



# AmpVolt : module de mesure de puissance (1)

mesurer la puissance CC et la consommation énergétique jusqu'à 50 V et 5 A

Saad Intiaz (Elektor)

Pour la surveillance de l'énergie et du courant via des ports USB, il existe des modules peu coûteux adaptés. De même, de nombreux appareils permettent de mesurer la consommation énergétique du réseau CA. Cependant, les options se raréfient pour les tensions continues supérieures à 12 V et surtout supérieures à 24 V. Lorsque l'on pense aux petits systèmes solaires et aux vélos électriques, cette plage de tension devient de plus en plus intéressante, et c'est là que notre projet entre en jeu. Optant pour une flexibilité maximale, nous avons choisi une approche modulaire. Dans cette première partie de la série AmpVolt, nous vous présentons une petite carte permettant de mesurer des courants et des tensions jusqu'à 50 V et 5 A, qui peut être connectée au microcontrôleur de votre choix.

When developing the AmpVolt module, my aim was to combine precision, flexibility, and modularity. The core components — a precision INA169 current sensor, a voltage divider Lors du développement du module AmpVolt, mon objectif était de combiner précision, flexibilité et modularité. Pour cela, j'ai sélectionné les composants de base - un capteur de courant de haute précision INA169, un diviseur de tension et un module CAN (ADC) 12 bits ADS1015. Ces composants fonctionnent ensemble pour fournir des mesures précises de courant et de tension sur une vaste gamme.

Le module présenté ici est conçu pour être connecté à tout type de microcontrôleur alimenté en 3,3 V ou 5 V avec une interface I<sup>2</sup>C. Afin de renforcer la polyvalence du module, nous y avons également ajouté un écran OLED, contrôlé via le même bus I<sup>2</sup>C. Pour plus d'options de connectivité, nous avons intégré trois connecteurs pour les signaux I<sup>2</sup>C sur le circuit imprimé : un connecteur Grove et un connecteur Qwiic au pas de 2 mm et un connecteur JST XH au pas de 2,54 mm. Envisageant l'intégration avec les systèmes de batteries usuels, nous avons utilisé des connecteurs XT60PW pour la connexion à la charge et à la source. La **figure 1** représente le schéma fonctionnel du projet AmpVolt, où vous pouvez observer le module de mesure ainsi que le(s) microcontrôleur(s) connecté(s). L'INA169 et l'ADS1015 sont tous deux des produits de Texas Instruments [1][2]. Pour ce projet, nous avons utilisé des cartes de connexion fabriquées en Chine (BoB) pour connecter ces deux puces [3][4]. Il existe plusieurs cartes BoB compatibles avec l'ADS1015 avec des brochages identiques et qui sont également adaptées au projet AmpVolt. Contrairement aux capteurs de courant à effet

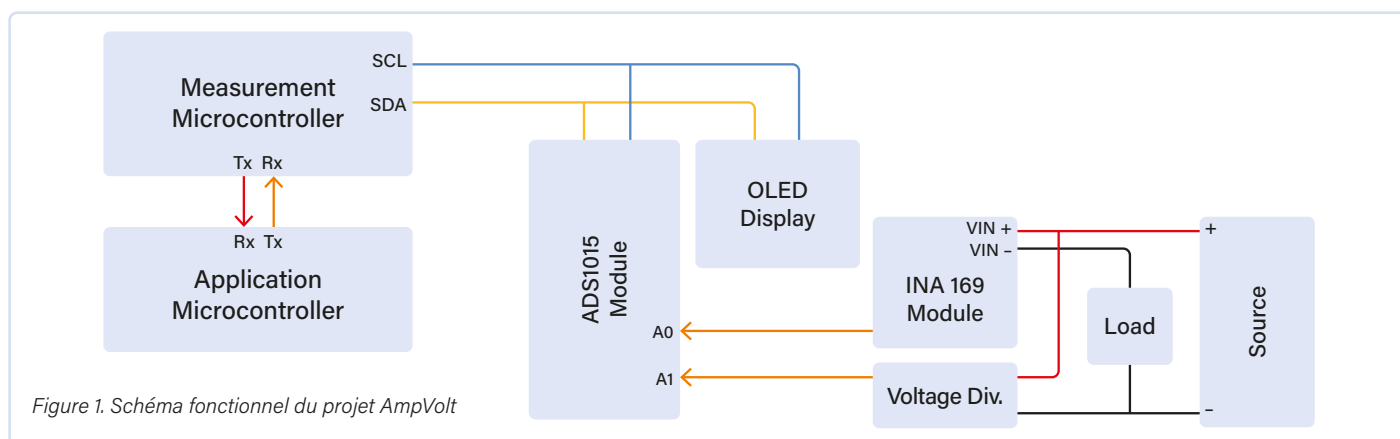


Figure 1. Schéma fonctionnel du projet AmpVolt

Hall qui sont moins précis que les résistances shunt à faible courant, le module INA169 utilise une méthode de mesure du courant par résistance shunt. Ces modules sont peu coûteux et largement disponibles sur le marché.

