

Montage d'un régulateur de temperature sur Spotnik Delta
pour le Réseau des Répéteurs Francophones

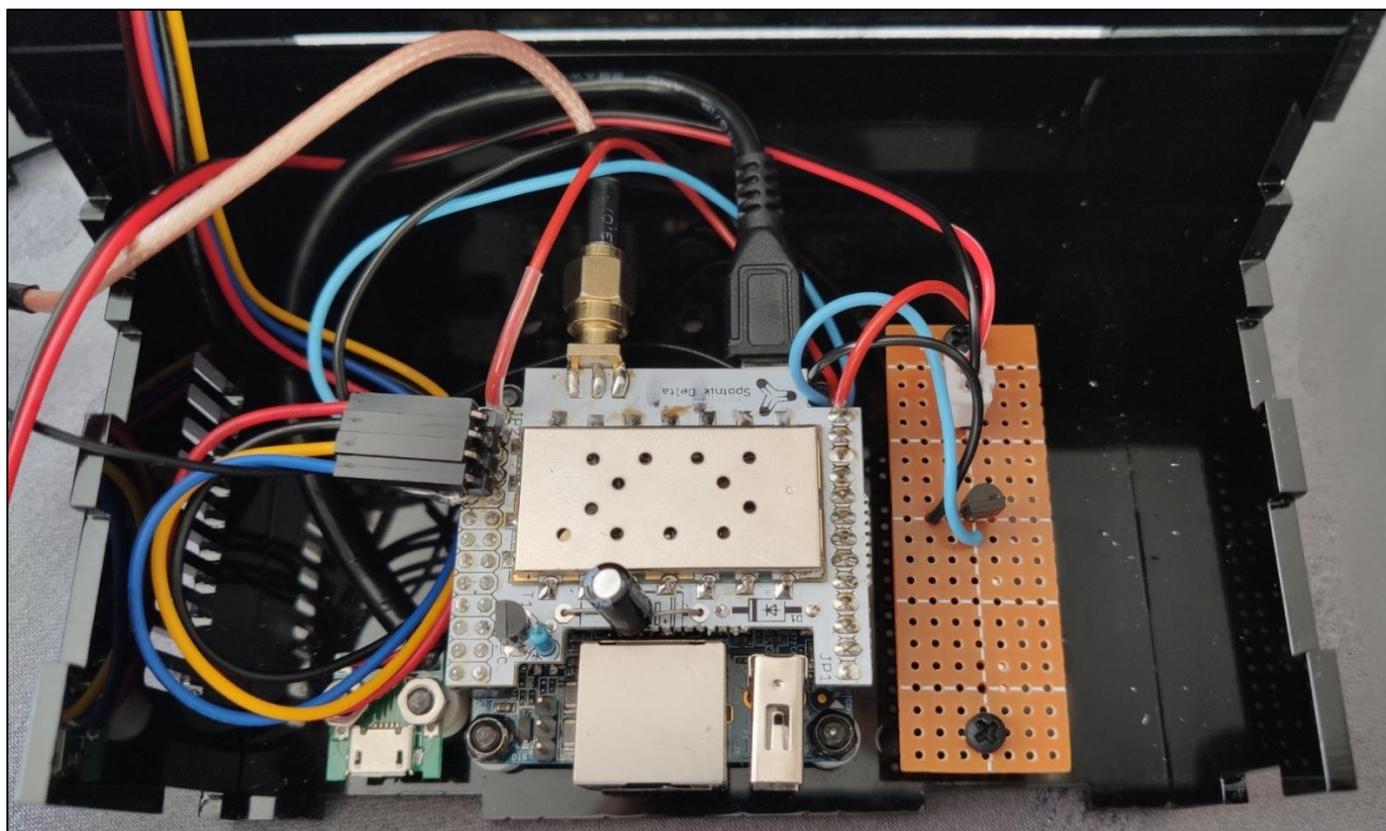


Par F4BPP

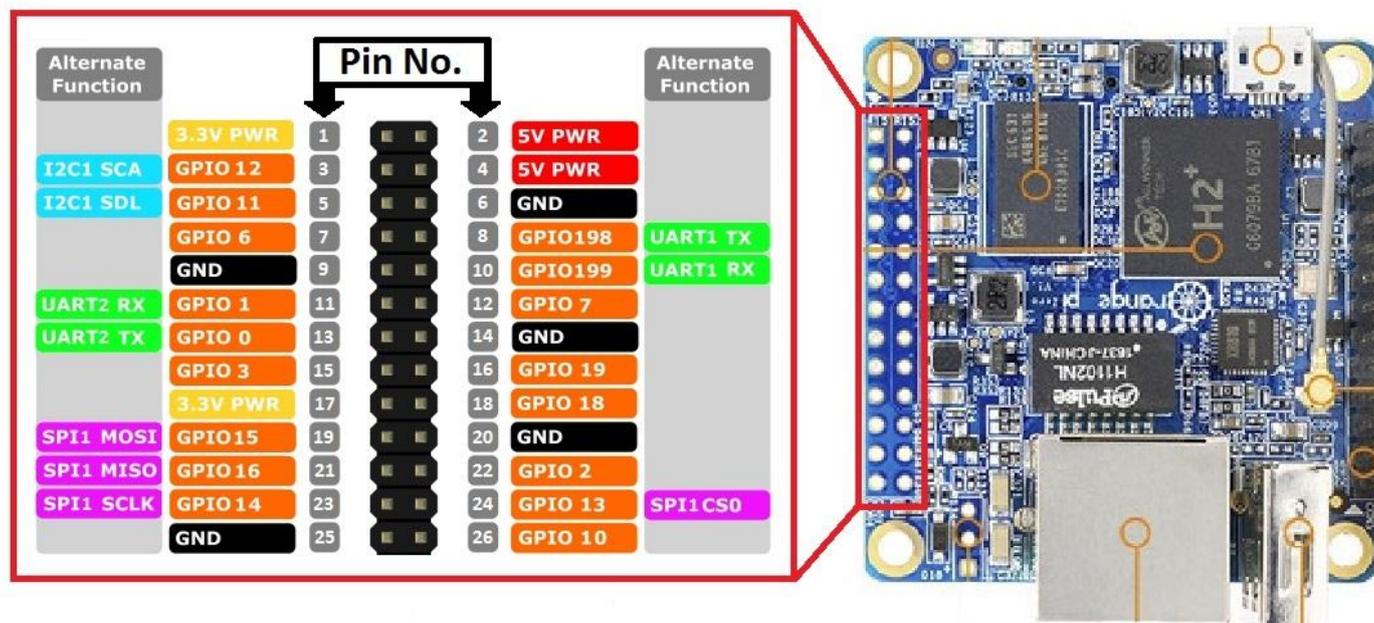
Protocole basé sur le tutoriel anglais de Fabien

(<http://www.d0wn.com/orange-pi-zero-the-battle-against-heat/>)

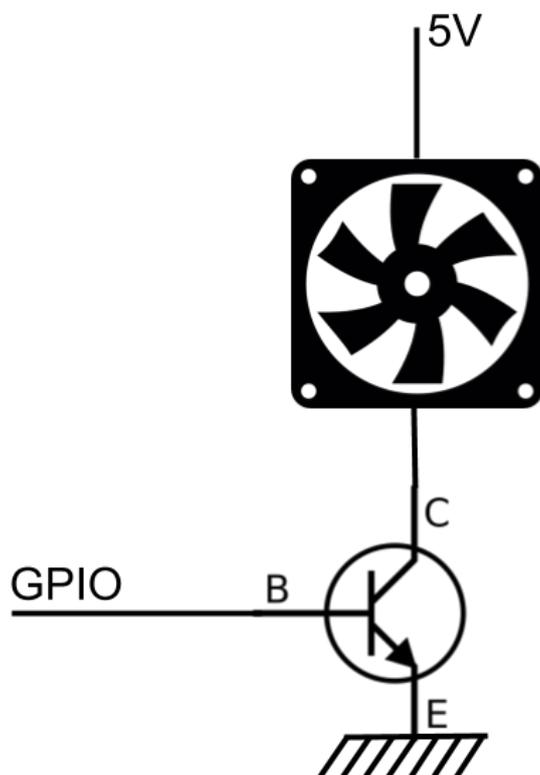
Mon spotnik Delta est connecté à un écran Nextion et le tout a été incorporé dans le boîtier « Box4Nextion » à découvrir sur le blog de F8ASB (<http://blog.f8asb.com>). Cependant, ce tutoriel peut être appliqué à tout Spotnik Delta auquel on veut ajouter un ventilateur qui sera contrôlé par le spotnik.



