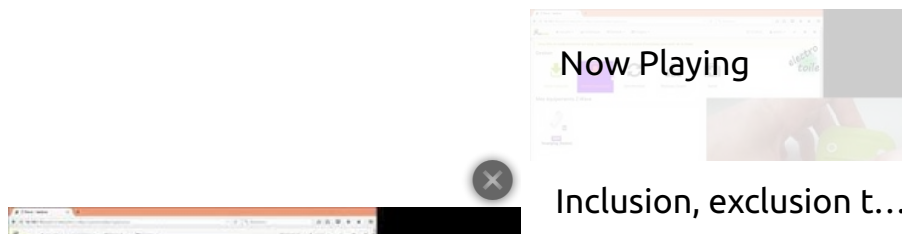
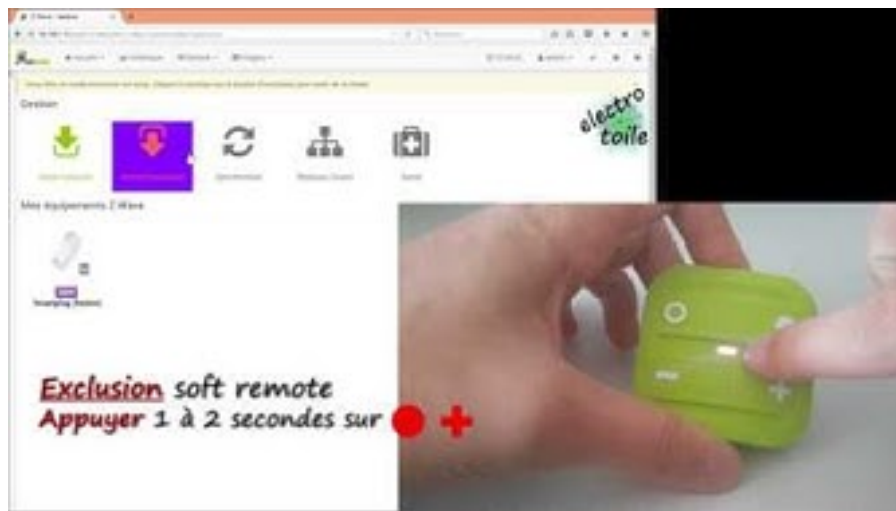




## TUTORIELS

# Activer UART sur Raspberry Pi : le guide pas à pas





Inclusion, exclusion télécommande Soft Remote (Nodon) da...

Un Raspberry Pi peut envoyer et recevoir des informations de nombreuses façons différentes. Le protocole UART (Universal Asynchronous Receiver/Transmitter), permet la communication entre des appareils connectés. Sur le Raspberry Pi, cette méthode de communication est couramment référencée sous l'appellation « Port Série ».

**Sur Raspberry Pi OS, le port série (UART) peut être activé en utilisant l'interface graphique ou en lignes de commande pour une utilisation en terminal. Il existe également une option dans le fichier config.txt.**

Avant de discuter des différentes options pour activer UART sur Raspberry Pi, commençons par une brève introduction à ce qu'est exactement UART et comment le configurer pour le Raspberry Pi.

À propos, si vous débutez sur Raspberry Pi et que vous souhaitez accélérer les choses (tout en vous amusant), n'hésitez pas à **télécharger mon e-book Maitrisez Raspberry Pi en 30 jours**. Il s'agit d'un défi de 30 jours dans lequel vous apprendrez une nouvelle compétence chaque jour. Vous économiserez du temps et de l'argent et pourrez enfin profiter de votre Raspberry Pi à son plein potentiel. Je vous offre d'ailleurs 10% de remise aujourd'hui si vous utilisez ce lien !

## Qu'est-ce que l'UART et le protocole de communication série ?

**UART signifie Universal Asynchronous Receiver-Transmitter protocol. C'est un protocole de communication répandu pour les appareils qui ont recours à une transmission de données bit par bit, en série.**

Il peut utiliser sur un ou deux fils en fonction de l'implémentation.

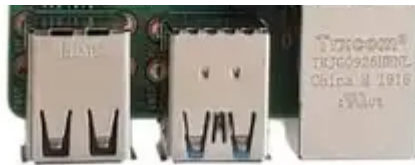
Il permet de nombreuses options de configuration pour des choses comme le format des données et les vitesses de transmission (souvent appelées taux de baud, qui correspond au nombre de bits transmis par seconde). Dans l'ensemble, c'est un protocole de communication très polyvalent et largement utilisé.

Concernant le Raspberry Pi, ce protocole a été défini comme Port Série. Il existe également une option Console Série supplémentaire, qui est configurable de manière similaire à la dernière option. Nous allons brièvement discuter de ces deux options pour bien comprendre.

## Le Port Série

**Le Port Série est l'interface UART réelle qui est disponible via les broches GPIO 14 et 15 du Raspberry Pi (appelées broches TX et RX). Cette interface est désactivée par défaut, donc si nous voulons utiliser UART sur le Pi, nous devons l'activer.**





GPIO 6	31	32	GPIO 12
GPIO 13	33	34	GROUND
GPIO 19	35	36	GPIO 16
GPIO 26	37	38	GPIO 20
GROUND	39	40	GPIO 21

Les différentes manières de faire seront discutées en détail dans cet article.