Le choix du module ADC ADS1015 a été crucial pour ce projet en raison de sa précision de 12 bits et son gain programmable qui permettent d'obtenir des mesures très précises. Il est important de noter que tous les microcontrôleurs ne disposent pas de CAN interne avec une précision de 12 bits, et certains, comme l'ESP32 d'Espressif, peuvent présenter des problèmes de tension de référence et de linéarité. Cette incohérence entre différents microcontrôleurs souligne la nécessité d'avoir un CAN externe fiable. L'ADC ADS1015 a donc été choisi pour sa précision de 12 bits, offrant un bon équilibre entre précision, vitesse et efficacité énergétique adaptés à une large variété d'applications. Bien qu'un ADC ADS1115 de 16 bits puisse offrir une résolution plus élevée, le choix de l'ADS1015 s'explique principalement par son taux d'échantillonnage plus élevé.

## Schéma

En bref, le schéma (figure 2) illustre les connexions entre les cartes BoB INA169 et ADS1015 référencées respectivement comme K1 et K2, le diviseur de tension, et l'écran OLED. Comme le module INA169 fournit 1 V par 1 A, nous avons utilisé un diviseur de tension constitué des résistances R3 et R4 pour s'adapter aux systèmes de 3,3 V ; mais, s'il doit être utilisé avec un contrôleur de 5 V si l'usage avec un microcontrôleur de 5 V est requis, le cavalier JP1 peut être court-circuité pour bypasser le diviseur de tension. Pour la mesure de tension, nous avons utilisé un diviseur de tension formé par les résistances R1 de 100 kΩ et R2 de 6,7 kΩ. Ce circuit permet de mesurer des tensions jusqu'à 50 V, couvrant largement les besoins de la plupart des applications en courant continu.

En ce qui concerne la connectivité, il était essentiel de faciliter l'intégration du module dans divers systèmes, ce qui a été une priorité absolue. Ainsi, le module est équipé d'un connecteur Qwiic en K3 permettant des connexions *plug-and-play* rapides, un connecteur Grove en K4 pour une intégration transparente avec l'écosystème de Seed

## Caractéristiques techniques

- Tension d'entrée (pour les modules) : 3,3 V ou 5 V, selon la carte microcontrôleur.
- Tension d'entrée (source) : 50 V CC max.
- Charge de courant max : 5 A.
- Diviseur de tension pour la mesure de la tension de la source.
- Module INA169 pour la mesure du courant
- Diviseur de tension pour le module INA169 pour les microcontrôleurs 3,3 V.
- Module ADC ADS1015 pour une précision de 12 bits.
- Écran OLED pour afficher les mesures de puissance en temps réel.

Studio, et un connecteur au pas de 2,54 mm pour des configurations personnalisés. L'ajout des connecteurs XT60PW K6 et K12 pour les connexions de la batterie et de la charge reflète la fiabilité et la facilité d'utilisation dans les environnements exigeant une haute puissance. Le connecteur K7 est utilisé si l'on n'utilise pas l'ADC ADS1015, de sorte que la sortie non traitée des capteurs pour un échantillonnage avec un autre ADC, tel qu'un ADC interne dans votre microcontrôleur. Le connecteur K8 permet d'exploiter pleinement l'ADS1015 lorsqu'il est nécessaire de mesurer une autre tension analogique.

Les connecteurs K9 et K10 sont dédiés à l'écran OLED. Vous vous demandez peut-être : pourquoi deux connecteurs ? Nous voulions offrir une flexibilité maximale. Selon les modules d'affichage OLED, la broche de 3,3 V se trouve sur la broche 1 pour certains, tandis que pour d'autres, elle est sur la broche 2.

La tension d'entrée, qu'elle provienne d'une batterie ou d'une autre source au connecteur K12, ne doit pas dépasser 50 V, compte tenu de la configuration actuelle du diviseur de tension formé par les résistances R1 de 100 kΩ et R2 de 6,7 kΩ. Pour plus de flexibilité, nous avons ajouté une résistance variable RV1 pour régler le diviseur de tension en fonction des exigences de votre application. De plus, la charge maximale de courant ne doit pas dépasser 5 A - la limite maximale de la carte d'expansion INA169. Vous trouverez un résumé des caractéristiques techniques dans l'encadré ci-dessus.

