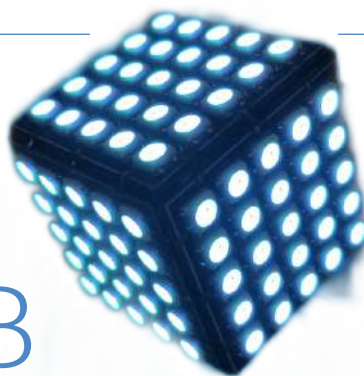


Cube magique de LED RGB

circuit avec un RP2040



Mohamed Belkhir Sayari (Tunisie)

Vous voulez décorer votre bureau ou votre atelier avec un cube de LED RGB original ? Vous aimeriez construire votre propre cube « magique » avec un Raspberry Pi RP2040 ? Alors, ce cube à 6 faces $\times 5 \times 5$ LED est fait pour vous.



Figure 1. Cube de LED RGB.

Vous pouvez construire votre propre gadget en réalisant votre propre circuit. Nous vous proposons dans cet article quelques astuces pour concevoir un appareil de qualité. Entrons dans le vif du sujet. Je vais vous raconter comment j'ai fabriqué un cube de LED RGB commandé par le microcontrôleur Raspberry Pi RP2040 (**figure 1**). J'expliquerai en détail comment utiliser ce microcontrôleur exceptionnel dans vos projets et je présenterai un circuit de base à cet effet.

Conception d'un cube de LED RGB basée sur RP2040

Depuis le début de l'année 2021, il est devenu difficile de se procurer de microcontrôleurs efficaces et abordables à cause de la pénurie mondiale des puces. C'est alors que j'ai découvert que le RP2040 peut être utile aux électroniciens, en ces temps difficiles, grâce à sa petite taille, sa facilité d'utilisation, ses hautes performances et son prix abordable. Le RP2040 était assez nouveau et peu utilisé, donc ce projet m'a donné l'opportunité de l'utiliser pour la première fois [1]. Souhaitant découvrir les configurations les plus appropriées en termes d'alimentation et de programmation pour mon RP2040, j'ai choisi au hasard de réaliser une matrice de 5×5 LED. Mais par la suite, j'ai eu l'idée d'en créer six et de les connecter pour former un cube.

D'abord, il est important de se familiariser avec ce microcontrôleur, et vous devez tenir compte de ses caractéristiques. Reportez-vous au schéma détaillé de la **figure 2**. Le schéma comporte quatre blocs : le microcontrôleur, l'alimentation, l'USB-C et la mémoire. Examinons chaque bloc.

Configuration du microcontrôleur

Le premier bloc de la **figure 2** représente le RP2040 et les composants électroniques de base nécessaires pour son fonctionnement, notamment les condensateurs. Je vous recommande d'utiliser chaque composant présent dans cette partie afin d'éviter toute surprise lorsque vous réalisez votre circuit. La plupart de ces condensateurs sont des condensateurs de découplage de 100 nF, et ils doivent être placés sur chaque ligne d'alimentation du microcontrôleur.

Ces condensateurs de découplage filtrent le bruit de l'alimentation causé par les autres composants du circuit, car ce bruit induit sera shunté par les condensateurs. Ils agissent également comme une source de charge locale qui alimente les circuits internes du RP2040 lors des chutes de tension importantes causées par une augmentation soudaine de la demande de courant.

