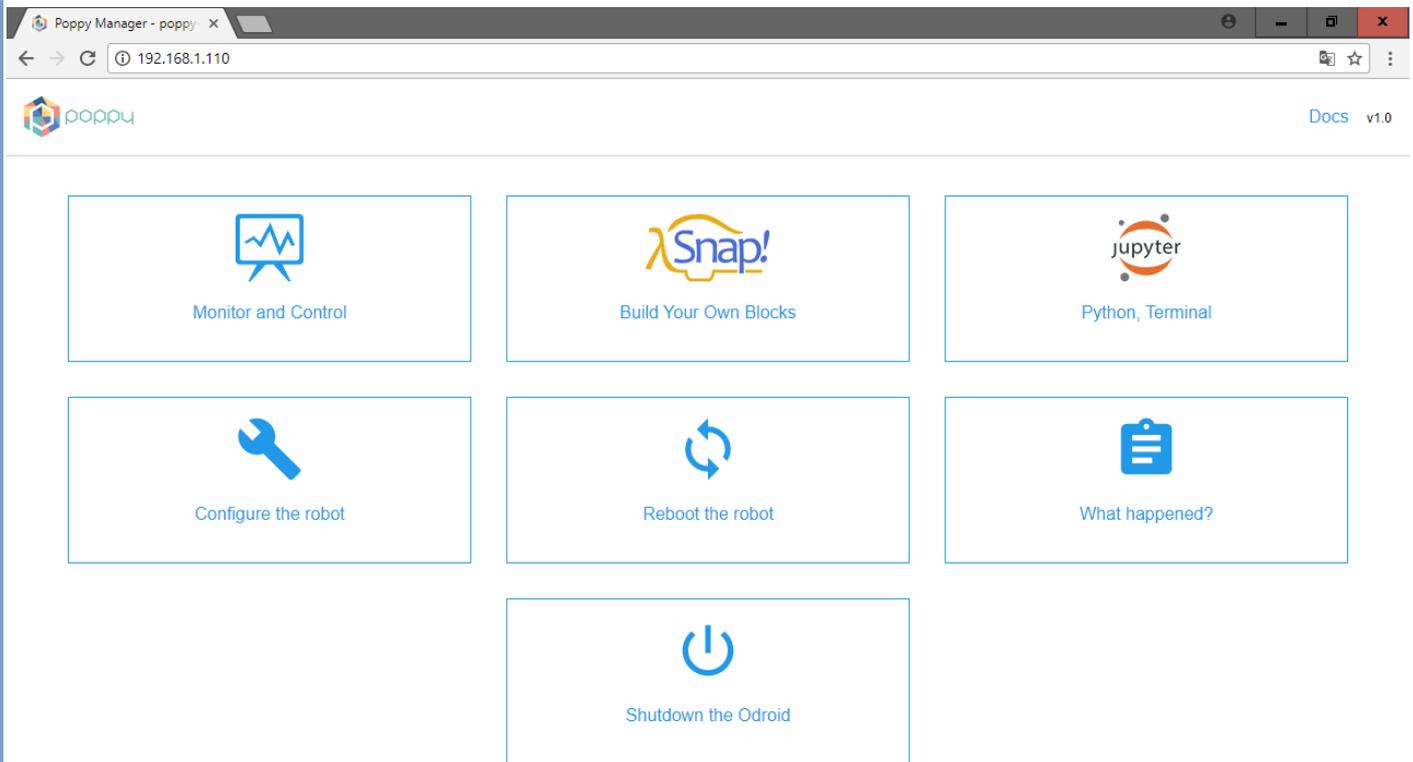
Envoi d'une requête 'ping' sur poppy.local [fe80::21e:6ff:fe30:589c%2] avec 32 octets de données :
Réponse de fe80::21e:6ff:fe30:589c%2 : temps<1ms

Statistiques Ping pour fe80::21e:6ff:fe30:589c%2:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
    Durée approximative des boucles en millisecondes :
        Minimum = 0ms, Maximum = 0ms, Moyenne = 0ms