## Téléchargez mon antisèche !

Format PDF pratique, avec les 74 commandes à retenir pour maîtriser son Raspberry Pi.

Télécharger maintenant

## La Console Série

Imaginez un scénario où vous ne pouvez pas accéder à un affichage HDMI ou à SSH pour le Pi. Comment accéderiez-vous au Pi dans ce cas ? C'est là que la Console Série entre en jeu.

**La Console Série vous permet d'accéder au shell du Pi via UART.** Par défaut, cette option est désactivée, car c'est un risque de sécurité pour le Pi. N'importe qui ayant un accès physique à votre Pi pourrait y accéder, si l'option a été activée. Cependant, c'est très utile pour le débogage, pour comprendre ce qui ne fonctionne pas avec une interface existante.

Nous allons voir comment accéder à votre terminal Pi via UART dans la dernière section de ce tutoriel. Maintenant que nous avons une idée de ce qu'est UART et de la manière dont il a été configuré pour le Raspberry Pi, voyons les différentes façons d'activer UART sur le Pi.

## Configuration de Raspberry Pi OS pour les connexions série UART

Cette section couvrira les différentes manières d'activer UART sur le Raspberry Pi (fonctionnant sous Raspberry Pi OS Bookworm).

Il y a trois principales manières d'activer UART sur le Pi :

- **Via l'application de configuration Raspberry Pi.**
- **Via le terminal Raspberry Pi.**
- **En modifiant les fichiers de démarrage pour activer UART.**

Alors, explorons chaque méthode en détail maintenant.

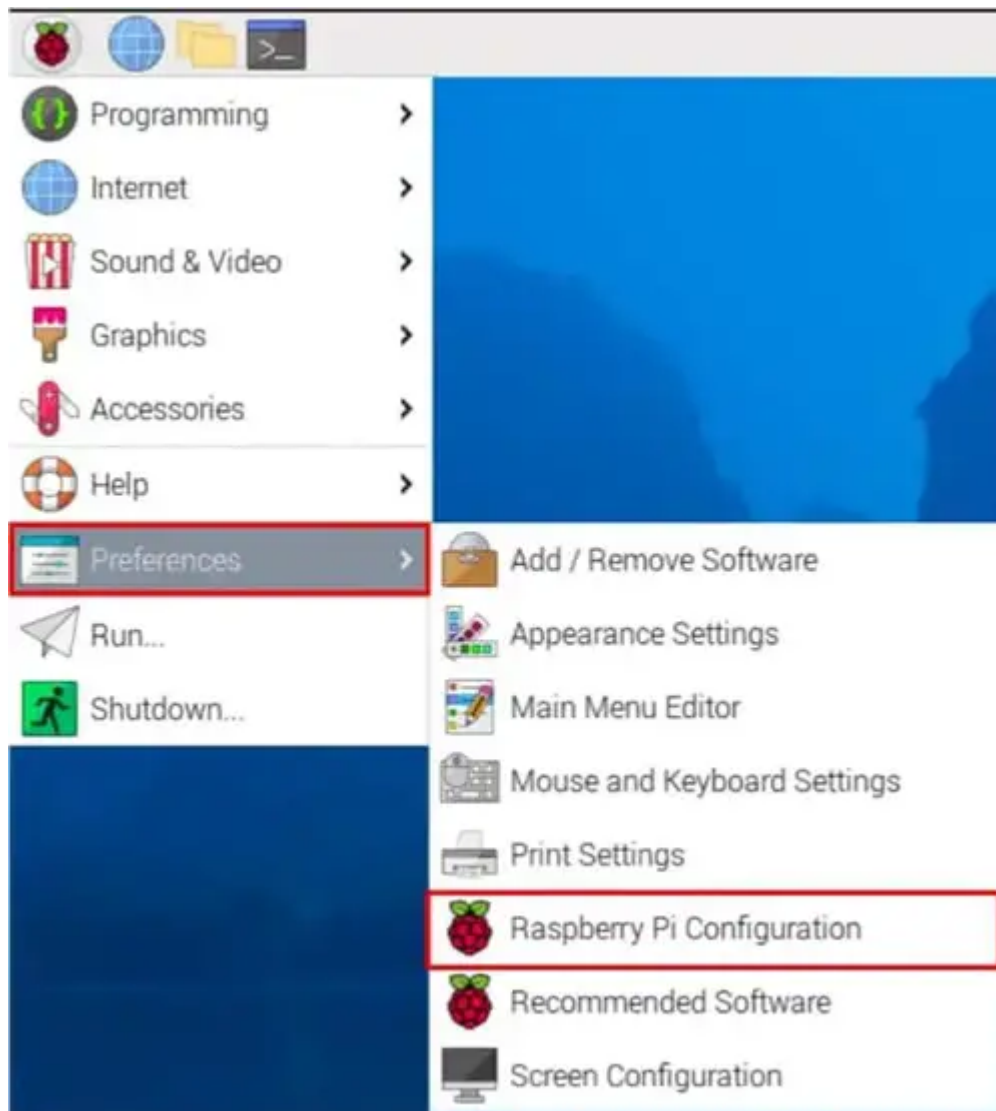
**Remarque** : l'ensemble de ce tutoriel a été testé pour un Raspberry Pi 4 exécutant la dernière version de Bookworm. Cependant, les étapes montrées dans ce tutoriel devraient fonctionner pour les anciens modèles de Pi exécutant des systèmes d'exploitation hérités.