Voici le brochage du connecteur GPIO de l'orange PI Zero :



Le montage que nous allons réaliser consiste à connecter un ventilateur au spotnik de sorte que ce dernier puisse être activé ou désactivé en fonction de la température du processeur de l'Orange PI Zero. Pour une fonctionnement en continu, il suffit de relier la borne « + » du ventilateur à la broche n°2 et la borne « - » à la broche n°9. Mais pour pouvoir contrôler le ventilateur, nous allons utiliser le schéma de montage suivant :



Pour réaliser ce montage, il est nécessaire d'utiliser un transistor de type NPN avec comme caractéristique un ampérage supérieur à la consommation du ventilateur. Pour ma part, j'ai utilisé un KSP42 que j'avais sous la main pour faire ce montage.

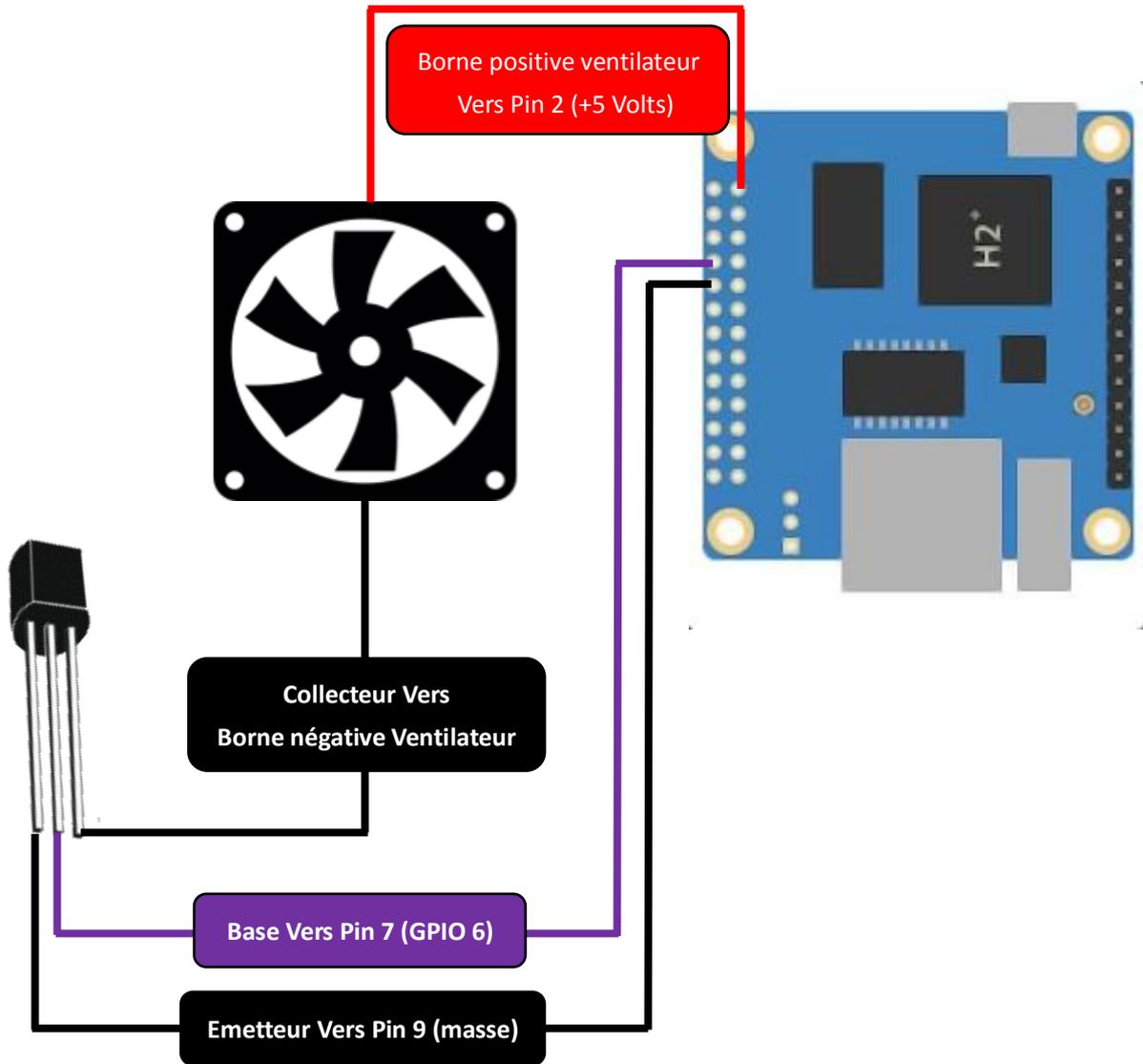


KSP42 pinout

1-Emetteur
2-Base
3-Collecteur

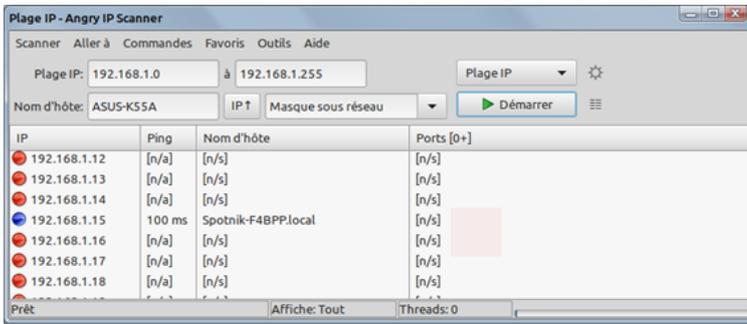
- Type - **NPN**
- Collector-Emitter Voltage: **300 V**
- Collector-Base Voltage: **300 V**
- Emitter-Base Voltage: **6 V**
- Collector Current: **0.5 A**
- Collector Dissipation - **0.625 W**
- DC Current Gain (h_{fe}) - **40**
- Transition Frequency - **50 MHz**
- Operating and Storage Junction Temperature Range **-55 to +150 °C**
- Package - **TO-92**

Voici le schéma de câblage :



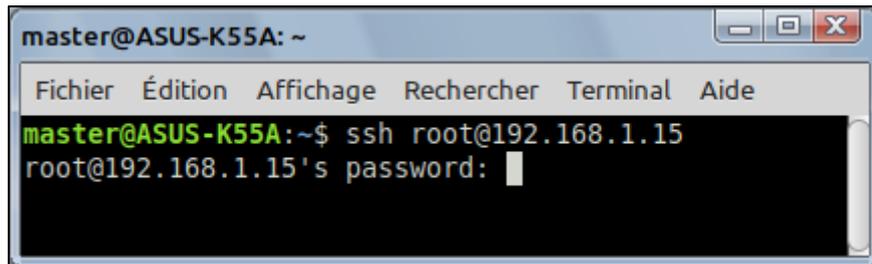
Lorsque le câblage est terminé, il est nécessaire de procéder à l'installation d'un script qui va être appelé par le système toutes les minutes afin de comparer la température maximale désirée et celle relevée sur le processeur.

Tant que le processeur est en dessous de cette température, on va laisser le ventilateur éteint. En revanche, si la température relevée sur le processeur est supérieure à la température maximale désirée, alors on va activer le ventilateur via le port GPIO 6.