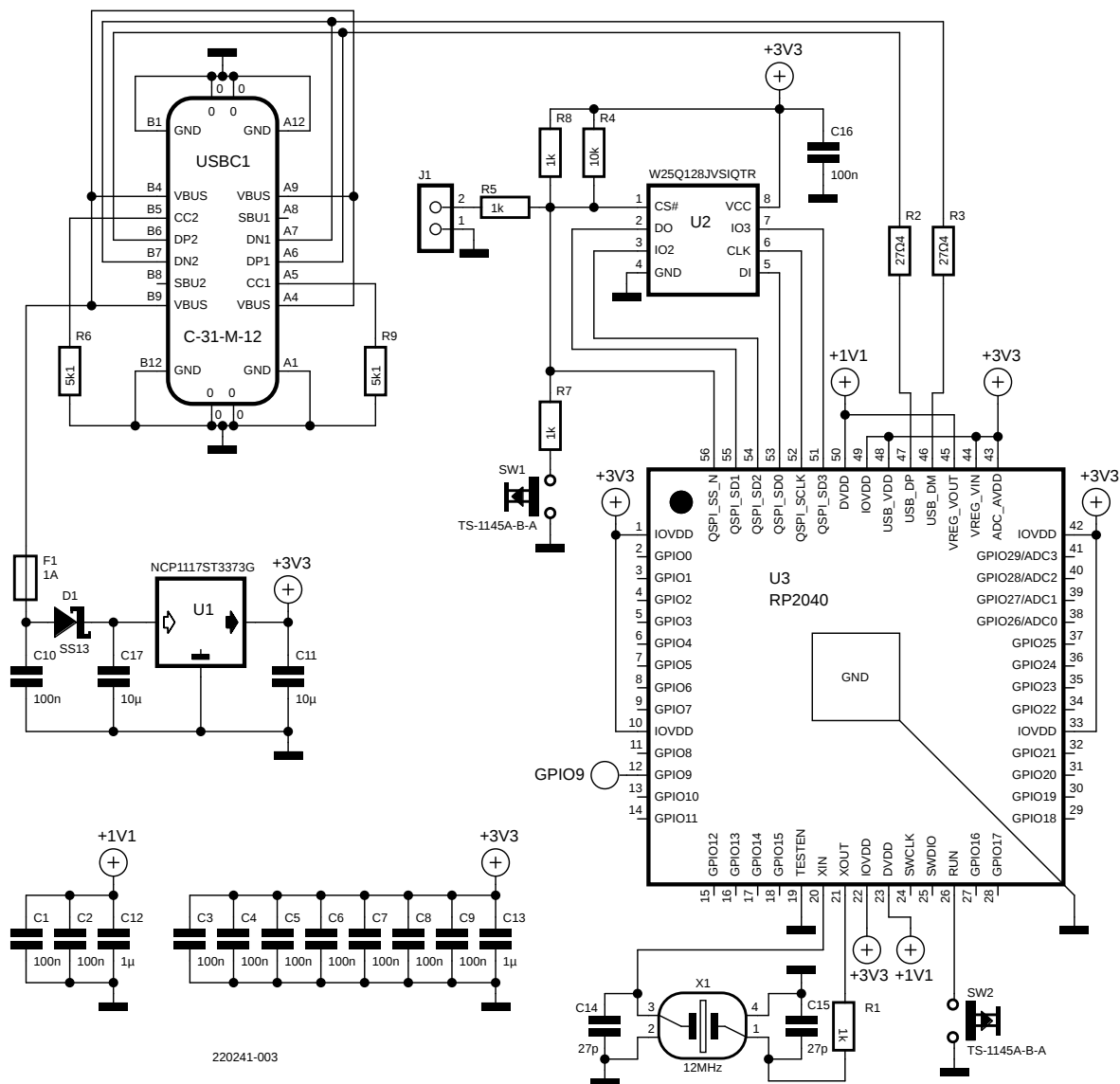


Figure 2. Le RP2040 est le cœur du cube magique RGB.

Vous comprenez maintenant l'importance des condensateurs de découplage. Mais ce n'est pas tout ! Pendant la réalisation du circuit imprimé (**figure 3**), assurez-vous de placer les condensateurs aussi proches que possible des broches d'alimentation de votre microcontrôleur. Cela garantira l'efficacité du filtrage.