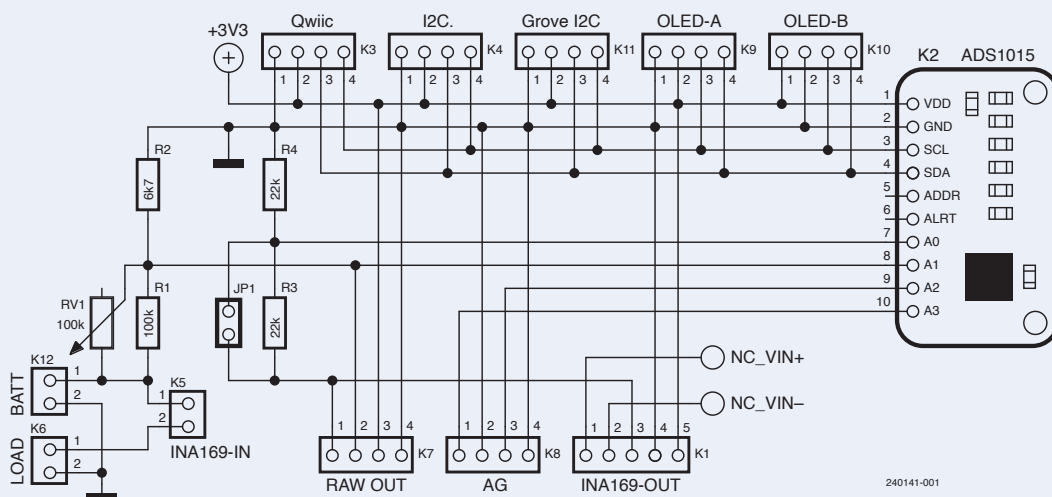


Figure 2. Schéma du module AmpVolt.





La puissance (W) exprimée en watts (W), est calculée en multipliant la tension (V) par le courant (I), selon la formule  $V \times I$ . L'intervalle de temps correspondant à la durée entre deux mesures successives de l'énergie est enregistré en secondes, puis converti en heures en divisant par 3 600 (nombre de secondes dans une heure). Cette méthode permet d'intégrer efficacement la puissance dans le temps pour calculer la consommation totale d'énergie.

Dans la fonction loop, l'énergie consommée est mise à jour en ajoutant la puissance calculée pour chaque intervalle (toutes les secondes, dans ce cas) à un total cumulatif. Cette addition continue fournit une mesure dynamique de l'énergie utilisée au fil du temps, ce qui permet au système de suivre et de signaler avec précision la consommation d'énergie au fur et à mesure que le module fonctionne.

Le calcul de l'état de charge (SoC) permet d'évaluer l'autonomie de la batterie et de contrôler l'efficacité de son utilisation. Il est déterminé par le rapport entre l'énergie consommée et la capacité totale de la batterie, exprimé en pourcentage. La formule utilisée est la suivante :

$$\text{State of Charge (SoC)} = \left( \frac{\text{Energy Consumed}}{\text{Battery Capacity}} \right) \times 100\%$$

L'«énergie consommée» est l'énergie totale accumulée au fil du temps, exprimée en Wh, et la «capacité de la batterie» est la capacité énergétique totale de la batterie, également exprimée en Wh. Ce calcul donne un aperçu en temps réel de la quantité d'énergie consommée par la batterie par rapport à sa capacité totale, ce qui donne une estimation importante de la durée de vie restante de la batterie et indique le moment optimal pour procéder.

## Taux d'échantillonnage

L'ADC ADS1015 peut atteindre, selon sa fiche technique, une fréquence d'échantillonnage maximale de 3.300 échantillons par seconde. Cependant, lorsque cet ADC est utilisé dans l'environnement Arduino, ce taux est sensiblement réduit. En effet, en mode de test single-shot, le taux d'échantillonnage se réduit à 312 échantillons/s, et en mode continu, il peut atteindre jusqu'à 1500 échantillons/s. Cette diminution est due à la couche d'abstraction matérielle (HAL) d'Arduino qui, bien qu'elle facilite l'abstraction et la simplicité du code, impose une restriction significative sur la vitesse de traitement de l'ESP32.

