

## API MaxPV! Version 3.0

Pour plus d'informations :

<https://github.com/Jetblack31/MaxPV>

L'accès à l'API se fait par **requête http GET**.

Une requête de type : `http://adresseIP:port/api/get?data=XX`

renverra une réponse : `YYY` où `YYY` est la valeur du paramètre `XX`.

**La réponse se fait donc en texte brut (plain text) sans formatage.**

Par exemple : la requête `http://adresseIP:port/api/get?data=01`

renverra `230` car `data=01` correspond à la lecture de la tension secteur.

`adresseIP` correspond à l'adresse IP de votre système MaxPV!

Port correspond au port http (par défaut 80)

`/api/get?data=XX`      lecture data de fonctionnement

### XX

```
01 : tension secteur
02 : courant mesuré par la pince ampèremétrique
03 : puissance active
04 : puissance apparente
05 : puissance routée
06 : puissance importée
07 : puissance exportée
08 : cosinus phi
09 : index du compteur d'énergie routée
10 : index du compteur d'énergie importée
11 : index du compteur d'énergie exportée
12 : index du compteur d'énergie produite (compteur à impulsion)
13 : puissance produite (lié au compteur à impulsion)
14 : mode de fonctionnement du SSR (0 = STOP, 1 = FORCE, 9 = AUTO)
15 : mode de fonctionnement du relais (0 = STOP, 1 = FORCE, 9 = AUTO)
16 : délai minimal de déclenchement du SSR (ms)
17 : délai moyen de déclenchement du SSR (ms)
18 : délai maximal de déclenchement du SSR (ms)
19 : tension de point milieu (Vbias)
20 : byte de statut/erreurs (en binaire)
    // Signification des bits
    (0 = LSB bit de droite, 7 = MSB bit de gauche)
        // bits 0..3 : informations
        bit 0: Routage en cours
        bit 1: Commande de routage à 100 %
        bit 2: Relais secondaire de délestage activé
        bit 3: Exportation d'énergie
        // bits 4..7 : erreurs
        bit 4: Anomalie signaux analogiques : ADC I/V overflow, biasOffset
        bit 5: Anomalie taux d'acquisition
        bit 6: Anomalie furtive Détection passage à 0 (bruit sur le signal)
        bit 7: Anomalie majeure Détection passage à 0
21 : durée de fonctionnement du routeur (jjj:hh:mm:ss)
22 : nombre d'échantillons par secondes traités
```

Exemple : `http://adresseIP:port/api/get?data=05`

**/api/get?alldata**      lecture de toutes les data de fonctionnement ci-dessus, dans l'ordre, et séparées par une virgule

Exemple :    `http://adresseIP:port/api/get?alldata`

**/api/get?param=XX**      lecture des paramètres du routeur (Arduino Nano)

**XX**

01 : V\_CALIB  
02 : P\_CALIB  
03 : PHASE\_CALIB  
04 : P\_OFFSET  
05 : P\_RESISTANCE  
06 : P\_MARGIN  
07 : GAIN\_P  
08 : GAIN\_I  
09 : E\_RESERVE  
10 : P\_DIV2\_ACTIVE  
11 : P\_DIV2\_IDLE  
12 : T\_DIV2\_ON  
13 : T\_DIV2\_OFF  
14 : T\_DIV2\_TC  
15 : CNT\_CALIB    → calibrage du compteur à impulsion en Wh/impulsion  
16 : P\_INSTALLPV → puissance de l'installation photovoltaïque en Wc

Exemple :    `http://adresseIP:port/api/get?param=10`

**/api/get?allparam**      lecture de tous les paramètres de configuration ci-dessus, dans l'ordre, et séparées par une virgule

Exemple :    `http://adresseIP:port/api/get?allparam`

**/api/get?version**      version du code arduino

**/api/get?versionmaxpv**      version du code WEMOS/ESP

**/api/get?versionweb**      version du site web

**/api/get?configmaxpv**      configuration WEMOS/ESP **au format JSON**

**/api/get?relaystate**      état du relais (STOP, FORCE, ON, OFF)

**/api/get?ssrstate**      état du SSR (STOP, FORCE, MAX, ON, OFF)

**/api/get?ping**      état du routeur Arduino (running, offline)

**/api/get?time**      heure solaire française (hh:mm:ss)

Exemple :    `http://adresseIP:port/api/get?version`

`/api/set?param=XX&value=YY`      changement du paramètre XX du routeur à la valeur YY

**XX**

01 : V\_CALIB  
02 : P\_CALIB  
03 : PHASE\_CALIB  
04 : P\_OFFSET  
05 : P\_RESISTANCE  
06 : P\_MARGIN  
07 : GAIN\_P  
08 : GAIN\_I  
09 : E\_RESERVE  
10 : P\_DIV2\_ACTIVE  
11 : P\_DIV2\_IDLE  
12 : T\_DIV2\_ON  
13 : T\_DIV2\_OFF  
14 : T\_DIV2\_TC  
15 : CNT\_CALIB      → calibrage du compteur à impulsion en Wh/impulsion  
16 : P\_INSTALLPV → puissance de l'installation photovoltaïque en Wc

Exemple : `http://adresseIP:port/api/set?param=02&value=0.115`

*Note : le point décimal pour YY est indifféremment '.' ou ','*

`/api/set?relaymode&value=yyyy`      changement du mode de fonctionnement du relais

**yyyy**

stop :      force l'arrêt du relais  
force :      force la marche du relais  
auto :      mode automatique (mode normal)

Exemple : `http://adresseIP:port/api/set?relaymode&value=force`

`/api/set?ssrmode&value=yyyy`      changement du mode de fonctionnement du SSR

**yyyy**

stop :      force l'arrêt du relais  
force :      force la marche du relais  
auto :      mode automatique (mode normal, routage)

Exemple : `http://adresseIP:port/api/set?ssrmode&value=auto`

`/api/set?configmaxpv&value=jsonString`      changement de la configuration WEMOS/ESP (**format JSON**)

Exemple : `http://adresseIP:port/api/set?configmaxpv&value={"ip":"192.168.2.100","gateway":"192.168.2.1","subnet":"255.255.255.0","dns1":"192.168.2.1","dns2":"8.8.8.8","http_port":8080}`

**/api/action?xxxx**

réalise l'action xxxx

**xxxx**

resetindex : remise à 0 des index (dans l'Arduino)  
saveindex : enregistrement des index (dans l'Arduino)  
saveparam : enregistrement des paramètres en EEPPROM (dans l'Arduino)  
loadparam : chargement des paramètres de l'EEPPROM (dans l'Arduino)  
format : formatage de l'EEPPROM de l'Arduino  
restart : redémarrage de l'Arduino / routeur EcoPV  
eraseconfigesp : suppression du fichier de configuration de l'ESP / interface WEB MaxPV!. Cela force le prochain démarrage en mode point d'accès pour se connecter à un autre réseau Wifi par le portail captif.  
rebootesp : redémarrage de l'ESP / interface WEB MaxPV!

Exemple : <http://adresseIP:port/api/action?saveindex>