Le RP2040 est doté de son propre oscillateur interne, il n'a donc pas

vraiment besoin d'une horloge externe. Comme ce quartz interne n'est pas super stable, il est recommandé d'utiliser un oscillateur externe de 12 MHz (**figure 4**). Nous avons utilisé l'oscillateur X322512MSB4SI. Rappelons que nous avons choisi les composants les plus répandus sur le marché, mais vous pouvez toujours faire votre propre sélection). En choisissant les quartz, vous devez tenir compte de deux

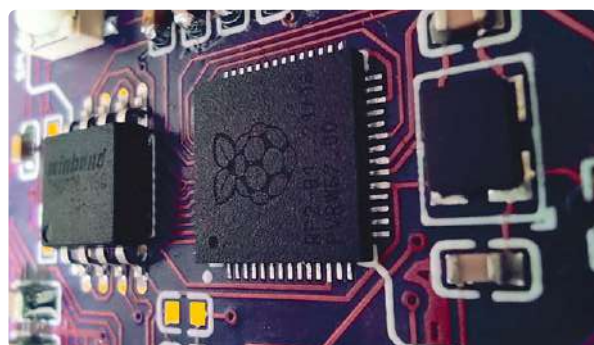


Figure 3. Positionnement des condensateurs de découplage et du quartz.

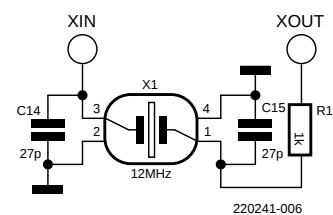
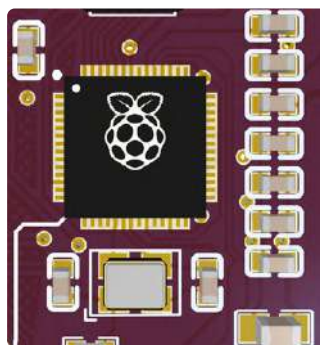


Figure 4. L'oscillateur à quartz externe.

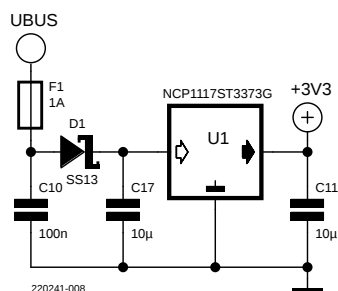


Figure 5. Circuit du régulateur de tension 3,3 V.

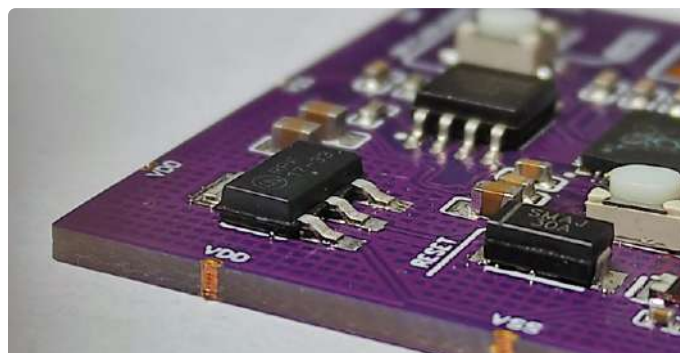


Figure 6. Gros plan sur le régulateur de tension NCP111733T3G.

facteurs principaux : la capacité de charge (LC) et la résistance série équivalente (ESR). Vous pouvez obtenir les paramètres nécessaires en consultant la fiche technique du composant. Celui que nous avons utilisé devait avoir une capacité de charge de 20 pF. Afin de satisfaire un tel besoin, nous avons placé deux condensateurs (C15 et C14) en parallèle aux XIN et XOUT du quartz. Vous vous demandez peut-être : comment déterminer les valeurs de capacité appropriées ? La valeur de la capacité de charge (LC) souhaitée est calculée à partir de C15 et C14 par une formule simple :

$$LC = \frac{C14 \times C15}{C14 + C15}$$

Puisque C14 et C15 ont la même valeur, alors $LC = C14/2$, nous avons donc utilisé une valeur de 27 pF pour C14 et C15. Je sais que $27/2 = 13,5$ pF qui est une valeur inférieure à 20 pF, mais n'oubliez pas que vous utiliserez un circuit imprimé, dont les pistes ont une capacité supplémentaire qui pourrait atteindre 7 pF. Donc, au total, nous aurons une capacité de 20,5 pF, ce qui est assez proche de 20 pF.