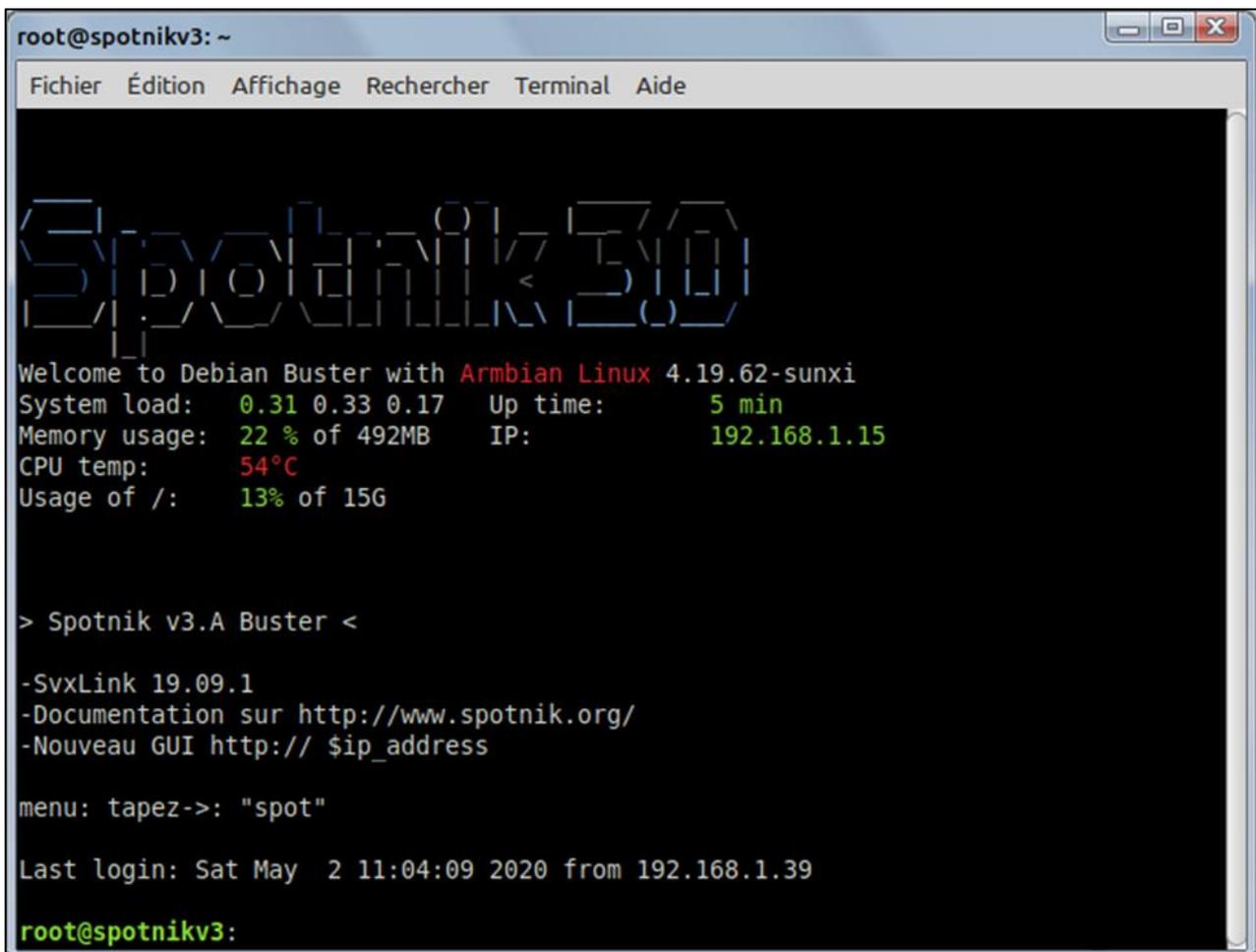


Pour commencer l'installation du script, il faut se connecter au spotnik via son adresse IP avec une liaison SSH. Pour ce faire, il faut d'abord utiliser un logiciel qui permet de scanner les adresses IP actives sur votre réseau afin de repérer celle affectée au spotnik.

Ensuite dans un terminal, on entre généralement la commande suivante : **sudo ssh root@XXX.XXX.XXX.XXX** en remplaçant les « X » par l'adresse IP du spotnik.

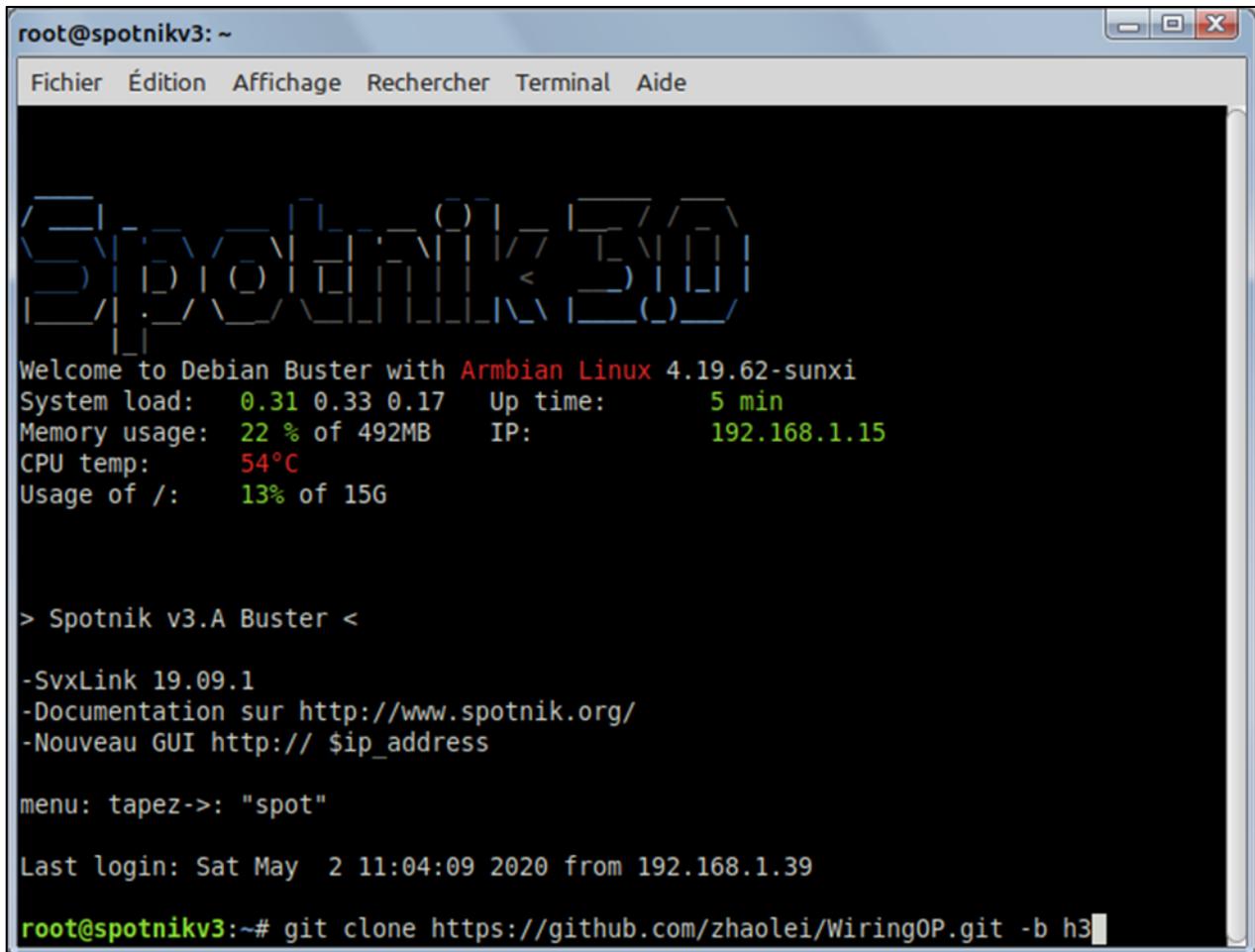


Entrez ensuite le mot de passe « **spotnik** ». Vous arrivez à la page suivante :



Pour que le script puisse fonctionner, il est nécessaire d'installer une bibliothèque de fonctions qui lui permettent de contrôler les ports GPIO.

Entrez la commande suivante via un copier-coller pour procéder au téléchargement :
git clone https://github.com/zhaolei/WiringOP.git -b h3



```
root@spotnikv3: ~
Fichier  Édition  Affichage  Rechercher  Terminal  Aide

Spotnik v3.A

Welcome to Debian Buster with Armbian Linux 4.19.62-sunxi
System load:  0.31 0.33 0.17  Up time:      5 min
Memory usage: 22 % of 492MB  IP:         192.168.1.15
CPU temp:     54°C
Usage of /:   13% of 15G

> Spotnik v3.A Buster <

-SvxLink 19.09.1
-Documentation sur http://www.spotnik.org/
-Nouveau GUI http:// $ip_address

menu: tapez->: "spot"

Last login: Sat May  2 11:04:09 2020 from 192.168.1.39

root@spotnikv3:~# git clone https://github.com/zhaolei/WiringOP.git -b h3
```

Cette commande va cloner via le téléchargement, le dossier « **WiringOP** » dans le dossier « **root** » de votre système.

Dans ce dossier se trouvent les programmes à installer après avoir lancé une compilation.

