

assemblage du kit

CLOC 2.0 D'ELEKTOR

un produit Elektor déballé par Espressif

Jeroen Domburg (Espressif)

Le Cloc 2.0 d'Elektor est une horloge flexible avec alarme à base d'un ESP32-Pico-Kit d'Espressif. Il se présente sous la forme d'un kit de pièces à assembler soi-même. Dans cet article, nos amis d'Espressif en ont assemblé un et nous on fait part de leurs impressions.

Comment vous réveillez-vous le matin ? Avez-vous une alarme programmée sur votre téléphone, avez-vous encore un vieux radio-réveil à côté de votre lit, êtes-vous réveillé par le coq le matin ou laissez-vous simplement les rideaux ouverts pour que les premiers rayons du soleil atteignent votre visage ?

Vous pensez peut-être que toutes ces méthodes présentent des inconvénients. Votre téléphone ne vous permet pas vraiment d'ouvrir un œil pour voir combien de temps de sommeil il vous reste, et le radio-réveil n'a probablement qu'une seule alarme, qui vous réveillera également le matin pendant un week-end ou un jour férié si vous ne l'éteignez pas. Et bien que le coq et les rayons de soleil soient bon marché et généra-

lement fiables, il est assez difficile de les configurer pour une alarme. D'une manière générale, aucune de ces solutions n'est « véritablement » flexible.

Si vous souhaitez tout de même bénéficier de cette souplesse, l'option la plus simple est évidemment de créer votre propre réveil. Si vous le construisez vous-même, vous pouvez y intégrer toutes les fonctionnalités que vous souhaitez. Les horaires d'alarme hebdomadaires, la connectivité Wifi, les sons d'alarme amusants, l'augmentation de la température ambiante, la mise en marche de la machine à café, tout ce que vous souhaitez..

Découvrez Cloc 2.0

Mais que faire si vous n'avez pas les compétences, le temps ou la motivation de passer par tout le chemin de l'architecture, de la conception matérielle et logicielle, de la fabrication, etc. Elektor peut vous aider avec le Cloc 2.0. Ce modèle est doté d'un double affichage LED à 7 segments pour visualiser l'heure actuelle et l'alarme, d'une connexion Wifi, afin de toujours afficher l'heure exacte et de permettre de nombreuses options configurables. De plus, elle dispose d'un émetteur et d'un récepteur de télécommande IR qui vous permet d'allumer, par exemple, un système de sonorisation lorsque l'heure de l'alarme à sonné. Mieux encore, elle est basée sur un ESP32-Pico-Kit et le logiciel est open-source, ce qui signifie qu'il est facile d'ajouter une fonction s'il en manque une. Elektor a eu la gentillesse de nous envoyer un kit pour que nous puissions l'évaluer. Le projet complet de la Cloc 2.0 est publié sur le site d'Elektor Labs ici [1].

Kit

Le Cloc 2.0 nous a été livré en kit, et donc en pièces détachées (**figure 1**). Le plus étonnant était le boîtier rouge de la société Hammond, sur lequel était imprimé le logo d'Elektor Labs. Outre l'aspect esthétique, la combinaison d'un boîtier rouge et d'un afficheur à 7 segments à LED rouge est intéressante car elle améliore le contraste des LED, ce qui rend l'horloge plus facile à lire. Le reste des composants se trouvait dans plusieurs sachets, étiquetés avec les valeurs des



composants se trouvant à l'intérieur. Chose intéressante, chaque sachet contient un ensemble de composants différents et reconnaissables. Par exemple, aucun sachet ne contenait plus de deux valeurs de résistance, et il n'est donc pas nécessaire de regarder les codes de couleurs pour savoir de quelle résistance il s'agit. C'est une idée intelligente qui permet d'économiser du plastique tout en permettant simplement de trouver le bon composant. Bravo à l'équipe d'Elektor pour avoir imaginé ce système ! À noter que le kit ne comporte pas de boutons accessibles depuis l'extérieur du boîtier, l'idée étant que vous en avez peut-être encore quelques-uns à votre disposition quelque part.