Le facteur ESR est également disponible sur la fiche technique du fabricant. Dans notre exemple, il est spécifié comme 80 Ω , et j'ai constaté que cette valeur, associée à une résistance de 1 k Ω (R1), est suffisante pour éviter la suralimentation du quartz.

Gestion de l'alimentation

Ce fut la première fois que j'utilise un microcontrôleur qui nécessite une tension de 1,1 V. Mais je n'avais pas de soucis, car le microcontrôleur dispose déjà d'un régulateur de tension interne destiné à fournir 1,1 V. Il devait être connecté à une source de tension et j'ai utilisé un régulateur de tension externe pour fournir 3,3 V au RP2040 (figure 5). Le NCP1117 est un régulateur de tension couramment utilisé, et il est disponible en plusieurs modèles avec différentes valeurs de tension de sortie. Il faut donc utiliser le régulateur 3,3 V de référence 33T3G. J'ai utilisé un fusible réarmable (PTC) au niveau de l'entrée de l'alimentation (UBUS) pour protéger mon circuit contre les surintensités. La taille compacte de ce composant (figure 6) le rend adapté à notre petit circuit imprimé. Il faut garder à l'esprit que nous réalisons cette carte pour construire un petit cube de LED magique. Nous devons donc disposer tous nos composants sur un circuit imprimé compact

de 45 mm $\times$ 45 mm au maximum. Voilà pourquoi nous avons choisi les composants aux petits boîtiers. Vous pouvez choisir des composants plus grands pour d'autres projets avec le RP2040.

Connecteur USB-C

Vous n'avez pas besoin d'un convertisseur USB-TTL car le RP2040 possède une interface USB interne à pleine vitesse. Il suffit d'utiliser un connecteur USB de votre choix et de connecter l'USB_DP et l'USB_DM du RP2040 aux D+ et D- USB de votre connecteur. J'ai utilisé un connecteur USB de type C (figure 7).

Vous pouvez éliminer ces résistances de rappel au niveau bas (R9 et R6) car le microcontrôleur n'en a pas besoin, mais comme ce montage du connecteur USB C est le plus adapté à tous nos projets, nous les conserverons tant qu'elles seront de petite taille. J'ai donc choisi des composants en boîtiers 0402 pour éviter tout encombrement sur le circuit. Cela a certainement facilité le routage. Avec ce connecteur, le circuit imprimé peut être un hôte ou un périphérique selon le mode de démarrage.

Le placement du connecteur sur la carte doit être légèrement éloigné du bord, car nous devons assembler les pièces du cube par leurs bords. Fondamentalement, nous avons besoin d'une surface libre d'environ 1 à 2 mm autour des bords.

Mémoire flash (stockage du programme)

En comparaison avec les microcontrôleurs classiques AVR et ARM avec lesquels nous sommes familiers, le RP2040 a besoin d'une mémoire externe pour stocker le code qui sera exécuté au démarrage. Le démarrage et l'exécution du code impliquent un transfert de données à haut débit de la puce de stockage vers le microcontrôleur, ce qui explique le choix d'une puce de mémoire flash à liaison QSPI (*Quad Serial Peripheral Interface*) pour stocker le programme (figures 8 et 9). Ces mémoires flash peuvent avoir de différentes capacités. Dans notre exemple, nous utilisons spécifiquement la W25Q128JVSQTR, qui a un espace de stockage de 128 Mbit (16 Mo), la mémoire d'espace maximum que le RP2040 pourrait traiter par son bus de données.