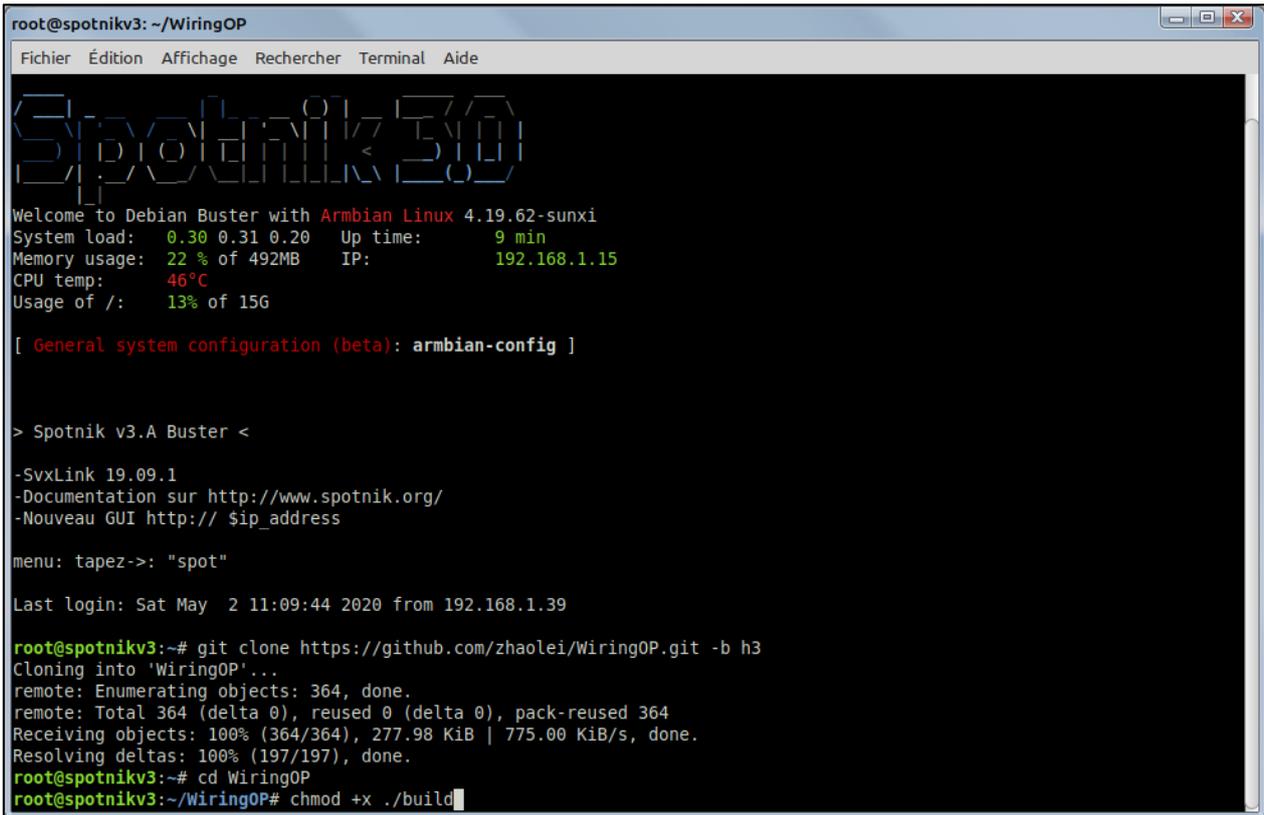
Voici les informations qui s'affichent à l'écran si tout s'est bien déroulé :

```
root@spotnikv3: ~  
Fichier  Édition  Affichage  Rechercher  Terminal  Aide  
Spotnik v3.0  
Welcome to Debian Buster with Armbian Linux 4.19.62-sunxi  
System load:  0.30 0.31 0.20  Up time:    9 min  
Memory usage: 22 % of 492MB  IP:       192.168.1.15  
CPU temp:    46°C  
Usage of /:  13% of 15G  
  
[ General system configuration (beta): armbian-config ]  
  
> Spotnik v3.A Buster <  
  
-SvxLink 19.09.1  
-Documentation sur http://www.spotnik.org/  
-Nouveau GUI http:// $ip_address  
  
menu: tapez->: "spot"  
  
Last login: Sat May  2 11:09:44 2020 from 192.168.1.39  
  
root@spotnikv3:~# git clone https://github.com/zhaolei/WiringOP.git -b h3  
Cloning into 'WiringOP'...  
remote: Enumerating objects: 364, done.  
remote: Total 364 (delta 0), reused 0 (delta 0), pack-reused 364  
Receiving objects: 100% (364/364), 277.98 KiB | 775.00 KiB/s, done.  
Resolving deltas: 100% (197/197), done.  
root@spotnikv3:~#
```

Entrez la commande « **cd WiringOP** » pour entrer dans le dossier qui vient d'être téléchargé :

```
root@spotnikv3: ~  
Fichier  Édition  Affichage  Rechercher  Terminal  Aide  
Spotnik v3.0  
Welcome to Debian Buster with Armbian Linux 4.19.62-sunxi  
System load:  0.30 0.31 0.20  Up time:    9 min  
Memory usage: 22 % of 492MB  IP:       192.168.1.15  
CPU temp:    46°C  
Usage of /:  13% of 15G  
  
[ General system configuration (beta): armbian-config ]  
  
> Spotnik v3.A Buster <  
  
-SvxLink 19.09.1  
-Documentation sur http://www.spotnik.org/  
-Nouveau GUI http:// $ip_address  
  
menu: tapez->: "spot"  
  
Last login: Sat May  2 11:09:44 2020 from 192.168.1.39  
  
root@spotnikv3:~# git clone https://github.com/zhaolei/WiringOP.git -b h3  
Cloning into 'WiringOP'...  
remote: Enumerating objects: 364, done.  
remote: Total 364 (delta 0), reused 0 (delta 0), pack-reused 364  
Receiving objects: 100% (364/364), 277.98 KiB | 775.00 KiB/s, done.  
Resolving deltas: 100% (197/197), done.  
root@spotnikv3:~# cd WiringOP
```

