

Débuter en C avec le Pico

Avec les dernières mises à jour, le **Raspberry Pi Pico** peut désormais être programmé directement depuis l'**IDE Arduino**. Cette option simplifie considérablement le développement pour les utilisateurs habitués à l'environnement Arduino et rend l'utilisation du Pico accessible même aux débutants. Dans cet article, nous allons explorer comment configurer l'IDE Arduino pour programmer un Raspberry Pi Pico, et vous présenter un exemple de code simple pour démarrer.

 Découverte

Pourquoi utiliser l'IDE Arduino pour le Raspberry Pi Pico ?

L'IDE Arduino offre une interface conviviale et un ensemble de bibliothèques intégrées pour faciliter la programmation de microcontrôleurs. Utiliser l'IDE Arduino avec le Pico présente plusieurs avantages :

1. **Facilité d'utilisation** : L'interface de l'IDE Arduino est simple et bien connue de nombreux utilisateurs.
2. **Compatibilité avec les bibliothèques Arduino** : Vous pouvez utiliser de nombreuses bibliothèques Arduino existantes sans modification.
3. **Simplicité de téléversement** : Le processus de téléversement est intégré dans l'IDE, rendant la mise en place du code rapide et intuitive.



L'interface de l'IDE Arduino

Menus déroulants

Bouton de compilation

Bouton de téléversement

Nouveau

Ouvrir

Enregistrer

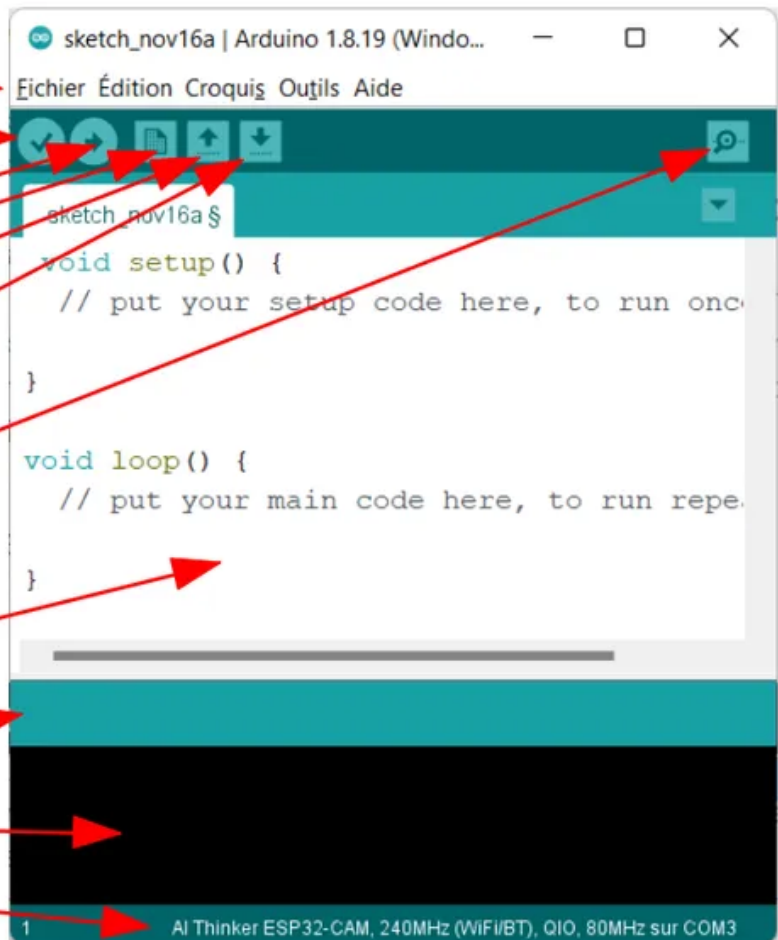
Moniteur série

Éditeur de texte

Opération en cours

Informations de compilation

Microcontrôleur sélectionné



Installation de l'IDE Arduino pour Windows : Version Setup et Version Portable

L'IDE Arduino peut être installé sur Windows de deux manières principales : la version **Setup** (classique) et la version **Portable**. Ces deux options offrent des avantages différents, selon vos besoins en termes d'installation et de portabilité.