## Comment activer UART sur Raspberry Pi en utilisant l'interface graphique

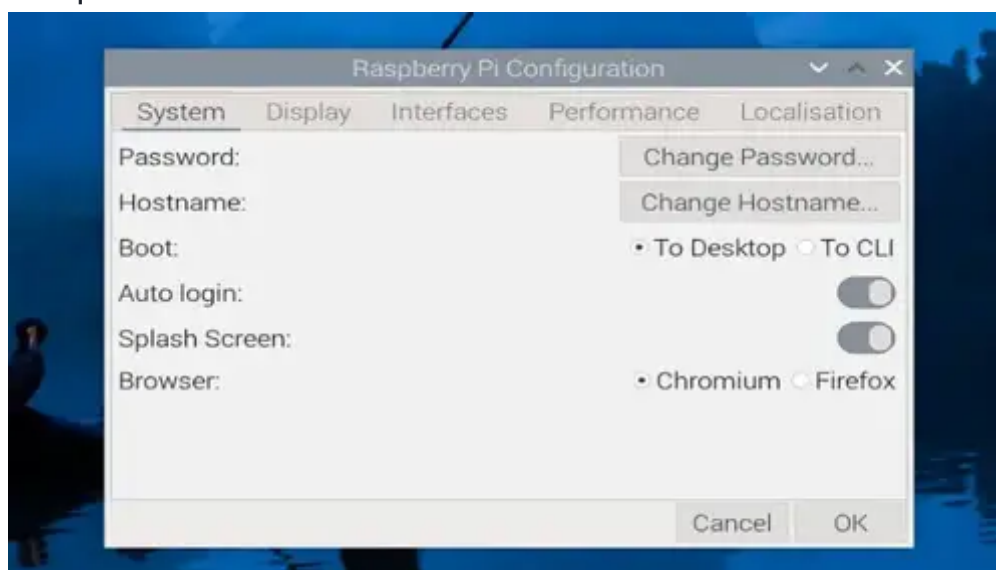
Conçu par la Fondation Raspberry Pi, Raspberry Pi Configuration est un outil facile à utiliser pour configurer votre Raspberry Pi. L'application de configuration vous offre un accès rapide à une multitude d'options de configuration disponibles au sein de Raspberry Pi OS.

**Comme discuté précédemment, deux options correspondent aux connexions en série UART : Port série et Console série.** Faisons une brève présentation du processus pour les activer, pas à pas :

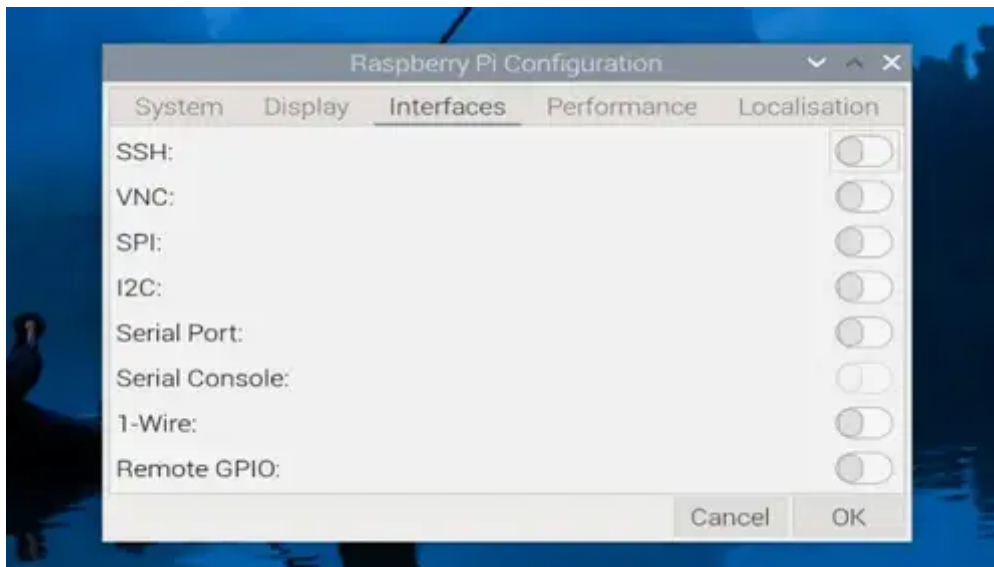
- Ouvrez le menu des applications en haut à gauche de l'environnement de bureau Raspberry Pi OS. Sélectionnez **Preferences >> Raspberry Pi Configuration**.