Il faut ensuite utiliser une commande pour changer les autorisations d'accès au dossier « **build** » qui se trouve à l'intérieur. Saisir « **chmod +x ./build** » :



```
root@spotnikv3: ~/WiringOP
Fichier  Édition  Affichage  Rechercher  Terminal  Aide

Spotnik 3.0

Welcome to Debian Buster with Armbian Linux 4.19.62-sunxi
System load:  0.30 0.31 0.20  Up time:    9 min
Memory usage: 22 % of 492MB  IP:       192.168.1.15
CPU temp:    46°C
Usage of /:   13% of 15G

[ General system configuration (beta): armbian-config ]

> Spotnik v3.A Buster <

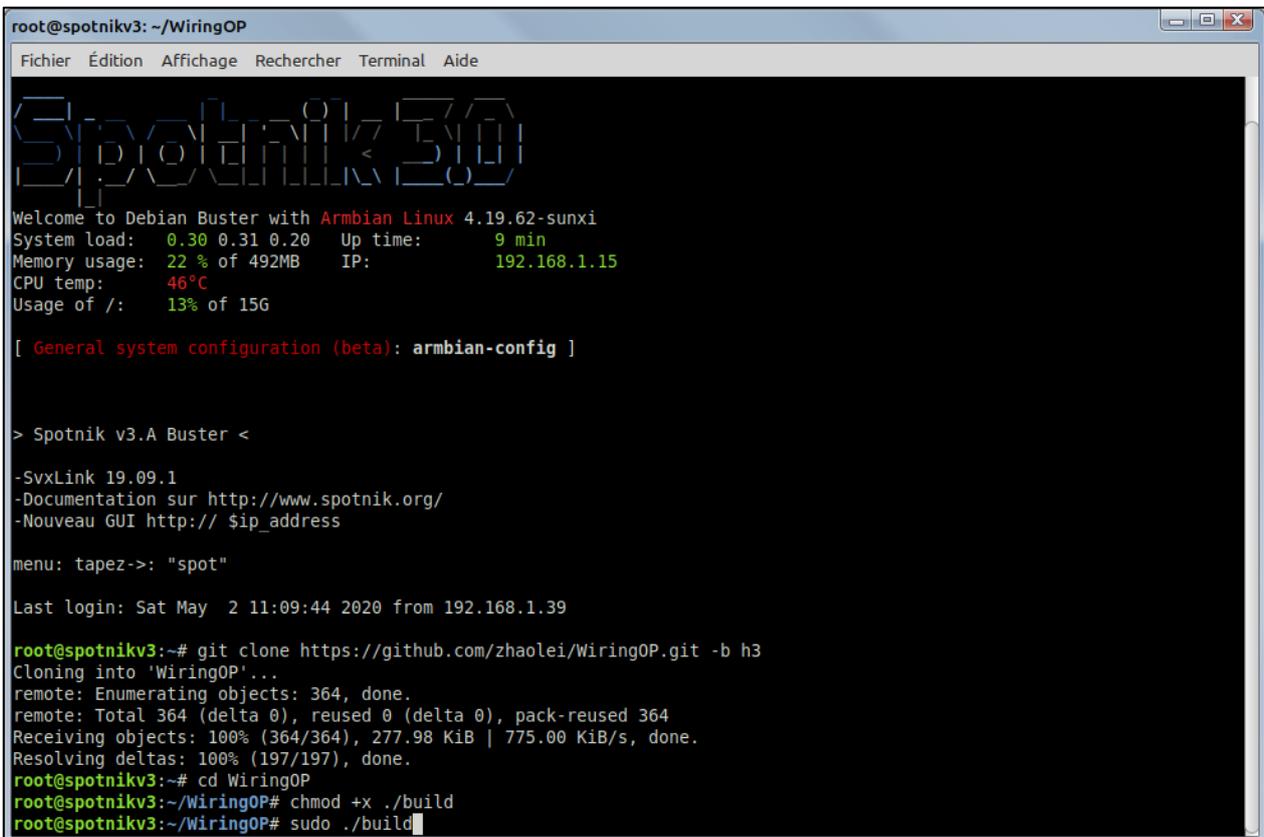
-SvxLink 19.09.1
-Documentation sur http://www.spotnik.org/
-Nouveau GUI http:// $ip_address

menu: tapez->: "spot"

Last login: Sat May  2 11:09:44 2020 from 192.168.1.39

root@spotnikv3:~# git clone https://github.com/zhaolei/WiringOP.git -b h3
Cloning into 'WiringOP'...
remote: Enumerating objects: 364, done.
remote: Total 364 (delta 0), reused 0 (delta 0), pack-reused 364
Receiving objects: 100% (364/364), 277.98 KiB | 775.00 KiB/s, done.
Resolving deltas: 100% (197/197), done.
root@spotnikv3:~# cd WiringOP
root@spotnikv3:~/WiringOP# chmod +x ./build
```

Pour lancer la compilation des programmes saisir **sudo ./build** dans le terminal :



```
root@spotnikv3: ~/WiringOP
Fichier  Édition  Affichage  Rechercher  Terminal  Aide

Spotnik 3.0

Welcome to Debian Buster with Armbian Linux 4.19.62-sunxi
System load:  0.30 0.31 0.20  Up time:    9 min
Memory usage: 22 % of 492MB  IP:       192.168.1.15
CPU temp:    46°C
Usage of /:   13% of 15G

[ General system configuration (beta): armbian-config ]

> Spotnik v3.A Buster <

-SvxLink 19.09.1
-Documentation sur http://www.spotnik.org/
-Nouveau GUI http:// $ip_address

menu: tapez->: "spot"

Last login: Sat May  2 11:09:44 2020 from 192.168.1.39

root@spotnikv3:~# git clone https://github.com/zhaolei/WiringOP.git -b h3
Cloning into 'WiringOP'...
remote: Enumerating objects: 364, done.
remote: Total 364 (delta 0), reused 0 (delta 0), pack-reused 364
Receiving objects: 100% (364/364), 277.98 KiB | 775.00 KiB/s, done.
Resolving deltas: 100% (197/197), done.
root@spotnikv3:~# cd WiringOP
root@spotnikv3:~/WiringOP# chmod +x ./build
root@spotnikv3:~/WiringOP# sudo ./build
```

Patiencez quelques instants pendant que la compilation s'opère...

```
root@spotnikv3: ~/WiringOP
Fichier  Édition  Affichage  Rechercher  Terminal  Aide
./build: 4: ./build: [[: not found
wiringPi Build script
=====

WiringPi Library
[UnInstall]
[Compile] wiringPi.c
[Compile] wiringSerial.c
[Compile] wiringShift.c
[Compile] piHiPri.c
[Compile] piThread.c
[Compile] wiringPiSPI.c
[Compile] wiringPiI2C.c
[Compile] softPwm.c
[Compile] softTone.c
[Compile] softServo.c
[Compile] mcp23008.c
[Compile] mcp23016.c
[Compile] mcp23017.c
[Compile] mcp23s08.c
[Compile] mcp23s17.c
[Compile] sr595.c
[Compile] pcf8574.c
[Compile] pcf8591.c
[Compile] mcp3002.c
[Compile] mcp3004.c
[Compile] mcp4802.c
[Compile] mcp3422.c
[Compile] max31855.c
[Compile] max5322.c
[Compile] sn3218.c
[Compile] drcSerial.c
[Link (Dynamic)]
[Install Headers]
```

```
root@spotnikv3: ~/WiringOP
Fichier  Édition  Affichage  Rechercher  Terminal  Aide
[Install Dynamic Lib]

WiringPi Devices Library
[UnInstall]
[Compile] ds1302.c
[Compile] maxdetect.c
[Compile] piNes.c
[Compile] gertboard.c
[Compile] piFace.c
[Compile] lcd128x64.c
[Compile] lcd.c
[Compile] piGlow.c
[Link (Dynamic)]
[Install Headers]
[Install Dynamic Lib]

GPIO Utility
[Compile] gpio.c
gpio.c:943:13: warning: 'doGbr' defined but not used [-Wunused-function]
  static void doGbr (int argc, char *argv [])
           ^~~~~~
gpio.c:901:13: warning: 'doGbw' defined but not used [-Wunused-function]
  static void doGbw (int argc, char *argv [])
           ^~~~~~
gpio.c:813:13: warning: 'doPadDrive' defined but not used [-Wunused-function]
  static void doPadDrive (int argc, char *argv [])
           ^~~~~~
[Compile] extensions.c
[Compile] readall.c
[Compile] pins.c
[Link]
[Install]

All Done.
```

Après quelques instant, si tout s'est bien déroulé, vous devriez obtenir l'affichage suivant :

```
root@spotnikv3: ~/WiringOP
Fichier  Édition  Affichage  Rechercher  Terminal  Aide
[Compile] piFace.c
[Compile] lcd128x64.c
[Compile] lcd.c
[Compile] piGlow.c
[Link (Dynamic)]
[Install Headers]
[Install Dynamic Lib]

GPIO Utility
[Compile] gpio.c
gpio.c:943:13: warning: 'doGbr' defined but not used [-Wunused-function]
static void doGbr (int argc, char *argv [])
      ^~~~~~
gpio.c:901:13: warning: 'doGbw' defined but not used [-Wunused-function]
static void doGbw (int argc, char *argv [])
      ^~~~~~
gpio.c:813:13: warning: 'doPadDrive' defined but not used [-Wunused-function]
static void doPadDrive (int argc, char *argv [])
      ^~~~~~
[Compile] extensions.c
[Compile] readall.c
[Compile] pins.c
[Link]
[Install]

All Done.

NOTE: To compile programs with wiringBP v2, you need to add:
-lwiringPi
to your compile line(s) To use the Gertboard, MaxDetect, etc.
code (the devLib), you need to also add:
-lwiringPiDev
to your compile line(s).

root@spotnikv3:~/WiringOP#
```