1. Installation via la version Setup

La version Setup de l'IDE Arduino est l'installation standard, où le logiciel est intégré au système Windows et se comporte comme une application classique.

- **Téléchargement** : Rendez-vous sur le site officiel d'Arduino <https://www.arduino.cc/en/software> et téléchargez le fichier d'installation pour Windows (généralement sous forme de fichier **.exe**).
- **Installation** : Lancez le fichier **.exe** téléchargé, puis suivez les instructions à l'écran. Vous pouvez accepter les options par défaut ou personnaliser le dossier d'installation et les composants.
- **Finalisation** : Une fois l'installation terminée, l'IDE Arduino peut être lancé depuis le menu Démarrer. Avec cette installation, l'IDE est lié au système, et ses paramètres seront sauvegardés dans les dossiers

2. Installation de la version Portable

La version Portable est idéale si vous souhaitez utiliser l'IDE Arduino depuis une clé USB ou conserver les paramètres et bibliothèques dans un dossier dédié sans toucher au système.

- **Téléchargement** : Téléchargez le fichier ZIP de l'IDE Arduino sur le même lien officiel.
- **Extraction** : Décompressez le contenu du ZIP dans un dossier de votre choix (par exemple, sur une clé USB ou dans un répertoire spécifique).
- **Création du mode portable** : Pour activer le mode portable, créez un dossier nommé `portable` à la racine du dossier de l'IDE Arduino (celui contenant `arduino.exe`). Ce dossier contiendra toutes les préférences, bibliothèques et extensions, séparées du système.
- **Utilisation** : Lancez `arduino.exe` depuis le dossier décompressé. L'IDE fonctionnera en mode portable, avec toutes les configurations enregistrées dans le dossier `portable`, ce qui permet de déplacer facilement l'IDE sur d'autres ordinateurs.

En choisissant entre la version Setup ou Portable, vous pouvez adapter l'IDE Arduino à vos besoins, que ce soit pour une installation fixe sur votre ordinateur personnel ou une version mobile à emporter dans vos projets.

Configurer l'IDE Arduino pour le Raspberry Pi Pico

Avant de commencer à programmer le Pico dans l'IDE Arduino, il faut ajouter la carte Raspberry Pi Pico à la liste des cartes compatibles dans l'IDE. Voici les étapes à suivre :

1. Ouvrez l'IDE Arduino et allez dans **Fichier > Préférences**.
2. Dans le champ **URL de gestionnaire de cartes supplémentaires**, ajoutez l'URL suivante :
`https://github.com/earlephilhower/arduino-pico/releases/download/global/package_rp2040_index.json`

Si vous avez déjà des URLs dans ce champ, séparez-les par une virgule.
3. Cliquez sur **OK** pour enregistrer les préférences.
4. Ensuite, allez dans **Outils > Type de carte > Gestionnaire de cartes**.
5. Dans la barre de recherche, tapez "Raspberry Pi Pico" et installez le package appelé **Raspberry Pi Pico/RP2040 by Earle Philhower**.
6. Une fois l'installation terminée, retournez dans **Outils > Type de carte** et sélectionnez **Raspberry Pi Pico** ou **Raspberry Pi Pico W** (si vous avez la version WiFi).

Vous êtes maintenant prêt à programmer le Pico dans l'IDE Arduino !

