

**Rapport GreenIT**

**Mesure de la consommation énergétique d’un Raspberry**

**Réalisé par:**

Ismatou Barry

Florent Baudon

Nicolas Lhomme

**Table de matière**

[1-Introductio](#h.i35rhmb1bg02)n

2- [Mesure directe](#h.ved2gdhswgto)

3-[Mesure indirecte](#h.sxlc68kpgah3)

4-[Conclusion](#h.vrkq5e9p7u3v)

# **Introduction**

Le Green IT est un concept qui vise à réduire l'empreinte écologique, économique, et sociale des [technologies de l'information et de la communication](https://fr.wikipedia.org/wiki/Technologies_de_l%27information_et_de_la_communication) (TIC). Il s'agit d'une manière globale et cohérente de réduire les nuisances rencontrées dans le domaine des équipements informatiques et ce, durant l'ensemble de la durée de vie de chaque équipement : soit aux différents stades de fabrication, d'utilisation (consommation d'énergie) et de fin de vie (gestion/récupération des déchets, pollution, épuisement des ressources non renouvelables). Ce concept s'inscrit plus largement dans la notion d'informatique éco-responsable » ou [développement durable](https://fr.wikipedia.org/wiki/D%C3%A9veloppement_durable).

Le green IT ou green computing, a pour objectif de réduire l'empreinte carbone générée par les Systèmes d'Information des entreprises tout en leur permettant de réaliser des économies.

Le green-IT, selon la définition du Journal officiel de la République Française du 12 juillet 2009, les écotechniques de l'information et de la communication, en abrégé éco-TIC, sont des techniques de l'information et de la communication dont la conception ou l'emploi permettent de réduire les effets négatifs de l'activité humaine sur l'environnement.



Avoir une politique GreenIT répond à plusieurs problématiques clés auxquelles les entreprises doivent faire face aujourd'hui.

On a plusieurs mesures qui sont classées en deux:

* Les mesures directes
* Les mesures indirectes

Dans un premier temps on parlera des mesures directes puis les mesures indirectes et enfin une conclusion

# **Mesure directe**

Dans un premier temps nous avions imaginés brancher un wattmètre entre le Raspberry Pi et son alimentation.



C’est très facile à mettre en place il suffit d’une simple électrique avec un Wattmètre intégrer pour afficher la consommation (watts, watts/h, etc…).

Bien que cette solution soit fonctionnelle, elle exige dans le cas d’une mesure de consommation sur toute une journée (ou même une heure), la présence d’une personne à côté de du wattmètre et prenne une mesure tout les *x* temps afin de pouvoir calculer la moyenne de consommation. Ainsi, en plus d’être physiquement contraignante, cette solution est coûteuse. Entre 15 et 20 euros la prise-wattmètre, le budget augmenterai très vite si on souhaite par la suite étudier la consommation de plusieurs appareils. Cette solution n’est donc pas viable à grande échelle, surtout si l’on souhaite automatiser la procédure.

# **Mesure indirecte**

Nous voulons créer un système plus intelligent où le simple lancement d’un script informatique (bash, perl ou python) nous permettrait de récupérer cette moyenne.

Pour faire cela nous avons donc imaginé le montage suivant :



Le raspberry pi étant alimenté par sa prise USB, l’ordinateur va fournir l’alimentation électrique au raspberry pi grâce à l’un de ces port USB. L’ordinateur va périodiquement mesurer la quantité d’énergie qu’il envoie à travers ce port, et le stocker dans un fichier texte.

Nous partons du principe que l’ordinateur et sous un système Unix. Sous Unix il existe un programme nommée “lsusb” qui permet de lister les propriétés de chaque port et notamment l’intensité consommée en *mA*.

Lsusb liste beaucoup de propriétés, afin de n’avoir que les renseignement qui nous intéresse nous utiliserons un commande *grep* pour ne récupérer que la consommation énergétique, voici donc la commande qui sera utilisée pour mesurer la consommation d’un port USB :

**lsusb -v|egrep "^Bus|MaxPower"**

Nous avons effectué un test pour vérifier que cette commande nous retournait bien une valeur. A notre grande surprise, parmi toute les données envoyées par la commande *lsusb*, le Raspberry Pi ne figurait pas dans la liste des périphériques USB.

Après plusieurs recherches nous avons découvert que pour récupérer des informations sur le port USB du Raspberry Pi, il fallait que celui ci soit en mode OTG (On The Go), l’OTG permet aux périphériques de s'échanger des informations entre eux sans ordinateurs. Seulement, d’après nos recherches sur internet seul les Raspberry Pi A peuvent se mettre en OTG, les autres de disposent pas des connections nécessaires pour faire de l’OTG.

Ne voulant pas abandonner cette solution, nous avons trouvé une idée :Ajouter un hub USB entre l’ordinateur et le Raspberry Pi. Un hub USB est un périphérique avec un mode OTG. Nous pouvons donc mesurer sa consommation électrique. Si nous branchons l’alimentation du Raspberry Pi sur le hub nous pourrons mesurer sur l'ordinateur la consommation du hub **et** du Raspberry Pi.



Comme dit précédemment, le hub USB consomme lui aussi de l'énergie. Il faut donc en tenir compte. Nous pensons que le hub USB à lui seul consomme toujours la même quantité d’énergie, et que les variations de consommation sont celles induites par les périphériques branchés au hub.

Pour que nos mesures ne soient pas faussées nous prévoyons de prendre plusieurs mesures de la consommation électrique du hub seul (afin de voir si la consommation varie dans le temps). Ensuite nous mesurerons la consommation du hub avec un Raspberry Pi branché dessus. Sur toutes les prises de mesures nous soustrairons la consommation du hub afin de ne garder que la consommation du Raspberry Pi (nous savons grâce à la loi des mailles que *I1+2 = I1 + I2*).

Avec l’intensité nous pouvons donc mesurer la puissance énergétique grâce à la formule P=U.I, qui est applicable car nous somme en courant continue. La tension sur un Raspberry est constante : 12 Volt.

# **Conclusion**

Facile à mettre en place, faire une mesure directement avec une prise avec wattmètre intégré est ce qu’il y a de plus évident. Mais comme exprimé précédemment, cette méthode est trop contraignante physiquement et surtout coûteuse.

En revanche, le hub usb est un matériel très peu cher (quelques euros seulement) et nous permet d’avoir ces mesures via son mode OTG sur un ordinateur. Ces mesures peuvent être automatisées via un script, qui va également se charger d’établir des statistiques détaillées. Plus efficace et moins cher, c’est cette solution que nous choisirons.

# 