Entrez ensuite la commande « `cd ..` » pour sortir du dossier « WiringOP » :

```
root@spotnikv3: ~/WiringOP
Fichier  Édition  Affichage  Rechercher  Terminal  Aide
[Compile] piFace.c
[Compile] lcd128x64.c
[Compile] lcd.c
[Compile] piGlow.c
[Link (Dynamic)]
[Install Headers]
[Install Dynamic Lib]

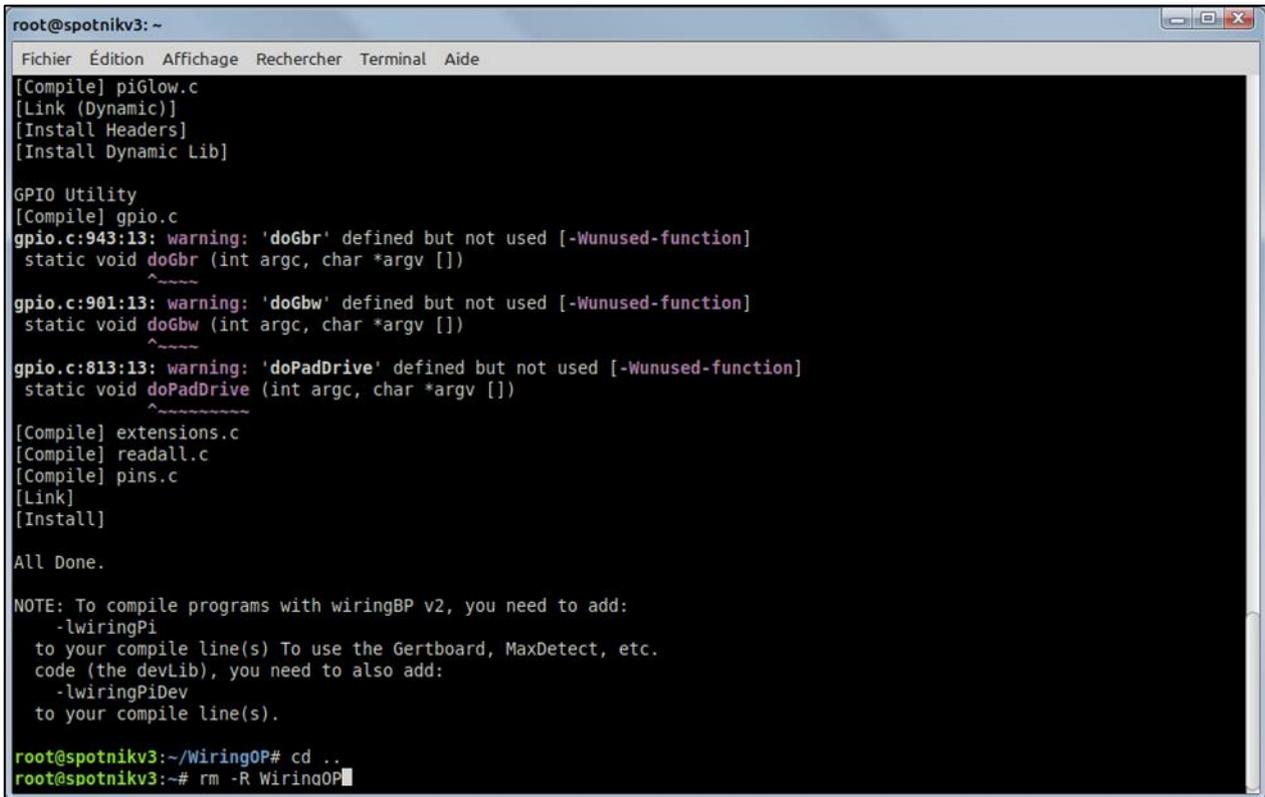
GPIO Utility
[Compile] gpio.c
gpio.c:943:13: warning: 'doGbr' defined but not used [-Wunused-function]
static void doGbr (int argc, char *argv [])
      ^~~~~~
gpio.c:901:13: warning: 'doGbw' defined but not used [-Wunused-function]
static void doGbw (int argc, char *argv [])
      ^~~~~~
gpio.c:813:13: warning: 'doPadDrive' defined but not used [-Wunused-function]
static void doPadDrive (int argc, char *argv [])
      ^~~~~~
[Compile] extensions.c
[Compile] readall.c
[Compile] pins.c
[Link]
[Install]

All Done.

NOTE: To compile programs with wiringBP v2, you need to add:
-lwiringPi
to your compile line(s) To use the Gertboard, MaxDetect, etc.
code (the devLib), you need to also add:
-lwiringPiDev
to your compile line(s).

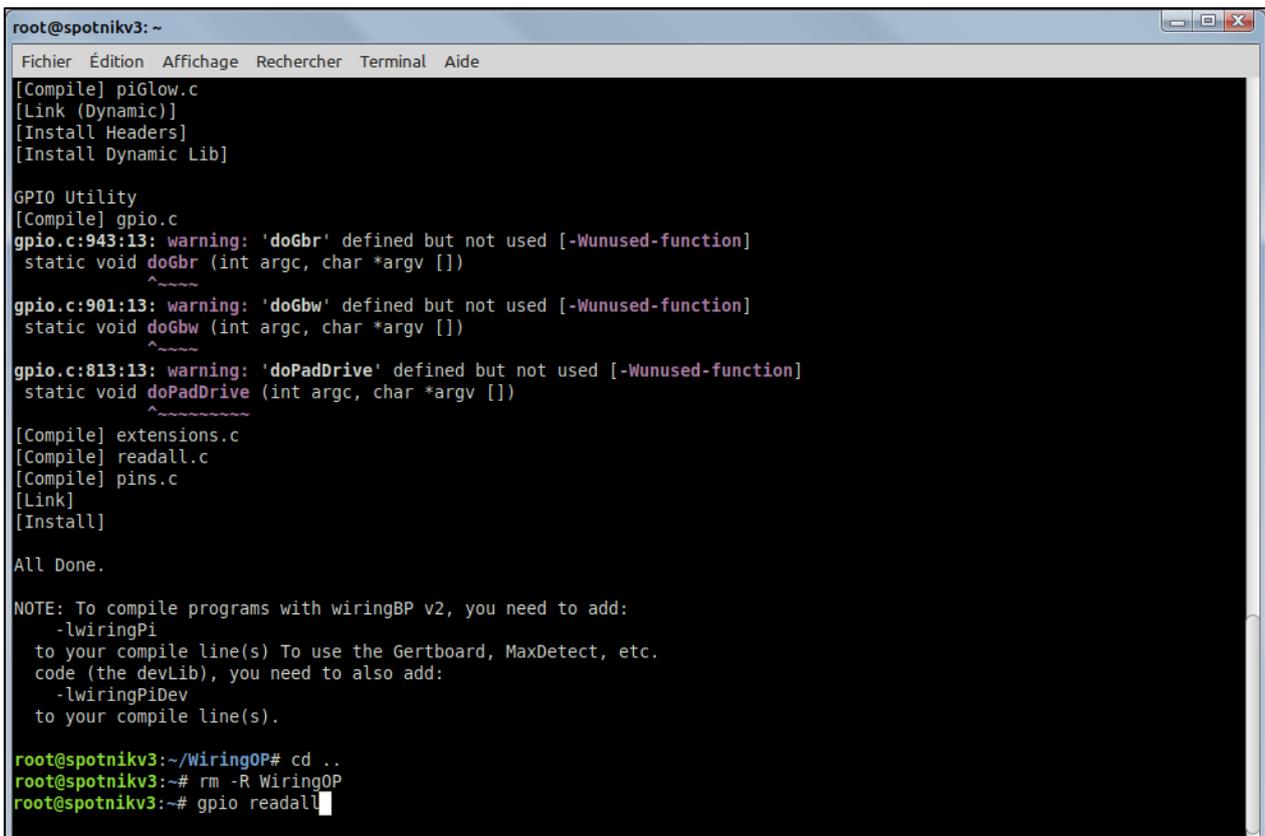
root@spotnikv3:~/WiringOP# cd ..
```

Nous pouvons à présent effacer le répertoire WiringOP qui ne nous est plus utile. Pour ce faire entrez la commande « **rm -R WiringOP** » :



```
root@spotnikv3: ~  
Fichier Édition Affichage Rechercher Terminal Aide  
[Compile] piGlow.c  
[Link (Dynamic)]  
[Install Headers]  
[Install Dynamic Lib]  
  
GPIO Utility  
[Compile] gpio.c  
gpio.c:943:13: warning: 'doGbr' defined but not used [-Wunused-function]  
static void doGbr (int argc, char *argv [])  
^~~~~~  
gpio.c:901:13: warning: 'doGbw' defined but not used [-Wunused-function]  
static void doGbw (int argc, char *argv [])  
^~~~~~  
gpio.c:813:13: warning: 'doPadDrive' defined but not used [-Wunused-function]  
static void doPadDrive (int argc, char *argv [])  
^~~~~~  
[Compile] extensions.c  
[Compile] readall.c  
[Compile] pins.c  
[Link]  
[Install]  
  
All Done.  
  
NOTE: To compile programs with wiringBP v2, you need to add:  
-lwiringPi  
to your compile line(s) To use the Gertboard, MaxDetect, etc.  
code (the devLib), you need to also add:  
-lwiringPiDev  
to your compile line(s).  
  
root@spotnikv3:~/WiringOP# cd ..  
root@spotnikv3:~# rm -R WiringOP
```