C:\Users\Luc>
```

On peut communiquer avec le robot avec un navigateur (Chrome de préférence) :

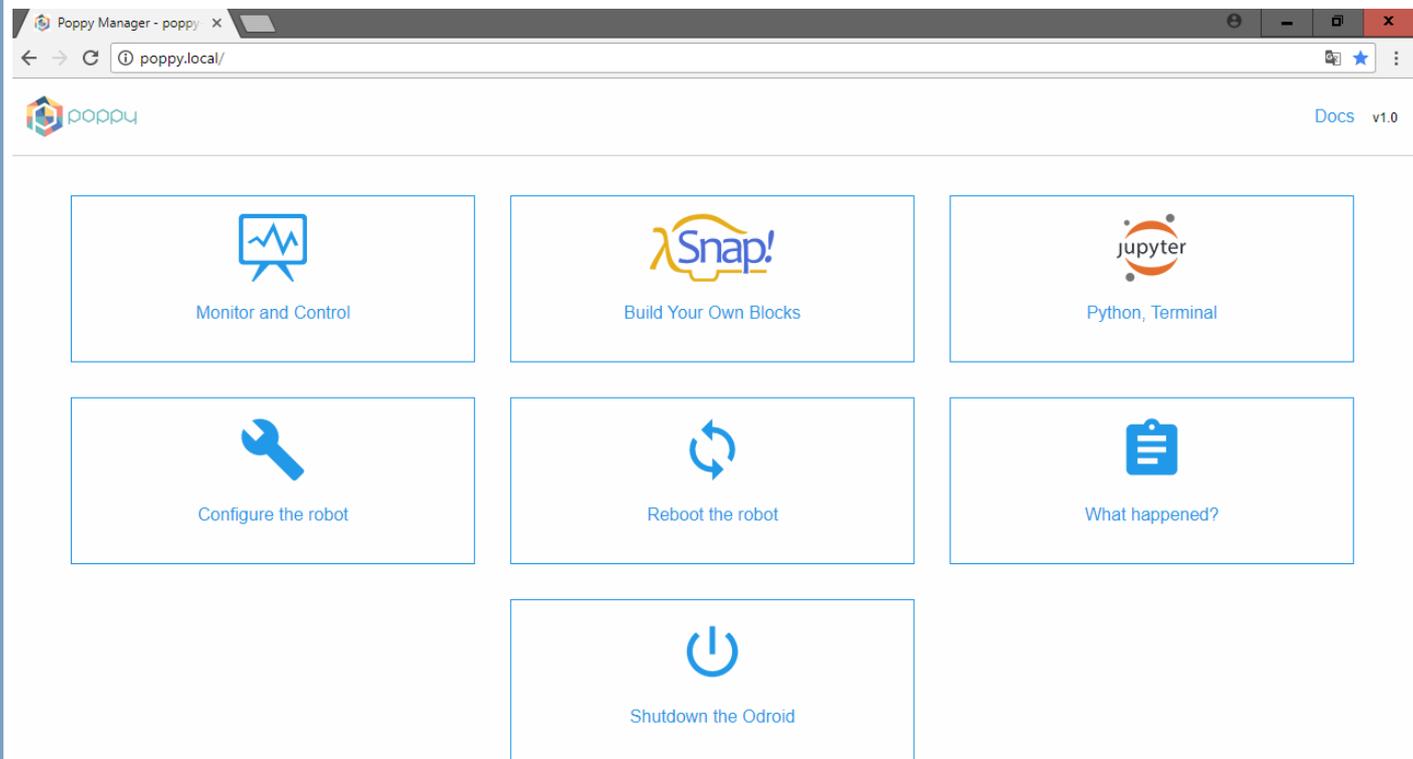
- En indiquant cette adresse.



The screenshot shows a web browser window with the address bar containing "192.168.1.110". The page title is "Poppy Manager - poppy". The interface features a "poppy" logo and a "Docs v1.0" link. The main content area contains seven buttons with icons and labels:

- Monitor and Control (Icon: monitor with waveform)
- Build Your Own Blocks (Icon: Snap! logo)
- Python, Terminal (Icon: Jupyter logo)
- Configure the robot (Icon: wrench)
- Reboot the robot (Icon: circular arrows)
- What happened? (Icon: clipboard)
- Shutdown the Odroid (Icon: power button)

- En indiquant le nom ***poppy.local***



The screenshot shows a web browser window with the address bar containing "poppy.local/". The page title is "Poppy Manager - poppy". The interface features a "poppy" logo and a "Docs v1.0" link. The main content area contains seven buttons with icons and labels:

- Monitor and Control (Icon: monitor with waveform)
- Build Your Own Blocks (Icon: Snap! logo)
- Python, Terminal (Icon: Jupyter logo)
- Configure the robot (Icon: wrench)
- Reboot the robot (Icon: circular arrows)
- What happened? (Icon: clipboard)
- Shutdown the Odroid (Icon: power button)

### Premiers tests

Avant toute chose commencer par cliquer sur l'icône **What happened ?**



What happened?

On observe souvent un problème dans le journal :



Docs v1.0

## Logs