Trouver la documentation

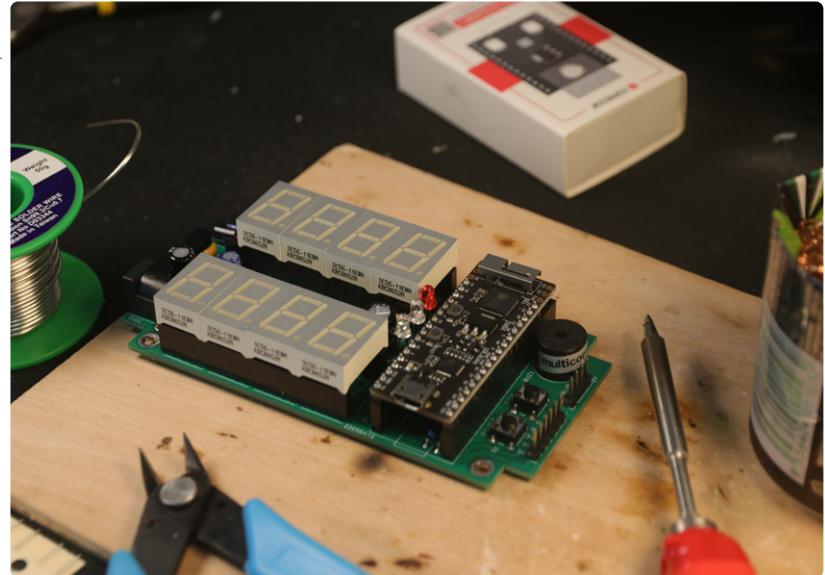
La recherche de la documentation nécessaire à la réalisation de ce projet aurait pu être un peu plus aisée. Le site d'Elektor comporte des pages web réparties sur les trois domaines (elektorlabs.fr, elektor.fr et elektor-magazine.fr) avec des téléchargements et des textes différents sur chaque page. C'est un peu déroutant, et il aurait été utile de trouver au moins un document imprimé détaillé, pour avoir tous les téléchargements et la documentation nécessaires concernant le kit. Cependant, une fois que nous avons tout trouvé, il s'est avéré très complet : l'article d'Elektor qui décrit le projet est inclus ainsi que tous les téléchargements, et même une vidéo [2] sur la meilleure façon d'assembler le tout.

Avec la documentation en main, le montage du circuit imprimé est un jeu d'enfant (**figure 2**). Tous les composants sont traversants, ce qui facilite le soudage. La vidéo propose un ordre de soudage des composants qui a très bien fonctionné pour nous.

Il faut remarquer que les connecteurs qui supportent les afficheurs LED à 7 segments sont du même type que ceux dans lesquels l'ESP32-Pico-Kit est inséré. C'est un peu un problème, car les broches des afficheurs LED sont plus fines et ne font pas toujours bon contact. Ceci est compréhensible, car il ne semble pas y avoir de connecteurs pour des broches plus petites disponibles à la même hauteur que celles-ci. Et de plus, tout ce qui est problématique ici est facilement résolu en pliant un peu les broches des afficheurs.

Programmation de l'ESP32-Pico-Kit

Une fois que la carte est assemblée, il fallait programmer l'ESP32-Pico-Kit. Il a fallu quelques étapes pour installer les bonnes bibliothèques et le support de la carte, mais cela n'a pas été trop compliqué non plus. Un tel processus aurait pu être simplifié en utilisant par exemple ESP-Launchpad [3], mais la configuration actuelle a l'avantage de construire le croquis à partir des sources. Cela signifie que si vous trouvez quelque chose que vous voulez changer dans le code, vous avez déjà tout ce qui est nécessaire pour reconstruire et flasher le micrologiciel.



Construction mécanique

En ce qui concerne le montage du circuit imprimé dans le boîtier, il y a un peu plus de flexibilité dans la façon dont vous voulez faire les choses. Elektor fournit un gabarit de perçage et quelques éléments de quincaillerie dans les kits récents, de sorte que vous pouvez le monter de la manière « standard ». Peut-être que vous souhaitez ou même devez modifier certaines choses afin de les installer selon vos souhaits. Par exemple, en fonction de la forme et de la taille des boutons que vous souhaitez utiliser, vous pouvez créer votre propre gabarit de perçage ici.

Alimentation

Dans notre cas, bien que nous soyons sûrs d'en avoir des dizaines quelque part, nous n'avons pas réussi à trouver un bloc de 5 V avec le bon connecteur jack dans la grande caisse contenant les diverses alimentations. Ce que nous avons trouvé, c'est un petit circuit imprimé avec un connecteur USB-C, que nous avons pu modifier pour qu'il délivre simplement 5 V avec les 500 mA ou plus qui sont nécessaires. Cette solution offre l'avantage supplémentaire de ne pas avoir à utiliser une alimentation dédiée pour notre Cloc. N'importe quelle alimentation USB-C et n'importe quel câble feront l'affaire. Évidemment, nous avons modifié les plans de perçage du boîtier en conséquence. Plutôt que de percer un trou pour le connecteur jack, nous avons découpé un peu de plastique à l'arrière du boîtier pour le connecteur USB-C. Le circuit imprimé USB peut alors être fixé à l'arrière du boîtier avec de la résine époxy pour le maintenir en place.

Percez avec soin

À ce propos, voici un bon conseil. Lorsque vous percez des trous et installez des éléments, veillez à garder à l'esprit l'ordre des choses et l'orientation du boîtier. Évidemment, en tant que professionnels

Figure 2. Le circuit imprimé étant entièrement traversant, le soudage a été un jeu d'enfant.