Nous allons à présent vérifier que le programme de gestion des ports GPIO a bien été installé avec la commande « **gpio readall** » :

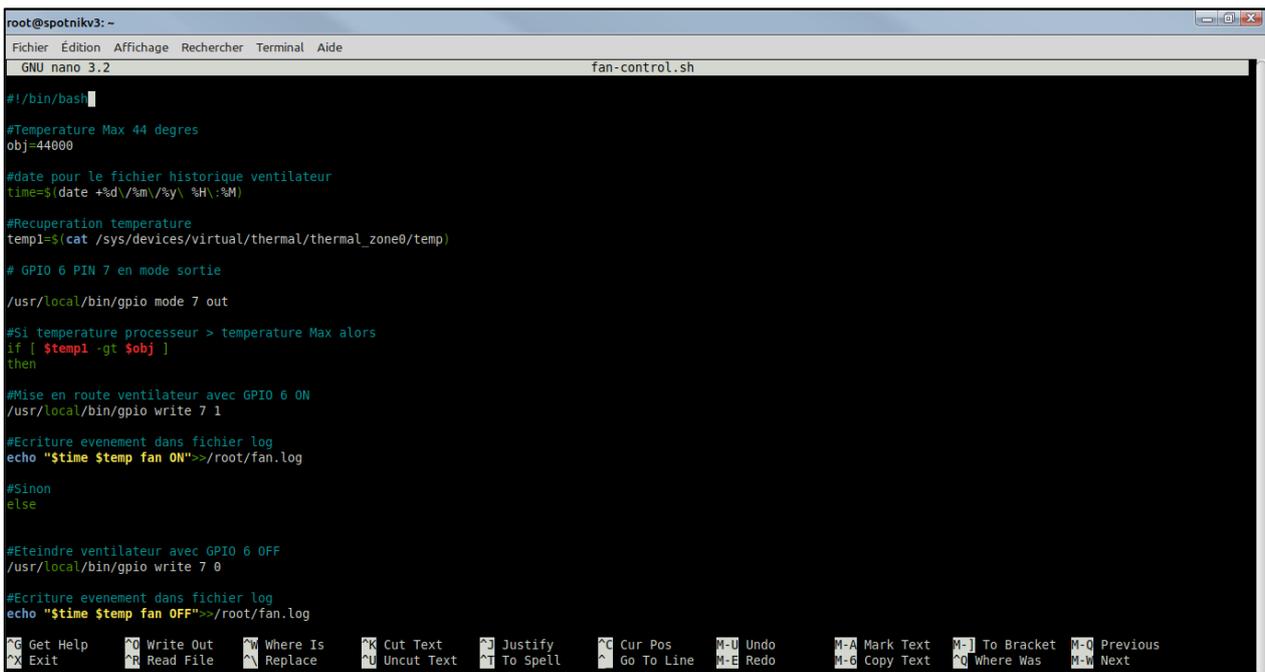


```
root@spotnikv3: ~  
Fichier Édition Affichage Rechercher Terminal Aide  
[Compile] piGlow.c  
[Link (Dynamic)]  
[Install Headers]  
[Install Dynamic Lib]  
  
GPIO Utility  
[Compile] gpio.c  
gpio.c:943:13: warning: 'doGbr' defined but not used [-Wunused-function]  
static void doGbr (int argc, char *argv [])  
^~~~~~  
gpio.c:901:13: warning: 'doGbw' defined but not used [-Wunused-function]  
static void doGbw (int argc, char *argv [])  
^~~~~~  
gpio.c:813:13: warning: 'doPadDrive' defined but not used [-Wunused-function]  
static void doPadDrive (int argc, char *argv [])  
^~~~~~  
[Compile] extensions.c  
[Compile] readall.c  
[Compile] pins.c  
[Link]  
[Install]  
  
All Done.  
  
NOTE: To compile programs with wiringBP v2, you need to add:  
-lwiringPi  
to your compile line(s) To use the Gertboard, MaxDetect, etc.  
code (the devLib), you need to also add:  
-lwiringPiDev  
to your compile line(s).  
  
root@spotnikv3:~/WiringOP# cd ..  
root@spotnikv3:~# rm -R WiringOP  
root@spotnikv3:~# gpio readall
```


Copiez-collez le texte du script suivant dans l'éditeur :

```
#!/bin/bash
# Temperature Max 44 degrees
obj=44000
#date pour le fichier historique ventilateur
time=$(date +%dV%mV%y\ %H\:%M)
# Recuperation temperature
temp1=$(cat /sys/devices/virtual/thermal/thermal_zone0/temp)
# GPIO 6 PIN 7 en mode sortie
/usr/local/bin/gpio mode 7 out
# Si temperature processeur > temperature Max alors
if [ $temp1 -gt $obj ]
then
# Mise en route ventilateur avec GPIO 6 ON
/usr/local/bin/gpio write 7 1
# Ecriture evenement dans fichier log
echo "$time $temp fan ON">>/root/fan.log
# Sinon
else
# Eteindre ventilateur avec GPIO 6 OFF
/usr/local/bin/gpio write 7 0
# Ecriture evenement dans fichier log
echo "$time $temp fan OFF">>/root/fan.log
#Fin Si
fi
```

Vous devriez obtenir l'affichage suivant :



```
root@spotnikv3: ~
Fichier Edition Affichage Rechercher Terminal Aide
GNU nano 3.2 fan-control.sh

#!/bin/bash
#Temperature Max 44 degrees
obj=44000
#date pour le fichier historique ventilateur
time=$(date +%dV%mV%y\ %H\:%M)
#Recuperation temperature
temp1=$(cat /sys/devices/virtual/thermal/thermal_zone0/temp)
# GPIO 6 PIN 7 en mode sortie
/usr/local/bin/gpio mode 7 out
#Si temperature processeur > temperature Max alors
if [ $temp1 -gt $obj ]
then
#Mise en route ventilateur avec GPIO 6 ON
/usr/local/bin/gpio write 7 1
#Ecriture evenement dans fichier log
echo "$time $temp fan ON">>/root/fan.log
#Sinon
else
#Eteindre ventilateur avec GPIO 6 OFF
/usr/local/bin/gpio write 7 0
#Ecriture evenement dans fichier log
echo "$time $temp fan OFF">>/root/fan.log
```

Pour sauvegarder le script, effectuez la combinaison de touches « **Ctrl** » + « **O** » et validez avec la touche « **Entrée** » :

```

root@spotnikv3: ~
Fichier Edition Affichage Rechercher Terminal Aide
GNU nano 3.2 fan-control.sh

#!/bin/bash

#Temperature Max 44 degrees
obj=44000

#date pour le fichier historique ventilateur
time=$(date +%d/%m/%y\ %H:%M)

#Recuperation temperature
temp1=$(cat /sys/devices/virtual/thermal/thermal_zone0/temp)

# GPIO 6 PIN 7 en mode sortie
/usr/local/bin/gpio mode 7 out

#Si temperature processeur > temperature Max alors
if [ $temp1 -gt $obj ]
then
#Mise en route ventilateur avec GPIO 6 ON
/usr/local/bin/gpio write 7 1

#Ecriture evenement dans fichier log
echo "$time $temp fan ON">>/root/fan.log

#Sinon
else

#Eteindre ventilateur avec GPIO 6 OFF
/usr/local/bin/gpio write 7 0

#Ecriture evenement dans fichier log
echo "$time $temp fan OFF">>/root/fan.log

File Name to Write: fan-control.sh
^G Get Help      ^M-DOS Format   ^M-A Append     ^M-B Backup File
^C Cancel        ^M-Mac Format   ^M-P Prepend    ^M-T To Files

```

Nous allons à présent demander au système d'exécuter ce script toutes les minutes. Pour ce faire, entrez la commande « **crontab -e** » :