J'ai dû connecter les broches QSPI entre la puce de mémoire flash et le RP2040. J'ai dû rapprocher ces deux composants afin de raccour-

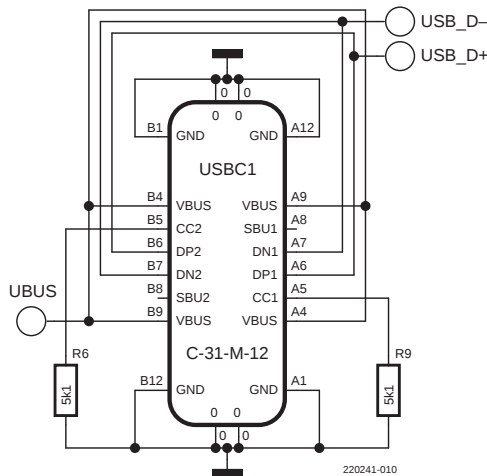


Figure 7. Le connecteur USB-C.

cir au maximum les pistes qui les relient et d'éviter toute erreur de transmission de données causée par les interférences des composants voisins. Je rappelle que lorsque nous avons abordé le circuit du connecteur USB C, j'ai mentionné que le RP2040 pouvait être utilisé comme hôte ou comme périphérique en fonction du mode de démarrage. Cela dépend spécifiquement de la connexion de la broche QSPI_SS pendant le démarrage de la mémoire (**figure 8**).

Et c'est ici que R4, la résistance de rappel au niveau haut 3,3 V, s'avère utile pour fournir 3,3 V à la broche de sélection de puce (CS#) lors de la mise sous tension. Lorsque nous allumons notre appareil, la broche QSPI_SS du RP2040 sera automatiquement tirée vers le niveau haut, mais l'état de la broche reste inconnu pendant la commutation de l'état. Cette résistance (R4) assure l'état de la broche et permet à la mémoire de fonctionner correctement pour charger le code au RP2040 pour l'exécution.

Pour le second mode de démarrage (mode périphérique de stockage),

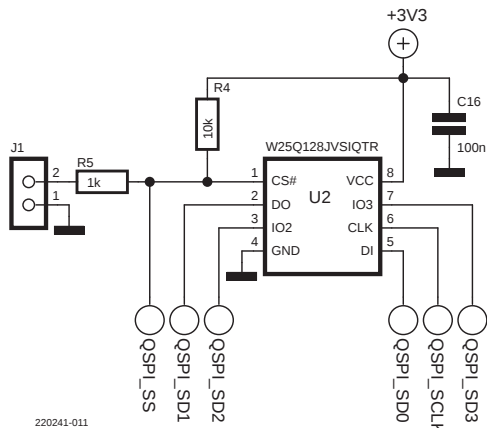


Figure 8. Circuit de la mémoire flash QSPI.