Figure 3. L'arrière du boîtier contient suffisamment d'espace. Les boutons et les câbles y trouvent leur place, mais vous pouvez également y glisser un circuit imprimé d'extension.

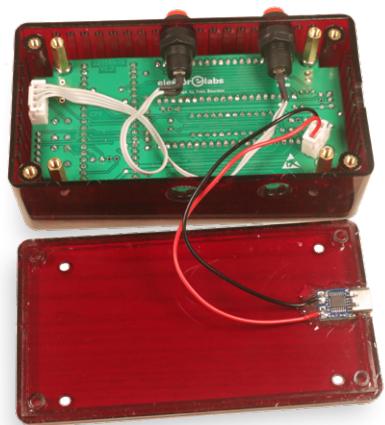




Figure 4. Notre horloge Cloc terminée, alimentée par un adaptateur USB-C.

chez Espressif, nous en sommes parfaitement conscients et nous pouvons donc vous assurer que... les trous de drainage... dans le fond de notre boîtier faisaient partie de nos plans initiaux depuis le début (**figure 3**).

Pour finir, nous avons tout assemblé proprement. Il nous restait quelques jolis boutons rouges assortis avec le boîtier que nous pouvions mettre en place. Plutôt que d'utiliser les connecteurs fournis, nous avons utilisé des connecteurs JST pour les connecter au circuit imprimé. Nous avons également utilisé un connecteur JST pour connecter la carte d'alimentation USB. De cette façon, nous pouvons sortir le circuit imprimé principal du boîtier au cas où il aurait besoin d'être modifié ou réparé.

Configuration de l'horloge Cloc

Une fois que tout est assemblé, la configuration de l'horloge Cloc s'est avéré très simple. Mettez-la sous tension, connectez-vous et allez à l'adresse IP indiquée, et vous pourrez la connecter au réseau Wifi local. A partir de là, chaque fois que le Cloc démarre, elle vous indiquera son adresse IP, ce qui est très bien car vous n'avez pas besoin de vous embêter avec des scanners réseau ou de regarder la configuration du routeur pour accéder à l'interface web. L'interface web elle-même est suffisamment simple pour ne pas avoir besoin d'un manuel, mais elle est très complète. Nous avons configuré les éléments de base comme le fuseau horaire et l'heure d'été en quelques instants seulement.

Dans son ensemble, nous aimons beaucoup l'horloge (**figure 4**). L'affichage LED est agréable et lisible, et le fait qu'il soit rouge signifie qu'il ne détruira pas votre vision nocturne si vous le regardez dans l'obscurité. L'horloge est très facilement configurable à l'aide de

l'interface web et si vous souhaitez la modifier pour vous réveiller comme vous le souhaitez, il y a moyen de le faire. À la fois au sens figuré avec l'accès facile grâce à l'utilisation de l'environnement Arduino, et au sens physique puisque le boîtier de l'horloge a de la place pour un circuit supplémentaire, par exemple derrière celui de le Cloc.

Suggestions pour une Cloc 3.0

Il y a quelques améliorations que nous aimerais voir pour une horloge Cloc V3, comme l'utilisation d'un connecteur USB comme alimentation (petite astuce : si le design utilisait par exemple un ESP32-S3, les broches d'E/S USB fournies pourraient être connectées au connecteur USB également, permettant la programmation et le débogage de l'horloge Cloc sans ouvrir le boîtier). De plus, l'ESP32 pourrait peut-être piloter un haut-parleur pour un son de réveil un peu moins rude que le signal sonore actuel. Nous connaissons l'émetteur IR qui peut allumer, par exemple, une radio, mais nous pensons que de moins en moins de gens en possèdent dans leur chambre à coucher de nos jours. D'un autre côté, si ce sont des choses dont vous avez vraiment, mais alors vraiment besoin, rien ne vous empêche de les intégrer dans la version actuelle du Cloc, étant donné sa conception et son logiciel ouverts. ↗

VF : Laurent Rauber — 230561-04

Questions ou commentaires ?

Contactez Elektor (redaction@elektor.fr).



Produits

➤ **Cloc 2.0 d'Elektor en kit**
www.elektor.fr/20438

➤ **ESP32-PICO-Kit V4**
www.elektor.fr/18423

LIENS

- [1] Le projet Cloc 2.0 sur Elektor Labs : <https://elektormagazine.fr/labs/cloc-le-reveil-20>
- [2] Horloge « maison » avec un ESP32 - guide d'assemblage du Cloc 2.0 d'Elektor : <https://youtu.be/9VSLdFz6jyl>
- [3] ESP-Launchpad : <https://espressif.github.io/esp-launchpad>