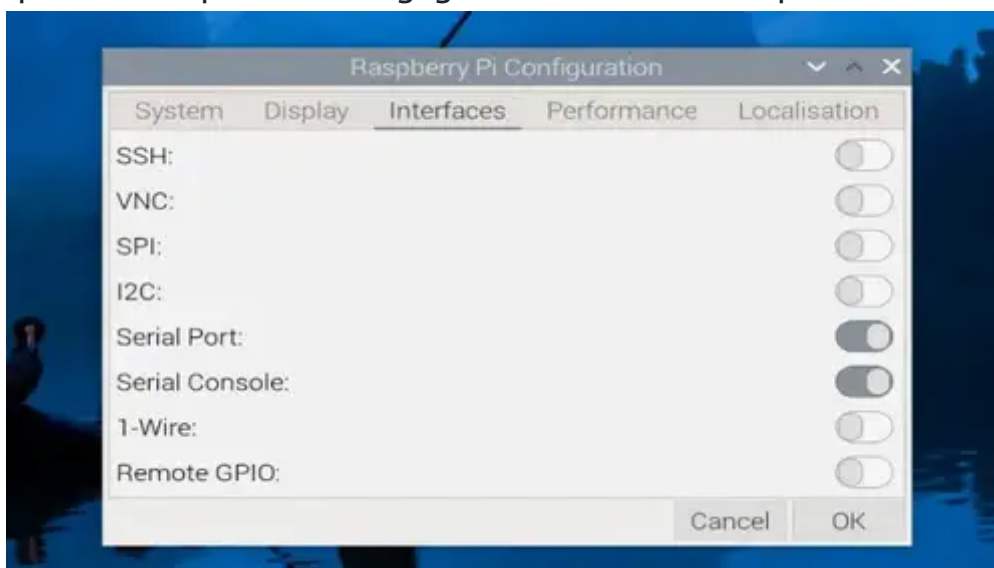
- Cela ouvrira une nouvelle **fenêtre de configuration Raspberry Pi**, comme indiqué ci-dessous.



- Ensuite, déplacez-vous dans l'onglet "Interfaces" au milieu des cinq groupes d'options. **Sous cet onglet, vous trouverez deux options pour le protocole de communication UART : Serial Port et Serial Console.**



- Lors de l'activation des options Série, vous avez peut-être remarqué que la Console série est activée par défaut et que vous ne pouvez pas la désactiver. C'est parce que la Console série a besoin du Port série pour fonctionner. Une fois le Port série activé, l'option Console série peut être à son tour activée.
- Activez les deux options car cela nous permettra d'accéder au terminal du Pi, ce qui est idéal pour le débogage comme mentionné précédemment.



- Une fois que vous avez terminé avec l'application de configuration Raspberry Pi, cliquez sur "OK" et laissez votre configuration se mettre à jour. On vous demandera ensuite si vous souhaitez redémarrer Raspberry Pi OS maintenant ou plus tard. Pour consolider les changements, il est toujours préférable de redémarrer immédiatement.

Super ! Vous avez activé UART sur votre Pi et pouvez désormais vous connecter en série aux appareils pris en charge via l'interface GPIO du Pi pour UART (broches Tx et Rx). Nous en apprendrons plus à ce sujet dans la dernière section de cet article.



## Comment activer l'interface UART / Série dans Raspberry Pi OS en utilisant les commandes du terminal

Utiliser les commandes bash offre une méthode différente de configuration de votre Raspberry Pi. L'écran de configuration du terminal ressemble plus ou moins à l'application de configuration du Raspberry Pi, même si la balance penche plutôt du côté « moins ».

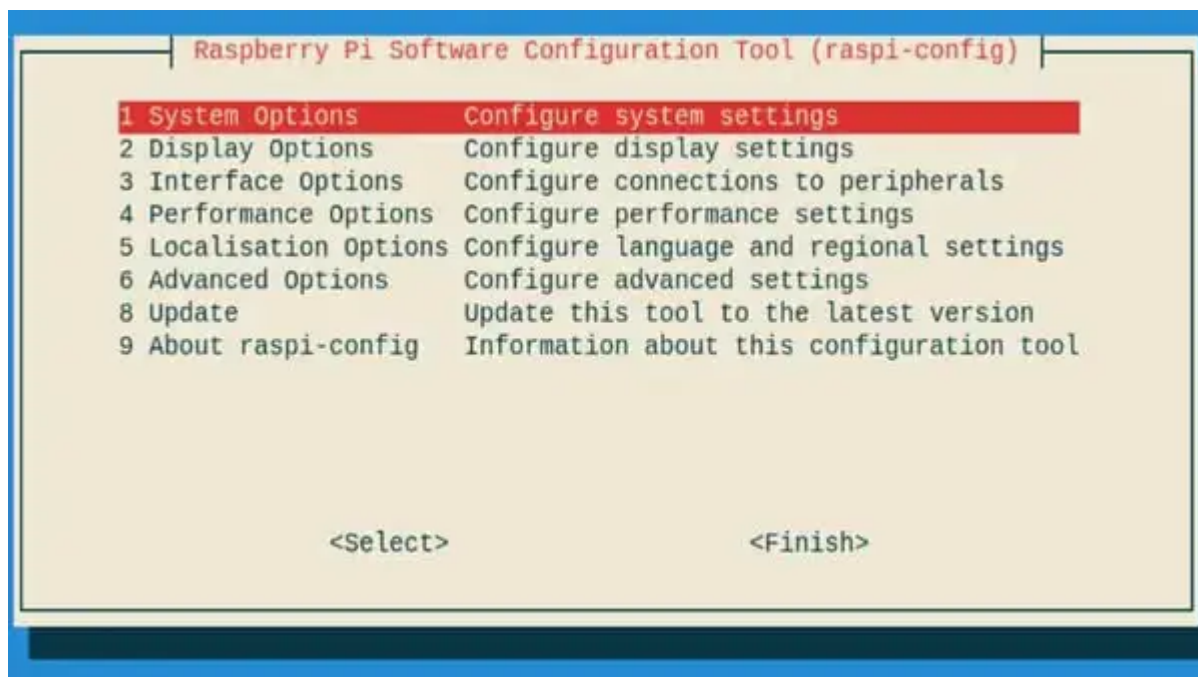
L'outil de configuration du terminal offre tout ce que fait l'application logicielle, avec quelques options en plus. Activer UART est tout aussi facile et plus rapide que la première méthode.