```
Attempt 1 to start the robot...  
[Errno 16] could not open port /dev/ttyACM0: [Errno 16] Device or resource busy: '/dev/ttyACM0'  
Attempt 2 to start the robot...  
[Errno 16] could not open port /dev/ttyACM0: [Errno 16] Device or resource busy: '/dev/ttyACM0'  
Attempt 3 to start the robot...  
[Errno 16] could not open port /dev/ttyACM0: [Errno 16] Device or resource busy: '/dev/ttyACM0'  
Attempt 4 to start the robot...
```

Dans ce cas cliquer sur **poppy** pour revenir à l'écran d'accueil puis cliquer sur l'icône **Reboot the robot ?**



Reboot the robot

Cliquer à nouveau sur l'icône **What happened ?**

Le journal vous indique maintenant un robot opérationnel



Docs v1.0

## Logs

```
Attempt 1 to start the robot...  
SnapRobotServer is now running on: http://0.0.0.0:6969  
  
You can open Snap! interface with loaded blocks at "http://snap.berkeley.edu/snapsource/snap.html#open:http://192.168.1.110:6969/sna  
  
HTTPRobotServer is now running on: http://0.0.0.0:8080  
  
Robot created and running!
```

Vous pouvez maintenant choisir l'un des modes :



Monitor and Control

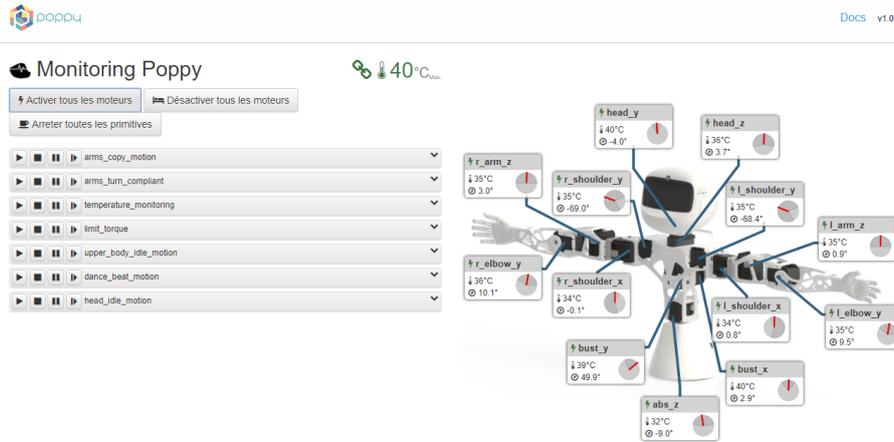


Build Your Own Blocks

