

# Tuto wattmètre

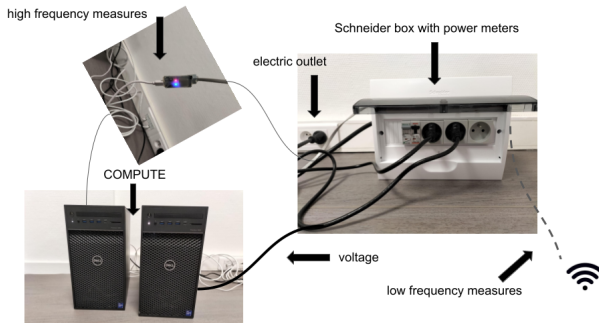
Green AI

Green AI UPPA & Prof En Poche

1 Février 2022



# Consommation réelle avec wattmètres



**Figure:** Le Wattmètre à Green AI

2 manières de récupérer les mesures :

- en **Wifi** : mesures à basse fréquence (1 sec, 3 sec, 1 min, 10 min, 2 heures)
- en **USB** : mesures haute fréquence (10 ms)

# Consommation réelle avec wattmètres

- en Wifi : se connecter à *mv\_esp\_02xxxxxx* et un dashboard s'ouvre (sinon se connecter à *www.mv.fr*):

Home

Download

Set Time

Wifi/Ftp

⌂

SN: 02000012, tri v822 06/07/21 10:42:20.14

☒ Auto-Refresh

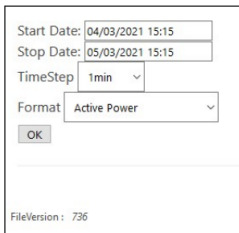
SN	C	Name	Phase 1,2,3	Inv	Values			
02000012	1	U1			244.52V	50.01Hz		
02000012	2	U2			237.64V			
02000012	3	U3			237.65V			
02000012	1	Voie1	● ○ ○	<input checked="" type="checkbox"/>	62.5W	0.58A	-77.93VAR	141.57VA
02000012	2	Voie2	○ ● ○	<input checked="" type="checkbox"/>	4.1W	0.08A	-10.35VAR	19.31VA
02000012	3	Voie3	● ○ ○	<input checked="" type="checkbox"/>	54.9W	0.57A	-79.16VAR	138.14VA

**Figure:** Mesure en temps réel avec rafraichissement toutes les 2 secondes

Les mesures :

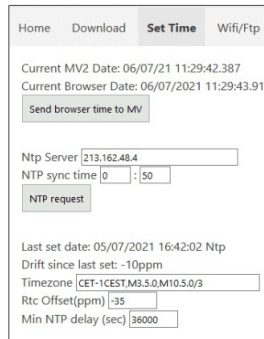
- Tension RMS par phase en V,
- Puissance active en W,
- Courant en A
- Puissance réactive en VAR
- Puissance apparente en VA

# Consommation réelle avec wattmètres



Start Date: 04/03/2021 15:15  
Stop Date: 05/03/2021 15:15  
TimeStep: 1min  
Format: Active Power  
OK  
FileVersion: 736

(a) Télécharger les mesures



Home Download **Set Time** Wifi/Ftp  
Current MV2 Date: 06/07/21 11:29:42.387  
Current Browser Date: 06/07/2021 11:29:43.91  
Send browser time to MV  
Ntp Server: 213.162.48.4  
NTP sync time: 0 : 50  
NTP request  
Last set date: 05/07/2021 16:42:02 Ntp  
Drift since last set: -10ppm  
Timezone: CET-1CEST,M3.5.0,M10.5.0/3  
Rtc Offset(ppm): -35  
Min NTP delay (sec): 36000

(b) Réglage de l'horloge

Télécharger au format *tsv*, 4 fichiers possibles : Active Power, Full etc...

# Consommation réelle avec wattmètres

- en USB, outil nécessaire : gcc + télécharger une archive . Il faut ensuite exécuter (en se plaçant dans le dossier dézippé :

```
gcc -std=c11 -o wattmetre-read wattmetre__main.c wattmetre__lib.c -lm
```

Puis pour chaque mesure individuelle :

## Remplissage d'un fichier logfile

```
./wattmetre-read -tty=/dev/ttyUSB0 -nb=6 > logfile
```

**Remarque :** /dev/ttyUSB0 pas toujours accessible :

```
sudo chmod 666 /dev/ttyUSB0
```

# Consommation réelle avec wattmètres

	#timestamp	#frame_is_ok	#U1	#U2	#U3	#current1	#activepow1	#current2	#activepow2	#current3	#activepow3	#current4	#activepow4
0	1.640267e+09	True	225.0	0.0	0.0	0.0	-0.1	0.0	0.0	0.3	58.3	0.5	85.7
1	1.640267e+09	True	225.7	0.0	0.0	0.0	-0.1	0.0	0.0	0.3	58.7	0.5	86.2
2	1.640267e+09	True	225.7	0.0	0.0	0.0	-0.1	0.0	0.0	0.3	58.7	0.5	86.2
3	1.640267e+09	True	225.0	0.0	0.0	0.0	-0.1	0.0	0.0	0.3	57.1	0.4	84.3
4	1.640267e+09	True	225.8	0.0	0.0	0.0	-0.1	0.0	0.0	0.3	57.4	0.5	84.8
...	...	...	...	...	...	...	...	...	...	...	...	...	...

**Figure:** Mesures hautes fréquences

⚠ Timestamp : Décalage d'une heure entre ordi et wattmètres ! ⚠

⚠ Vérifier le numéro de la prise pour le choix d' #activepow{1,2,...}  
(puissance active) ⚠

#current{1,2,...}  $\Rightarrow$  courant rms sur les voies 1 à 5 (current6 toujours à 0)

## Présence de valeurs négatives

Excedant d'énergie stockée en par l'alimentation et qui retourne vers le réseau, notamment lors des transitoires (fortes variations d'appel de puissance).

Installation à Hélioparc (et Prof en poche) :

Capteur USB haute fréquence sur Arriel !

Si Ansabère, lancer la commande `gcc...` sur Arriel puis lancer son code sur Ansabère (penser à bien récupérer le timestamp de départ de mesure).

Automatiser la mesure avec `os` ou `subprocess` pour ne pas lancer manuellement.

# Etude comparative entre AIPowerMeter et les Wattmètres

Démo notebook sur la doc.