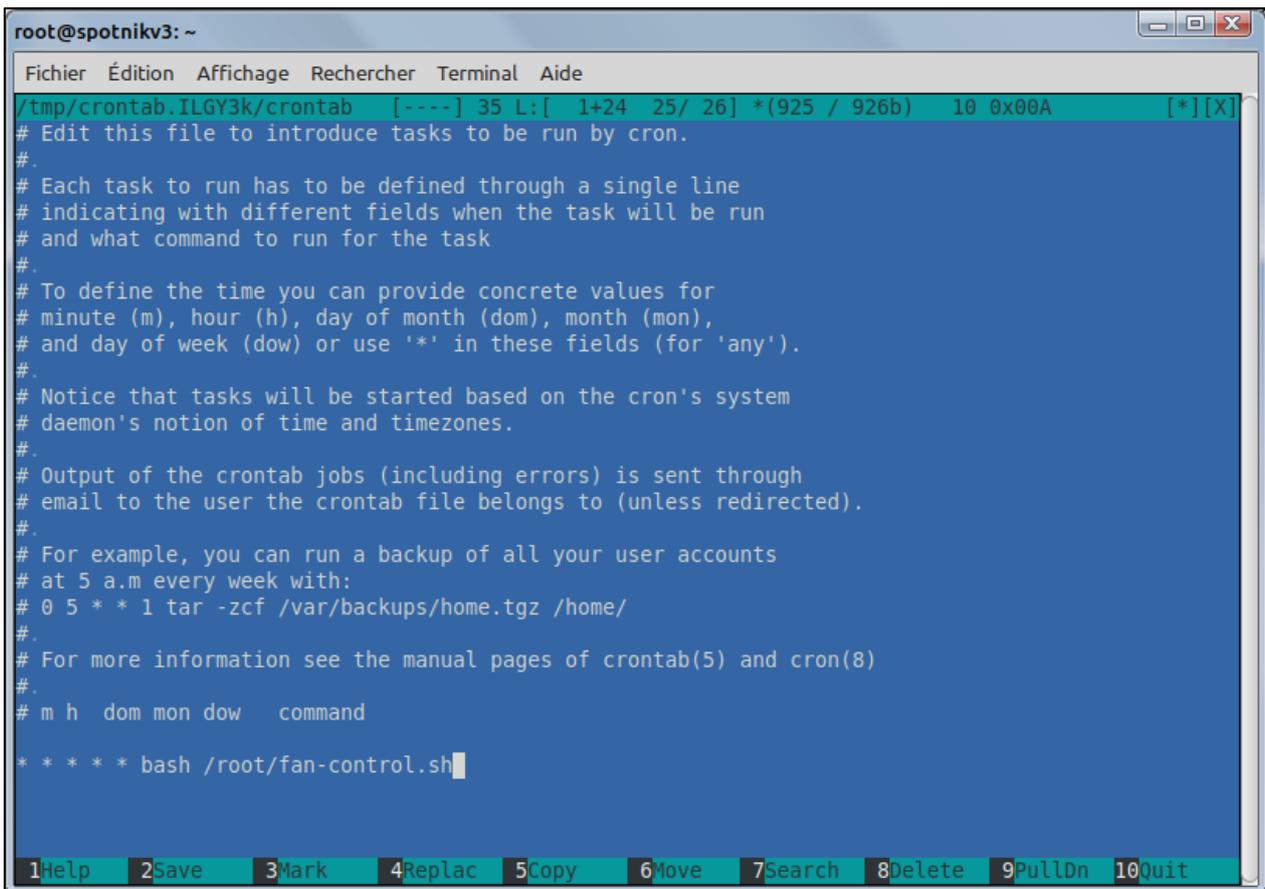
```

root@spotnikv3: ~
Fichier Edition Affichage Rechercher Terminal Aide
root@spotnikv3:~# gpio readall
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| BCM | wPi | Name | Mode | V | Physical | V | Mode | Name | wPi | BCM |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 12 | 8 | SDA.0 | ALT3 | 0 | 3 | 4 | | | 5v | | |
| 11 | 9 | SCL.0 | ALT3 | 0 | 5 | 6 | | | 5V | | |
| 6 | 7 | GPIO.7 | OUT | 1 | 7 | 8 | 0 | ALT4 | TxD3 | 15 | 13 |
| | | 0v | | | 9 | 10 | 0 | ALT4 | RxD3 | 16 | 14 |
| 1 | 0 | RxD2 | ALT5 | 0 | 11 | 12 | 0 | ALT3 | GPIO.1 | 1 | 110 |
| 0 | 2 | TxD2 | ALT5 | 0 | 13 | 14 | | | 0v | | |
| 3 | 3 | CTS2 | ALT3 | 0 | 15 | 16 | 0 | ALT3 | GPIO.4 | 4 | 68 |
| | | 3.3v | | | 17 | 18 | 0 | ALT3 | GPIO.5 | 5 | 71 |
| 64 | 12 | MOSI | ALT3 | 0 | 19 | 20 | | | 0v | | |
| 65 | 13 | MISO | ALT3 | 0 | 21 | 22 | 0 | IN | RTS2 | 6 | 2 |
| 66 | 14 | SCLK | ALT3 | 0 | 23 | 24 | 0 | ALT3 | CE0 | 10 | 67 |
| | | 0v | | | 25 | 26 | 0 | ALT3 | GPIO.11 | 11 | 21 |
| 19 | 30 | SDA.1 | ALT3 | 0 | 27 | 28 | 0 | ALT3 | SCL.1 | 31 | 18 |
| 7 | 21 | GPIO.21 | OUT | 0 | 29 | 30 | | | 0v | | |
| 8 | 22 | GPIO.22 | ALT3 | 0 | 31 | 32 | 0 | ALT3 | RTS1 | 26 | 200 |
| 9 | 23 | GPIO.23 | ALT3 | 0 | 33 | 34 | | | 0v | | |
| 10 | 24 | GPIO.24 | IN | 0 | 35 | 36 | 0 | ALT3 | CTS1 | 27 | 201 |
| 20 | 25 | GPIO.25 | OUT | 1 | 37 | 38 | 0 | ALT5 | TxD1 | 28 | 198 |
| | | 0v | | | 39 | 40 | 0 | ALT5 | RxD1 | 29 | 199 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| BCM | wPi | Name | Mode | V | Physical | V | Mode | Name | wPi | BCM |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
root@spotnikv3:~# nano fan-control.sh
root@spotnikv3:~# crontab -e

```

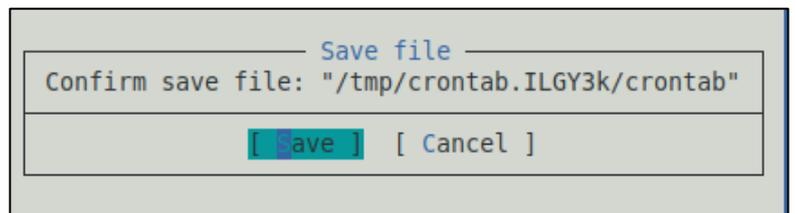
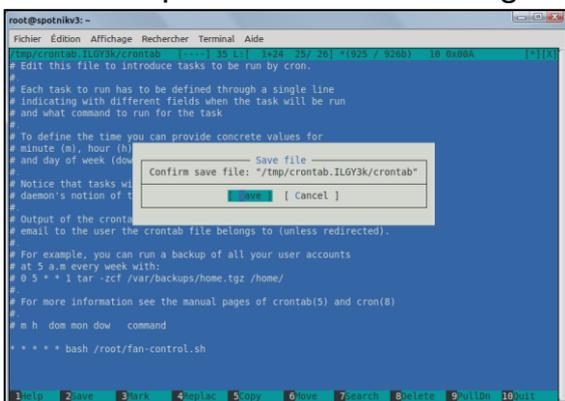
Ajoutez la ligne suivante à la fin du fichier :

*** * * * * bash/root/fan-control.sh**



Pour sauvegarder le script, pressez la touche « **F2** » du clavier ou cliquez directement sur [2 Save] au bas de l'écran avec la souris.

Une fenêtre de confirmation va apparaître au milieu de l'écran. Pressez la touche « **Entrée** » pour valider la sauvegarde.



Cliquer ensuite sur « **10 Quit** » en bas à droite de l'écran.

S'il n'y a pas d'erreurs dans l'ajout de la commande, vous devriez avoir le message suivant qui confirme que le lancement du script a bien été pris en compte par le gestionnaire de tâches du système :

```
root@spotnikv3: ~  
Fichier  Édition  Affichage  Rechercher  Terminal  Aide  
root@spotnikv3:~# crontab -e  
crontab: installing new crontab  
root@spotnikv3:~#
```

Il ne vous reste plus qu'à redémarrer le Spotnik !



73 de F4BPP.