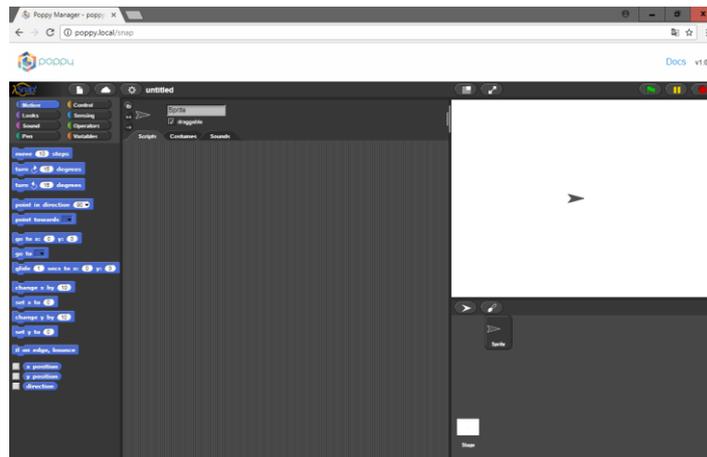


Python, Terminal

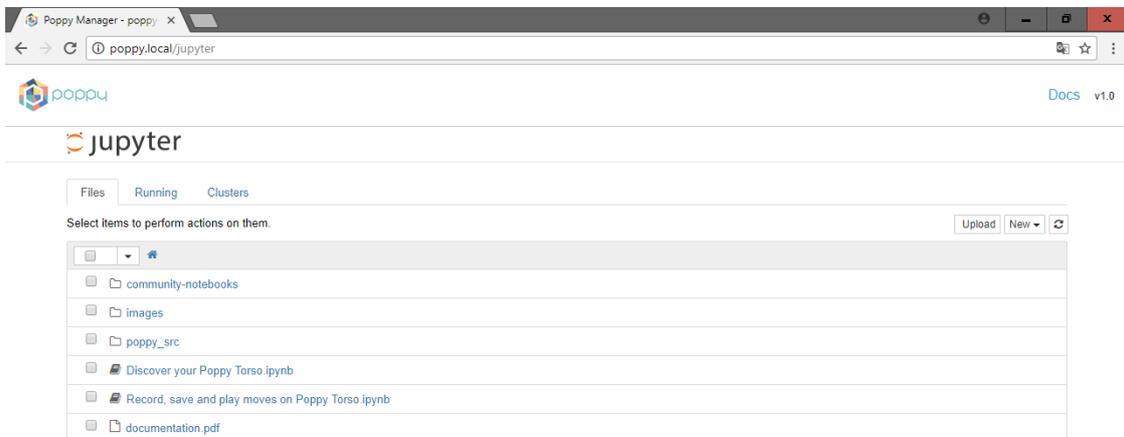
## Monitor and control



## Snap ! build your Own blocks



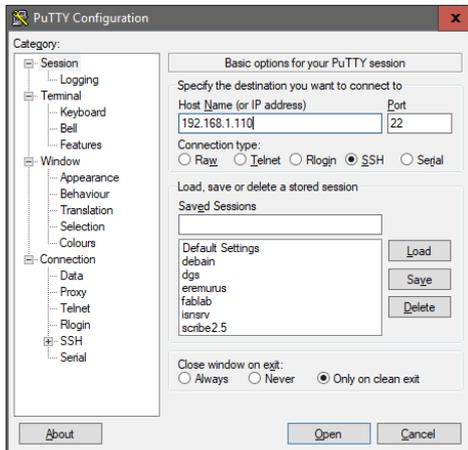
## Jupyter Python terminal



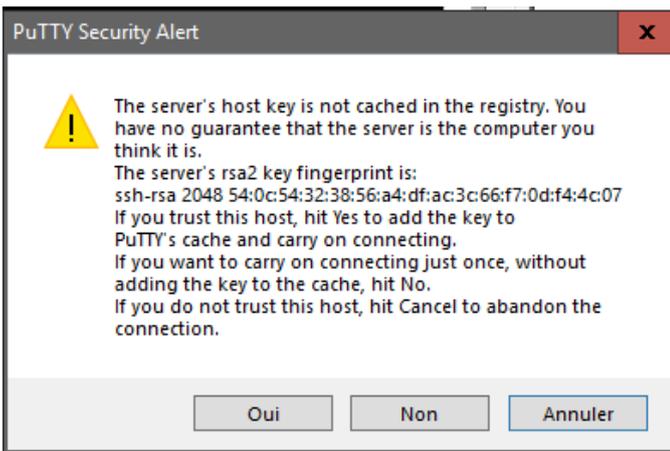
Ici on trouve le fichier **documentation.pdf**

## Communication en ssh

Secure Shell (SSH) est une fonctionnalité de Linux qui vous permet d'ouvrir une session de terminal sur votre Raspberry Pi à partir d'une ligne de commande de votre ordinateur hôte. SSH est activé sur Poppy. Depuis Windows, vous aurez besoin de télécharger un programme gratuit appelé **putty** : <http://www.putty.org/>.

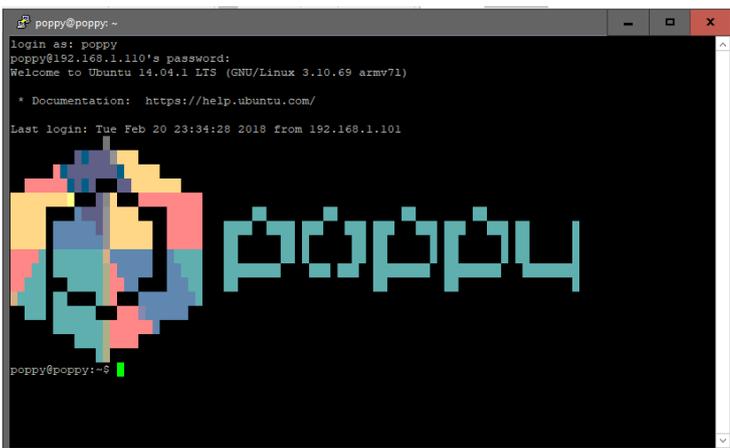


Sélectionner SSH et bien indiquer l'adresse du robot.



Valider le message de sécurité.

Identifiant/mot de passe : poppy/poppy



La connexion à distance est établie. Elle pourra être

fermée par la commande **exit**

## Configuration du wifi

### Vérifications préalables

Insérez l'adaptateur USB Edimax dans le port USB de votre Raspberry Pi. Une fois connecté en SSH,

Utilisez la commande suivante pour vérifier que le Pi "voit" l'adaptateur Edimax WiFi. Une fois connecté en SSH,