Activer UART depuis le terminal est un jeu d'enfant. Pour effectuer l'ensemble du processus, les étapes pour utiliser l'outil de configuration du terminal sont :

- Cliquez sur le raccourci pour le terminal sur le côté gauche de la barre des tâches et tapez la commande shell suivante :

```
sudo raspi-config
```

- **Cela lancera immédiatement l'outil de configuration logicielle Raspberry Pi** et vous affiche la fenêtre suivante dans votre terminal :



- Une fois « Serial Port » sélectionné, deux ou trois fenêtres vous proposent des options et des informations pour activer UART. **La première fenêtre demandera <Yes> ou <No> pour disposer d'un shell de connexion accessible**

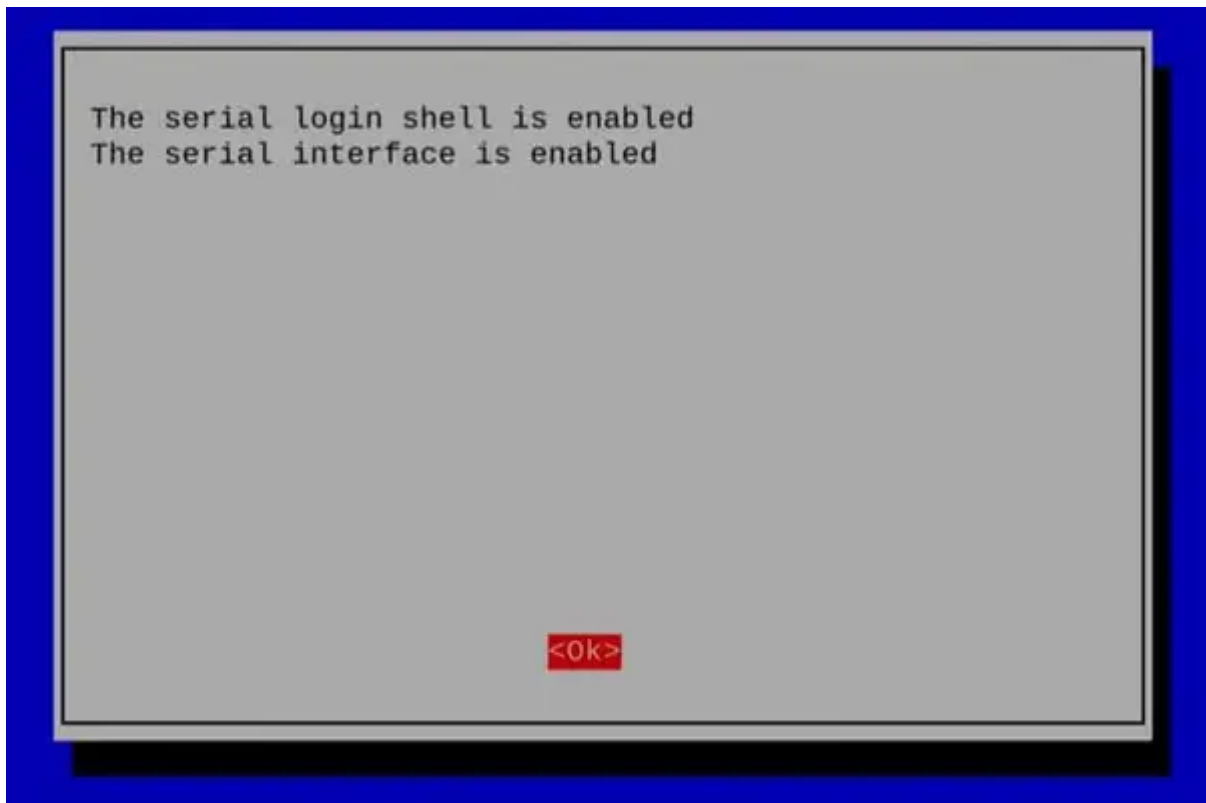


**par UART. En sélectionnant <Yes>, vous activez à la fois le port série et la console série.**

- Notez que si vous répondez <No> à la première fenêtre, une deuxième fenêtre avec une question <Yes> ou <No> sur l'activation du port série uniquement s'affiche. Ce n'est le cas que si vous avez répondu No à la première question.



- En fonction de votre choix, vous verrez l'écran indiquant que les deux options sont activées (shell de connexion série et interface série) ou seulement l'interface série. Pour ce tutoriel, nous avons activé les deux options comme le montre la capture ci-dessous.



- Finalement, le terminal affichera ce que vous aviez choisi comme moyen de présenter les modifications de l'interface.