Au cours des phases de test, nous avons fait des tentatives pour améliorer le taux d'échantillonnage en ajustant la vitesse I<sup>2</sup>C de la norme 10 kHz à 90 kHz. Ce réglage a permis d'obtenir une augmentation modeste d'environ 20 à 30 échantillons/s en mode single-shot et de 100 échantillons/s en mode continu. Ce résultat met en évidence les limites imposées par le HAL du Framework Arduino sur les capacités de performance de l'ESP32.

Il convient de noter que l'utilisation de l'ESP32 dans son environnement natif, en particulier l'ESP-IDF, pourrait considérablement améliorer l'efficacité de la fréquence d'échantillonnage. L'ESP-IDF permet un contrôle plus direct sur le matériel, ce qui peut permettre de libérer toutes les capacités d'échantillonnage de l'ADS1015.

Néanmoins, pour répondre aux besoins d'un public plus large et assurer l'accessibilité, notre première implémentation de micrologiciel décrite ici reste dans le cadre d'Arduino. Cette démarche vise à trouver un équilibre entre performance et simplicité. Cependant, pour ceux qui recherchent les meilleures performances de mesure tout en bénéficiant de la facilité de programmation offerte par Arduino et ses



## Liste des composants

### Résistances

R1 = 100 kΩ

R2 = 6,7 kΩ

RV1 = 100 kΩ trimmer multi-tour ou selon les besoins

R3,R4 = 22 kΩ

### Modules

INA169 Current Sensor Breakout Board [2]

ADS1015 Breakout Board [3]

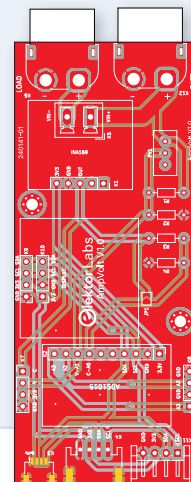
Module d'affichage OLED 128x64 0,96 pouce [4]

### Connecteurs

2x AMASS XT60 PW

Connecteur Grove

Connecteur Qwiic



## Aperçu du croquis Arduino

### Variables de configuration et variables globales

- Initialise la communication série et I2C avec des broches SDA et SCL spécifiques.
- Configure le module ADC ADS1015 pour la lecture des entrées
- Définit des constantes pour les valeurs de résistance des diviseurs de tension et de courant, permettant des mesures précises de la tension et du courant.
- Établit des variables pour le calcul de l'énergie consommée au cours du temps et le suivi de l'heure de début pour les calculs du temps écoulé.

### Principales fonctions

- `readVoltage()` : Lit et calcule la tension en fonction des valeurs de l'ADC et du rapport du diviseur de tension, en tenant compte de tout offset..
- `readCurrent()` : similaire à `readVoltage()`, mais calcule le courant en utilisant les valeurs de l'ADC et le rapport du diviseur de courant.
- `sendData()` : envoie des données (tension, courant ou puissance) par communication série en fonction de la commande reçue.
- `calculateAndSendAdditionalData()` : calcule des données supplémentaires telles que la puissance et l'état de charge, puis les envoie au format JSON via une liaison série.
- `readCommand()` : lit les commandes du port série et renvoie une commande complète lorsqu'un caractère de retour à la ligne est détecté.

### fonctions utilitaires

- `updateDisplay()` : actualise l'écran OLED avec la tension, le courant, la puissance et d'autres données pertinentes.
- `elapsedTimeAsString()` : calcule le temps écoulé depuis le démarrage du module et le formate sous forme de chaîne de caractères.

nombreuses bibliothèques, une option intéressante consiste à utiliser deux microcontrôleurs.

## Interface série

Dans le micrologiciel développé par Elektor, nous avons déjà intégré une interface simple pour envoyer les données à un autre «contrôleur d'application» (ou vers un PC via un pont série-USB). Le contrôleur d'application (qui pourrait être le contrôleur principal d'un projet IoT) est ainsi déchargé de la tâche d'échantillonnage et de tout calcul de puissance/énergie. Cela se fait dans le «microcontrôleur de mesure». La fonction `sendData()` gère la communication série en répondant aux commandes spécifiques du contrôleur d'application reçues via le port série. Lorsqu'une commande est reçue, la fonction détermine le type de données demandées - qu'il s'agisse de la tension (`#v`), du courant (`#i`) ou de la puissance (`#p`) - et effectue ensuite la mesure nécessaire en appelant `readVoltage()`, `readCurrent()` ou en calculant la puissance directement dans la fonction.