```
sudo lsusb
```

```
poppy@poppy:~$ sudo lsusb
Bus 006 Device 002: ID 0bda:8153 Realtek Semiconductor Corp.
Bus 006 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 005 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 004 Device 002: ID 05e3:0616 Genesys Logic, Inc.
Bus 004 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 003 Device 006: ID 16d0:06a7 MCS
Bus 003 Device 005: ID 058f:3822 Alcor Micro Corp.
Bus 003 Device 004: ID 0d8c:000c C-Media Electronics, Inc. Audio Adapter
Bus 003 Device 003: ID 1a40:0101 Terminus Technology Inc. 4-Port HUB
Bus 003 Device 002: ID 05e3:0616 Genesys Logic, Inc. 4-port hub
Bus 003 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 002 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
Bus 001 Device 002: ID 7392:7811 Edimax Technology Co., Ltd EW-7811Un 802.11n Wireless Adapter [Realtek RTL8188CUS]
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
poppy@poppy:~$
```

On peut repérer une ligne Edimax.

Maintenant que nous avons vérifié que l'adaptateur Edimax est reconnu, vérifions que le pilote du noyau est chargé.

Utilisez la commande suivante pour lister les modules du noyau :

```
sudo lsmod
```

```
bluetooth          227652  10 bnep,ricommon
rfkill              12805   3 bluetooth
ioboard_keyled     4962    0
8192cu              531585  0
ina231_sensor      8396    0
binfmt_misc        6671    1
fuse                74794   2
```

Vous devriez voir le module du noyau "8192cu" chargé. Si ce n'est pas le cas, essayez de retirer et de réinsérer l'adaptateur USB Edimax.

Comme vérification finale de validation pour vous assurer que l'Edimax est prêt à l'emploi, afficher la liste de la configuration actuelle du réseau sans fil. Utiliser la commande :

```
sudo iwconfig
```

```
poppy@poppy:~$ sudo iwconfig
sit0      no wireless extensions.

wlan2     unassociated  Nickname:"<WIFI@REALTEK>"
          Mode:Managed  Frequency=2.412 GHz  Access Point: Not-Associated

          Sensitivity:0/0
          Retry:off   RTS thr:off   Fragment thr:off
          Encryption key:off
          Power Management:off
          Link Quality=0/100  Signal level=0 dBm  Noise level=0 dBm
          Rx invalid nwid:0  Rx invalid crypt:0  Rx invalid frag:0
          Tx excessive retries:0  Invalid misc:0  Missed beacon:0

lo        no wireless extensions.

ip6tnl0   no wireless extensions.

eth3      no wireless extensions.
```

Nous cherchons simplement à nous assurer qu'un adaptateur (ici "wlan2") est présent. Nous sommes maintenant prêts à passer aux étapes de configuration.

### Configuration

Si vous avez installé l'adaptateur, vérifié que le pilote du noyau est chargé et confirmé qu'une interface réseau sans fil ("wlan2") est disponible, vous devez configurer les paramètres de connexion sans fil pour vous connecter en toute sécurité à votre réseau sans fil. Tout d'abord, ouvrons le fichier de configuration des interfaces réseau pour l'éditer, en utilisant la commande suivante :

```
sudo nano /etc/network/interfaces
```

```
poppy@poppy: ~
GNU nano 2.2.6 File: /etc/network/interfaces
interfaces(5) file used by ifup(8) and ifdown(8)
# Include files from /etc/network/interfaces.d:
source-directory /etc/network/interfaces.d