Vous avez réussi à activer UART sur votre Pi en utilisant le terminal ! Pour ce tutoriel, activez à la fois le Port série et la Console série, car nous allons voir comment accéder au terminal de votre Pi en série dans la dernière section.

## Téléchargez mon antisèche !

Format PDF pratique, avec les 74 commandes à retenir pour maîtriser son Raspberry Pi.

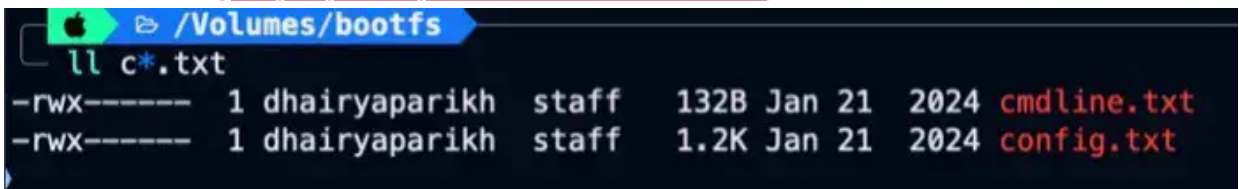
Télécharger maintenant

## Modification des fichiers de démarrage pour activer UART sans démarrer le système

Il peut y avoir des cas où vous n'avez ni moniteur HDMI ni d'accès SSH à votre Raspberry Pi. Dans ce cas, comment configureriez-vous votre Pi initialement ? De plus, il peut y avoir des problèmes techniques qui ne vous permettent pas d'accéder à votre Raspberry Pi du tout.

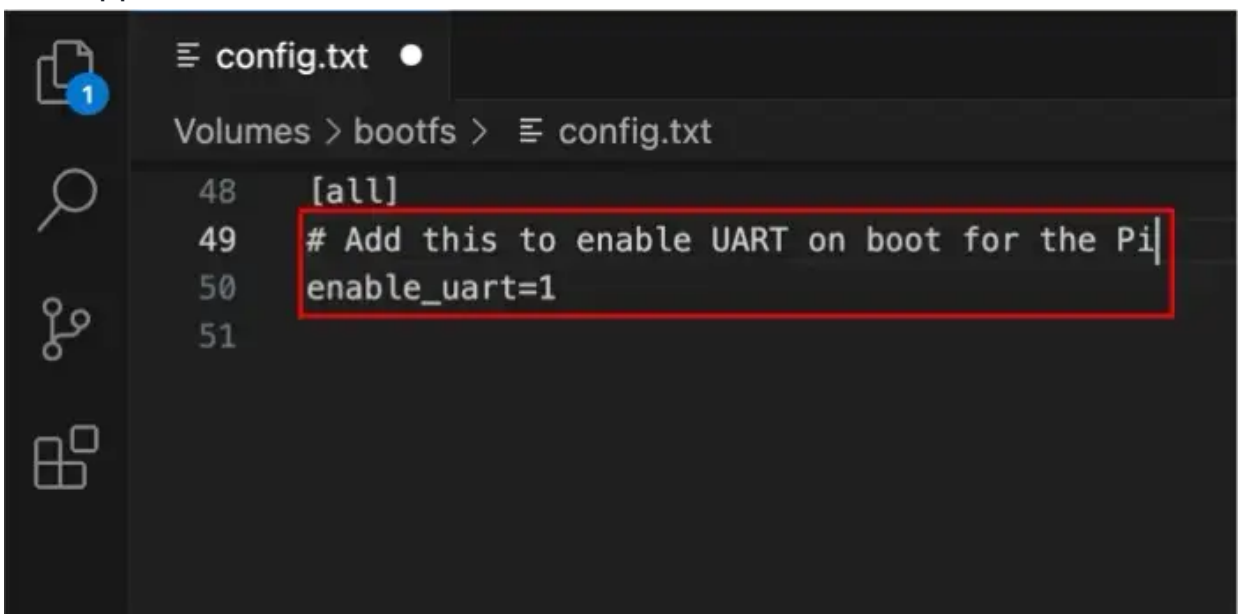
Dans ces cas, l'accès à la Console série est la meilleure solution. Mais pour cela, vous devez activer l'interface série globale pour votre Pi. Nous allons apprendre à faire exactement cela dans cette section bonus !

- Vous aurez besoin de la carte SD sur laquelle le système d'exploitation Raspberry Pi de votre Pi a été installé. Insérez-la dans votre système principal, puis naviguez vers ce répertoire particulier. Nous sommes intéressés par deux fichiers spécifiques :
  - config.txt
  - cmdline.txt ([j'explique à quoi il sert en détails ici](#))



```
ll c*.txt
-rwx----- 1 dhairyaparikh staff 132B Jan 21 2024 cmdline.txt
-rwx----- 1 dhairyaparikh staff 1.2K Jan 21 2024 config.txt
```

- Tout d'abord, nous devons activer l'interface série. Pour cela, **ouvrez le fichier config.txt** dans votre éditeur de texte préféré et **ajoutez ce qui suit à la fin du fichier : "enable\_uart=1"**. Cela indiquera à notre Pi d'activer l'interface série lors du démarrage. N'oubliez pas de sauvegarder les modifications que vous avez apportées.



```
48 [all]
49 # Add this to enable UART on boot for the Pi
50 enable_uart=1
51
```

- Parfait, vous avez activé UART pour votre Pi sans même le démarrer ! Génial, non ? Maintenant, si vous voulez accéder à la console série, vous devez apporter certaines modifications supplémentaires à un autre fichier de démarrage : **cmdline.txt**.

```
# Ajoutez ceci si ce n'est pas déjà ajouté  
console=serial0,115200
```

Une fois que vous avez effectué ces modifications, enregistrez-les et c'est terminé !