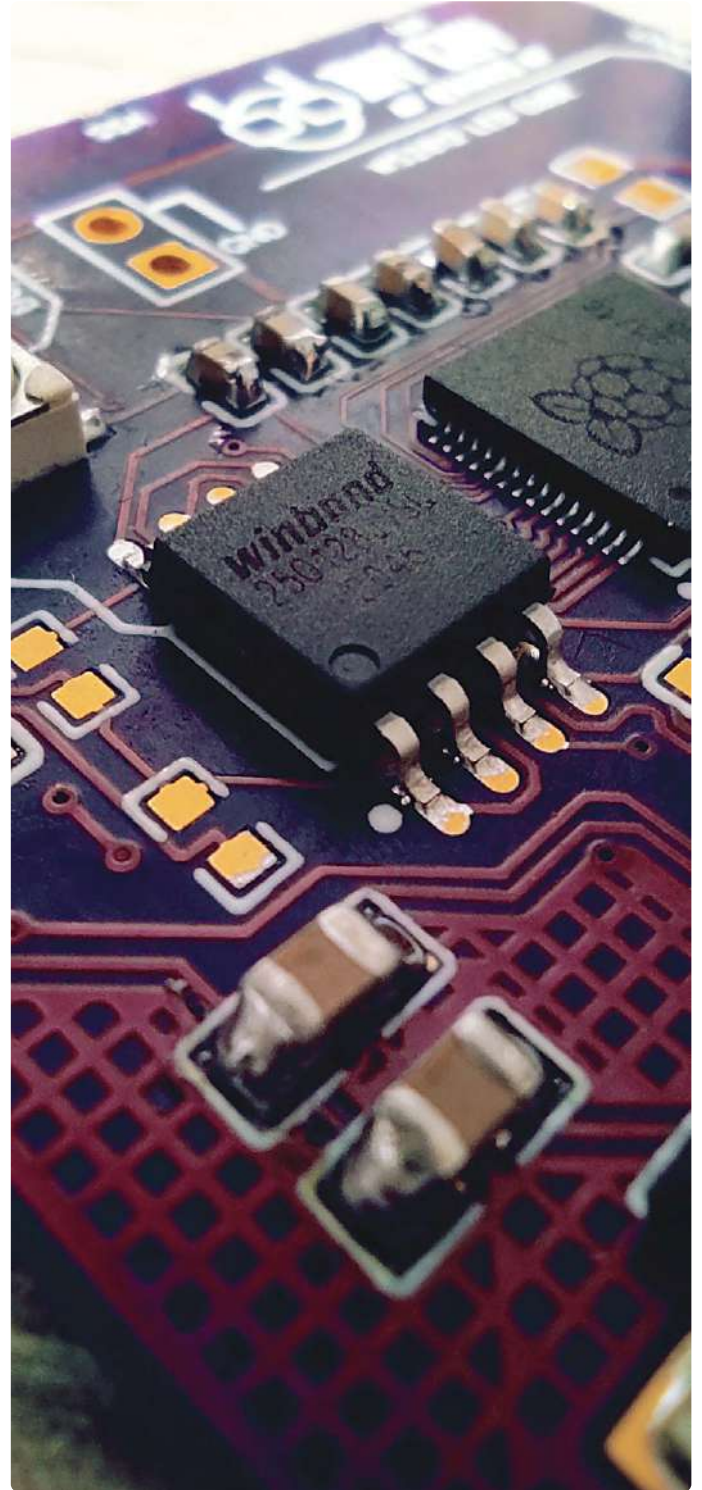
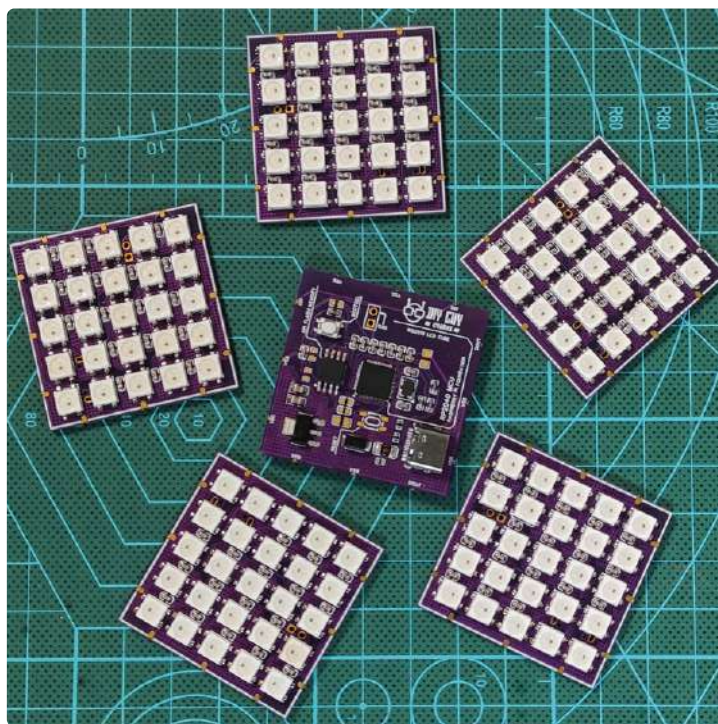
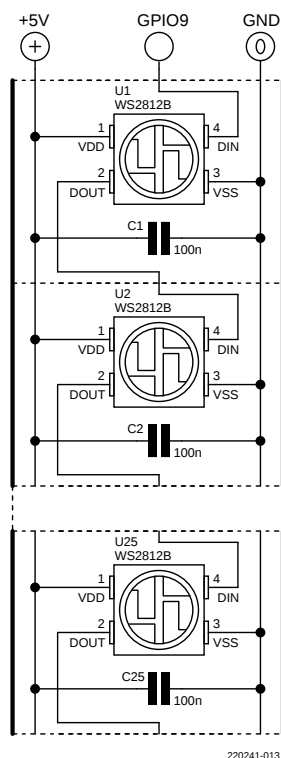


