

# Utiliser un GPS comme horloge pour le Raspberry Pi

---

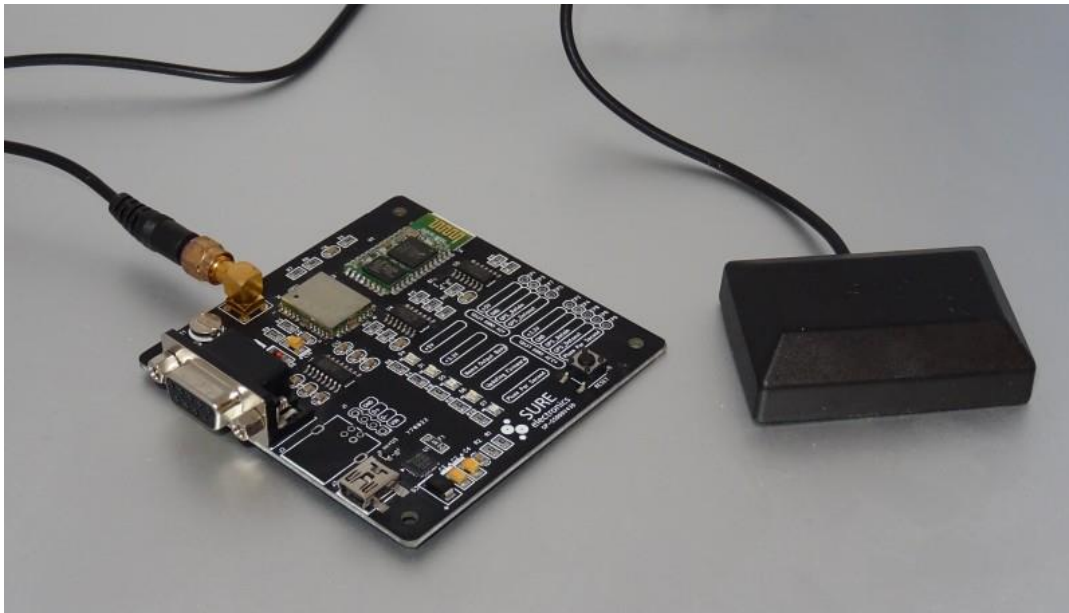
Le 24/07/2015 par [Pierre Dandumont](#)

*J'en avais parlé lors des projets de [serveurs](#) pour les [Nabaztag](#), le Raspberry Pi a beaucoup d'avantages mais aussi quelques défauts, comme l'absence d'horloge interne. La solution que j'ai utilisé consistait à installer une puce RTC, capable de garder l'heure quand la machine est éteinte, mais il existe une autre solution, parfois plus efficace : un récepteur GPS*

<https://www.journaldulapin.com/2015/07/24/a-terminer-utiliser-un-gps-comme-horloge-pour-le-raspberry-pi/>



Le problème du RTC, c'est qu'il faut fixer l'heure au départ (généralement avec un serveur NTP), ce qui n'est pas nécessairement l'idéal quand on veut séparer un Raspberry Pi du réseau. L'utilisation d'un récepteur GPS permet d'obtenir l'heure sans connexion au réseau, avec une précision très correcte dans la majorité des cas. Je ne vais pas expliquer ici comment faire un serveur NTP parfaitement synchronisé (Stratum 1), parce que je n'ai pas le matériel nécessaire ni véritablement besoin d'un grand niveau de précision : le but est juste d'avoir un appareil à l'heure, à quelques seconde près.



### Mon récepteur GPS de test

Première étape, récupérer un récepteur GPS. Il en existe en Bluetooth, en série, en GPIO ou même en USB, mais dans tous les cas, c'est en fait un signal série qui passe, avec un adaptateur dans certains cas. Mon GPS propose les quatre interfaces, mais le plus simple reste l'USB. Une fois le GPS branché, il suffit de vérifier qu'il envoie bien les données sur le port série, avec par exemple la commande suivante, à adapter en fonction de votre GPS

```
screen /dev/ttyUSB0 9600
```

Ca doit donner des lignes avec ce genre de résultats :

```
$GPVTG,0.00,T,,M,0.00,N,0.00,K,A*3D.
```

Quittez Screen (ctrl + a, puis k) et passez à la suite.