La fonction `calculateAndSendAdditionalData()` joue un rôle crucial en procédant à des calculs de mesures supplémentaires telles que la puissance et l'état de charge (SoC) d'une batterie. Elle prépare ensuite ces données dans une structure JSON pour la transmission en série. Voici une description étape par étape de son processus :

- **Calcul de la puissance** : cette étape implique l'appel `readVoltage()` et `readCurrent()` pour obtenir les mesures actuelles de tension et de courant. La puissance est ensuite calculée en multipliant ces deux valeurs.
- **Calcul de l'état de charge (SoC)** : le SoC est une évaluation de la capacité résiduelle de la batterie. Il est calculé en intégrant la puissance dans le temps pour obtenir l'énergie consommée. Ensuite, cette valeur est comparée à la capacité totale de la batterie, prédéfinie dans le code (`batteryCapacityWh`). Le SoC est exprimé en pourcentage de la capacité totale
- **Formatage et transmission des données** : après avoir calculé ces mesures, la fonction les convertit en une chaîne JSON. Cette chaîne comprend la tension, le courant, la puissance, l'énergie consommée et le SoC, ce qui facilite l'analyse sur tout appareil recevant les données en série.

Le code utilise la fonction `millis()` pour suivre et mettre à jour périodiquement la consommation d'énergie, introduisant ainsi un aspect temps réel dans la surveillance de la consommation énergétique sur une durée étendue. Il comprend également une fonction `updateDisplay()`, qui sert à actualiser l'écran OLED avec les relevés actuels et d'autres informations pertinentes telles que le temps écoulé depuis le début

de la mesure, calculé grâce à la fonction `elapsedTimeAsString()`. Dans l'ensemble, le code [5] est simple. Il assure une collecte précise des mesures électriques, une transmission fluide de données et des mises à jour en temps réel fournies à l'utilisateur via d'un écran OLED. Vous trouverez un bref aperçu du code dans l'encadré.

## Améliorations futures

Le parcours d'amélioration du module AmpVolt prévoit plusieurs avancées techniques visant à augmenter la précision et la facilité d'utilisation. Une mise à jour notable consistera en l'intégration d'un mode d'étalonnage logiciel. Cette fonction devrait améliorer la précision des mesures de basse tension et de courant grâce à des techniques avancées d'ajustement de courbe, réduisant ainsi les erreurs de mesure de manière significative.

En outre, nous sommes en train de développer une fonction innovante qui permet d'alimenter le microcontrôleur directement à partir du connecteur source. Cette fonction simplifiera la configuration des projets en réduisant le nombre de composants nécessaires, rationalisant ainsi le processus d'intégration.

En outre, nous travaillons actuellement à la conception d'une carte d'extension personnalisée compatible avec la gamme de cartes de contrôleurs XIAO de Seeed Studio, afin d'élargir la polyvalence et la gamme d'applications du module. La prochaine version V2 de ce projet promet d'incorporer ces améliorations, offrant une précision et une commodité accrues dans les solutions de surveillance de l'énergie. ◀

240141-04

## Questions ou commentaires ?

Envoyez un courriel à l'auteur ([saad.imtiaz@elektor.com](mailto:saad.imtiaz@elektor.com)) ou contactez Elektor ([redaction@elektor.fr](mailto:redaction@elektor.fr)).



## Produits

- **Qoitech Otii Arc - Power Supply, Power Meter and Data Acquisition**  
[www.elektor.fr/19270](http://www.elektor.fr/19270)
- **Renewable Energy at Home**  
[www.elektor.fr/20747](http://www.elektor.fr/20747)
- **ESP Terminal**  
[www.elektor.fr/20526](http://www.elektor.fr/20526)

## LIENS

- [1] INA169 Current Sense Amplifier | Fiche technique : <https://www.ti.com/product/INA169>
- [2] ADS1015 12-bit ADC | Fiche technique : <https://www.ti.com/product/ADS1015>
- [3] GY-169 - INA169 Fiche technique : [https://s.click.aliexpress.com/e/\\_DFZSO21](https://s.click.aliexpress.com/e/_DFZSO21)
- [4] ADS1015 Module 12-bit ADC: [https://s.click.aliexpress.com/e/\\_DnMmvRJ](https://s.click.aliexpress.com/e/_DnMmvRJ)
- [5] AmpVolt V1.0 | Source code and PCB files: <https://github.com/ElektorLabs/AmpVolt>
- [6] 0.96" OLED Display (Blue, I2C, 4-Pin): <https://elektor.com/products/0-96-oled-display-blue-i2c-4-pin>