#auto wlan0
#iface wlan0 inet dhcp

#wpa-ssid "turtle"
#wpa-psk "radacina"
```

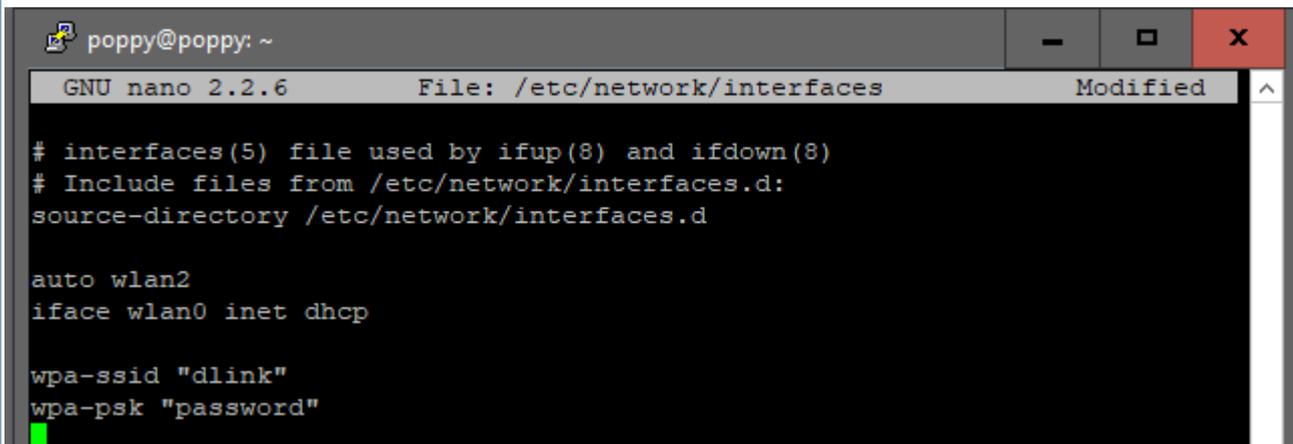
Assurez-vous que les lignes suivantes sont ajoutées à votre fichier (ou non commentées)

```
auto wlan2
iface wlan2 inet dhcp

wpa-ssid "dlink"
wpa-psk "password"
```

dlink est le nom SSID du réseau

password est le mot de passe en clair de la borne.



```
poppy@poppy: ~
GNU nano 2.2.6 File: /etc/network/interfaces Modified
# interfaces(5) file used by ifup(8) and ifdown(8)
# Include files from /etc/network/interfaces.d:
source-directory /etc/network/interfaces.d

auto wlan2
iface wlan0 inet dhcp

wpa-ssid "dlink"
wpa-psk "password"
```

Sauver le fichier par Ctrl+X puis Y puis entrée.

Une fois les fichiers de configuration terminés, utilisez la commande suivante pour redémarrer l'adaptateur / l'interface sans fil.

```
sudo ifup wlan0
```

Cette commande redémarre l'interface réseau et utilise les paramètres d'interface sans fil nouvellement définis pour établir une connexion réseau. En supposant que la configuration est correcte, l'interface (wlan2) doit se connecter et acquérir une adresse IP.

Si vous avez des problèmes en utilisant la commande "ifup", vérifiez vos paramètres de connexion sans fil et redémarrez votre Raspberry Pi.

Pour la vérification de la connexion et un autre moyen de voir votre adresse IP assignée, utilisez la commande suivante pour lister la configuration réseau :

```

poppy@poppy:~$ sudo ifconfig
eth3      Link encap:Ethernet  HWaddr 00:1e:06:30:58:9c
          inet addr:192.168.1.110  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::21e:6ff:fe30:589c/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:952 errors:0 dropped:2 overruns:0 frame:0
          TX packets:407 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:170814 (170.8 KB)  TX bytes:47058 (47.0 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:288 errors:0 dropped:0 overruns:0 frame:0
          TX packets:288 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:22752 (22.7 KB)  TX bytes:22752 (22.7 KB)

wlan2     Link encap:Ethernet  HWaddr 74:da:38:6c:6c:0f
          inet addr:192.168.1.111  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::76da:38ff:fe6c:6c0f/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1554 errors:0 dropped:3 overruns:0 frame:0
          TX packets:866 errors:0 dropped:1 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:335315 (335.3 KB)  TX bytes:569367 (569.3 KB)

poppy@poppy:~$ █

```

Notre robot est maintenant accessible :

- En liaison filaire sur l'adresse 192.168.1.110
- Sans fil (wifi) sur l'adresse 192.168.1.111

L'article suivant est une excellente ressource décrivant de nombreuses configurations différentes de réseau sans fil pour les distributions basées sur Debian : <https://wiki.debian.org/WiFi/HowToUse>