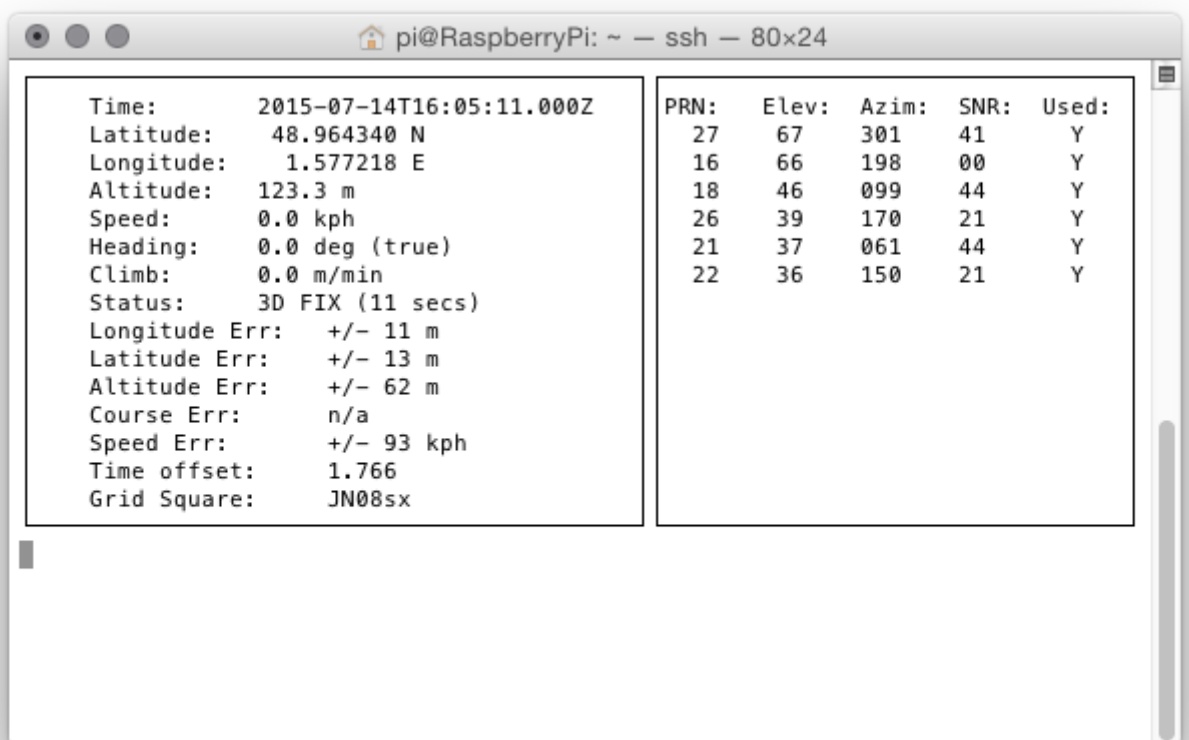
Ensuite, il faut installer GPSD, un logiciel qui va faire le pont entre le serveur de temps et le GPS. Pour la configuration (la seconde ligne), il faut lancer le logiciel au démarrage, ne pas rattacher le GPS automatiquement, utiliser le port série du GPS (normalement /dev/ttyUSB0) et laisser le reste en standard.

```
sudo apt-get install gpsd gpsd-clients  
sudo dpkg-reconfigure gpsd
```

Un petit redémarrage (sudo reboot) et on continue.

Maintenant, testons le GPS, avec la commande suivante (il faut évidemment attendre le *fix*).

```
cgps -s
```



Ça marche

Editons le fichier de configuration de NTP et mettons en commentaires les lignes qui commencent par `server 0.debian.pool`. Ce n'est nécessaire que si vous comptez déconnecter le Raspberry Pi, ceci dit.

```
sudo nano /etc/ntp.conf
```

Ajoutez les lignes suivantes à la fin du fichier.

```
# gps ntp
server 127.127.28.0 minpoll 4
fudge 127.127.28.0 time1 0.183 refid NMEA
```

Un petit redémarrage (encore) et on peut tester. Pour vérifier si ça marche, il suffit de regarder la présence d'une étoile devant la ligne qui va apparaître. Ou couper le Raspberry Pi pendant une heure, le rallumer et attendre qu'il se mette à l'heure.

```
ntpq -p
```

La solution a l'avantage de ne pas nécessiter une connexion à Internet mais il faut par contre un récepteur GPS avec évidemment une ligne de vue vers le ciel, donc dans les environs d'une fenêtre. Il est possible de faire la même chose en Bluetooth en plaçant le récepteur à l'extérieur, et bien évidemment de garder l'accès à un serveur NTP. Techniquement, il est

aussi possible de fournir l'heure aux autres machines de votre réseau, mais ce n'est pas forcément l'idéal avec tous les GPS : il faut idéalement un modèle avec une prise PPS pour calculer le délai au niveau de la réception et obtenir une base de temps totalement juste, ce qui n'est pas le cas des modèles classiques. C'est par contre parfaitement suffisant pour la majorité des projets, qui ne demandent pas une précision de l'ordre de la milliseconde.