Figure 9. Gros plan sur la mémoire QSPI W25Q128.



Connexion des LED et les cartes à LED terminées.

c'est là que le cavalier (J1) intervient en tirant broche QSPI_SS vers le niveau bas à la mise sous tension du circuit. La mémoire sera reconnue comme un périphérique de stockage USB et le programme pourra y être copié directement. La réinitialisation de l'alimentation du circuit activera le mode de démarrage (boot) et le RP2040 exécutera le programme copié..

En savoir plus sur l'aventure !

La face inférieure de ce circuit imprimé comporte une matrice de LED RGB 5 × 5 connectée en série (**figure 10**), donc si nous envisageons de connecter six exemplaires de ce circuit imprimé, nous nous retrouvons avec un total de 150 LED. Comment peut-on les contrôler toutes en utilisant ce RP2040 ?

En fait, vous utiliserez une seule broche pour contrôler ces LED adressables. Êtes-vous familiers avec ce type de LED (dans notre projet, nous avons utilisé le WS2812B) ? Il s'agit d'un ensemble de LED avec un contrôleur intégré, le tout dans un boîtier de 5 mm × 5 mm. Les données relatives à la luminosité et à la couleur de la lumière sont transmises par la broche de commande du RP2040 vers les LED pour mettre à jour le contrôleur de chacune.

Nous avons construit ce circuit imprimé pour tester nos capacités de conception avec un RP2040. Puisque ce projet de cube de LED a été couronné de succès, cette configuration matérielle fonctionnera certainement pour vos projets personnalisés. Il suffit de suivre les recommandations et les conseils concernant les composants nécessaires et vous réussirez facilement à faire fonctionner votre dispositif. Tous les fichiers associés à ce projet (schémas, fichiers gerber, logiciel RP2040 et fichiers STL pour l'impression 3D) peuvent être téléchargés depuis [1].

220241-04

À propos de l'auteur

Mohamed Belkhir Sayari – « DIY GUY Chris » sur YouTube (www.youtube.com/MEGADAS/) – est un ingénieur R&D en électronique tunisien qui a l'esprit d'un véritable innovateur. Il traite chaque projet comme une aventure d'apprentissage et une occasion de démontrer ses compétences.

Des questions, des commentaires ?

Envoyez un courriel à l'auteur (megadasfirstgate@gmail.com) ou contactez Elektor (redaction@elektor.fr).



PRODUITS

- > D. Ibrahim, *Raspberry Pi Pico for Radio Amateurs* (Elektor 2021) (SKU 20041) www.elektor.com/20041
- > Raspberry Pi RP2040 Microcontroller (SKU 19742) www.elektor.com/19742

LIEN

[1] DIY GUY Chris, « How to Build Magic RGB LED Cube (With Code and Files) », 23 mars, 2022: www.youtube.com/watch?v=A0OtHySzadk