Exemple de programme : Faire clignoter une LED

Pour tester que tout est bien configuré, voici un programme simple qui fait clignoter la LED intégrée du Pico. Cette LED est connectée au **GPIO 25**.

```

1  // Configuration initiale
2  void setup() {
3      pinMode(25, OUTPUT); // Définit le GPIO 25 comme une sortie
4  }
5
6  // Boucle principale
7  void loop() {
8      digitalWrite(25, HIGH); // Allume la LED
9      delay(1000);           // Attend 1 seconde
10     digitalWrite(25, LOW);  // Éteint la LED
11     delay(1000);           // Attend 1 seconde
12 }

```

Explication du code

- `pinMode(25, OUTPUT);` : Configure le GPIO 25 comme une sortie pour contrôler la LED.
- `digitalWrite(25, HIGH);` : Alimente le GPIO 25, ce qui allume la LED.
- `delay(1000);` : Pause d'une seconde (1000 ms).
- `digitalWrite(25, LOW);` : Coupe l'alimentation du GPIO 25, éteignant ainsi la LED.

Le programme alterne donc entre l'état allumé et éteint toutes les secondes.

Téléverser le programme sur le Pico

Pour téléverser le programme sur le Raspberry Pi Pico :

1. Connectez votre Pico à votre ordinateur via un câble USB tout en maintenant le bouton **BOOTSEL** enfoncé. Cela met le Pico en mode de stockage de masse.
2. Dans l'IDE Arduino, sélectionnez le port correspondant à votre Pico dans **Outils > Port**.
3. Cliquez sur **Téléverser**. L'IDE Arduino compile le code et le télécharge directement sur le Pico.

Une fois le téléversement terminé, la LED du Pico devrait commencer à clignoter toutes les secondes, indiquant que le programme fonctionne.

Conclusion

En utilisant l'IDE Arduino, le Raspberry Pi Pico devient un outil extrêmement accessible pour les débutants et les makers souhaitant explorer l'univers de la programmation embarquée. Cette configuration vous permet de tirer parti des bibliothèques Arduino et d'un environnement familier pour développer rapidement vos projets. Que vous soyez intéressé par la robotique, l'IoT ou la simple automatisation, le Pico et l'IDE Arduino forment un duo gagnant pour faire de vos idées une réalité.

Installation des drivers

Oui, pour utiliser le **Raspberry Pi Pico** avec l'IDE Arduino, vous pourriez avoir besoin de **drivers spécifiques** pour que votre ordinateur reconnaisse correctement le Pico. Ces drivers sont parfois déjà intégrés dans certains systèmes d'exploitation, mais voici les étapes pour vérifier et installer les drivers nécessaires si votre Pico n'est pas détecté correctement.

Installation des drivers pour Windows

1. **Mode BOOTSEL** : Lorsque vous connectez le Raspberry Pi Pico en maintenant le bouton **BOOTSEL** enfoncé, il doit apparaître comme un périphérique de stockage USB. Cette action ne nécessite pas de driver spécifique, mais si vous voulez téléverser du code directement depuis l'IDE Arduino, il faut installer un driver supplémentaire.
2. **Driver USB pour le Pico** :
 - Installez les drivers RP2040 via le **Gestionnaire de cartes Arduino**, qui inclut généralement le nécessaire pour reconnaître le Pico.
 - Si l'IDE Arduino n'arrive toujours pas à détecter le Pico, vous pouvez utiliser le **Zadig Utility**, un logiciel pour installer manuellement des drivers USB. Voici comment faire :
 1. Téléchargez **Zadig** depuis le site officiel : <https://zadig.akeo.ie/>.
 2. Ouvrez **Zadig** et sélectionnez **USB Serial (Interface 0)** ou un périphérique semblable dans la liste.
 3. Installez le driver **WinUSB** ou **libusb** selon ce qui est recommandé par le logiciel pour votre configuration.

Une fois cela fait, l'IDE Arduino devrait détecter correctement le Pico.

Vérification dans l'IDE Arduino

Une fois les drivers installés et les permissions configurées (si nécessaire), lancez l'IDE Arduino et allez dans **Outils > Port**. Le Raspberry Pi Pico devrait apparaître dans la liste des ports disponibles.

En résumé :

- **Windows** : Installez les drivers via le Gestionnaire de cartes Arduino, ou utilisez **Zadig** si le Pico n'est toujours pas détecté.

Avec ces étapes, votre Pico devrait être reconnu par l'IDE Arduino, vous permettant de téléverser du code sans difficulté.