Vous connaissez désormais trois différentes manières d'activer UART sur votre Raspberry Pi ! Ensuite, nous verrons cette interface en action. Pour cela, nous accèderons au terminal du Pi via UART dans la dernière section de ce tutoriel.

## Téléchargez mon antisèche !

Format PDF pratique, avec les 74 commandes à retenir pour maîtriser son Raspberry Pi.

Télécharger maintenant

Vous comprenez l'anglais et en voulez encore plus ? Je publie une nouvelle leçon chaque mois, réservée à mes abonnés premium. [Rejoignez la communauté dès maintenant](#) pour y avoir accès !

## Interface UART en action – Console Série

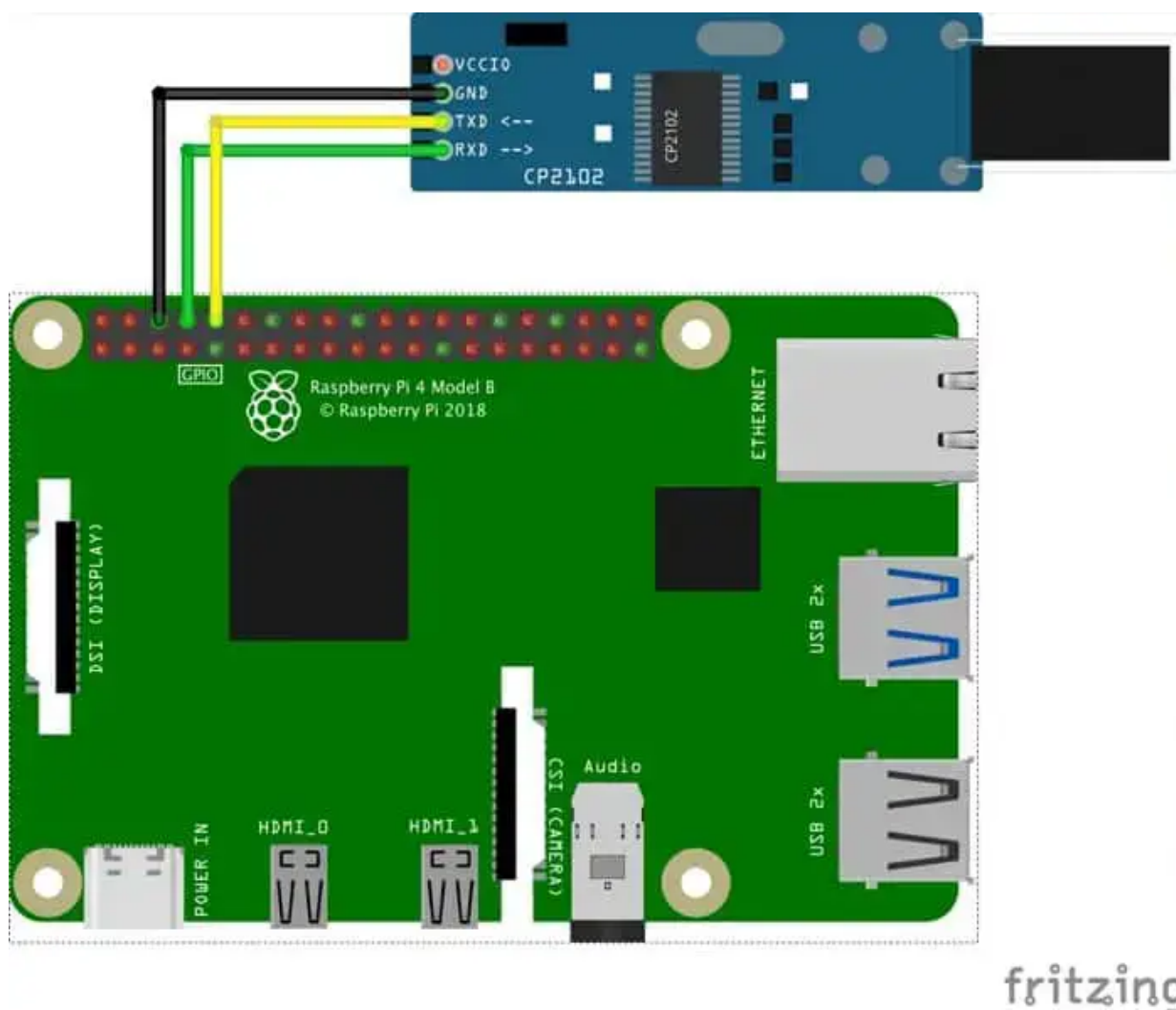
Une fois que vous avez activé l'interface UART sur votre Pi par l'une des méthodes décrites ci-dessus, vous devriez être capable d'utiliser l'interface série pour accéder au terminal du Pi. Pour cela, vous aurez besoin d'un périphérique de conversion USB vers TTL en plus de votre Pi, qui ressemble à ceci :





Une fois que vous avez tout préparé, suivez les étapes ci-dessous pour voir l'interface UART en action.

- La première étape consiste à connecter votre Raspberry Pi à cette carte de conversion USB-vers-TTL afin de pouvoir accéder à l'interface série du Pi via d'autres appareils. Suivez simplement le schéma d'implantation ci-dessous pour effectuer les connexions nécessaires.



Les connexions sont les suivantes :

**Pin GND du Raspberry Pi (Ground Pin) >>**

**Pin GND du UART BoardPin TXD du Raspberry Pi (GPIO14) >>**

**Pin RXD du UART BoardPin RXD du Raspberry Pi (GPIO15) >> Pin TXD du UART Board**

- Une fois que vous avez établi ces connexions, vous êtes prêt à voir cette interface en action ! **Facultatif** : Par défaut, les journaux de démarrage sur la console sont désactivés pour des raisons de sécurité. Cependant, une petite modification dans le fichier de démarrage **cmdline.txt** nous permet d'y accéder.

Pour imprimer les journaux de démarrage sur la console, nous devons supprimer le contenu suivant du fichier **cmdline.txt** :

```
quiet splash plymouth.ignore-serial-consoles
```

Cela nous permettra d'accéder aux journaux de démarrage du Pi, ce qui peut être très utile pour déboguer d'éventuels problèmes de démarrage.

```
[ OK ] Finished dphys-swapfile.service - Create/Update/Unmount, and delete a swap file.
[ OK ] Started polkit.service - Authorization Manager.
[ OK ] Started avahi-daemon.service - Avahi mDNS/DNS-SD Stack.
Starting ModemManager.service - Modem Manager...
[ OK ] Finished glamor-test.service - Check for glamor.
[ OK ] Finished rpi-test.service - Check for RPi displays for Xorg.
[ OK ] Started wpa_supplicant.service - WPA supplicant.
[ OK ] Started systemd-logind.service - User Login Management.
[ OK ] Finished e2scrub_reap.service - ext4 Metadata Check Snapshots.
[ OK ] Started accounts-daemon.service - Accounts Service.
[ OK ] Started NetworkManager.service - Network Manager.
[ OK ] Reached target network.target - Network.
Starting NetworkManager-wait-online.service - Network Manager Wait Online...
Starting cups.service - CUPS Scheduler...
Starting openvpn.service - OpenVPN service...
Starting ssh.service - OpenBSD Secure Shell server...
[ OK ] Started raspi-config.service - Raspberry Pi Configuration (unless shift key is pressed).
[ OK ] Finished rpi-eeprom-update.service - Raspberry Pi EEPROM updates.
[ OK ] Finished openvpn.service - OpenVPN service.
Starting systemd-hostnamed.service - Hostname Service...
[ OK ] Started udisks2.service - Disk Manager.
[ OK ] Started cups.service - CUPS Scheduler.
[ OK ] Started ModemManager.service - Modem Manager.
[ OK ] Listening on systemd-rfkill.socket - Switch Status /dev/rfkill Watch.
Starting systemd-rfkill.service - Load/Save RF Kill Switch Status...
[ OK ] Started ssh.service - OpenBSD Secure Shell server.
[ OK ] Created slice system-bluetooth.slice - Slice /system/bluetooth.
Starting hcibluetooth.service - Bluetooth Modems connected by UART...
[ OK ] Started systemd-rfkill.service - Load/Save RF Kill Switch Status.
[ OK ] Started hcibluetooth.service - Bluetooth Modems connected by UART.
Starting bthelper@hci0.service - Raspberry Pi bluetooth helper...
[ OK ] Started systemd-hostnamed.service - Hostname Service.
[ OK ] Finished bthelper@hci0.service - Raspberry Pi bluetooth helper.
Starting bluetooth.service - Bluetooth service...
Starting NetworkManager-dispatcher.service - NetworkManager Script Dispatcher Service...
[ OK ] Started NetworkManager-dispatcher.service - NetworkManager Script Dispatcher Service.
[ OK ] Started bluetooth.service - Bluetooth service.
[ OK ] Reached target bluetooth.target - Bluetooth Support.
```



- Maintenant, la seule chose qui reste est d'accéder à notre terminal Pi via UART. Une fois que vous avez connecté la carte UART au Pi (que vous devez garder éteint) et que vous avez connecté la carte UART à votre PC/portable, allumez simplement le Raspberry Pi.
- Pour accéder à la connexion série, vous pouvez faire ce qui suit, en fonction du système :

– **Windows** : Vous devrez effectuer quelques étapes supplémentaires pour les systèmes Windows. D'abord, vous devez télécharger les pilotes spécifiques du module UART (pilotes CP210x). Référez-vous à la vidéo suivante pour cela : [Comment installer les pilotes du module UART sur Windows](#).

Une fois que vous avez fait cela, vous pouvez utiliser **PuTTY** pour accéder à l'interface série. Référez-vous à [ce superbe article](#) pour un guide détaillé pas à pas à ce sujet.

– **Linux/Mac** : Pour les systèmes Linux/Mac, c'est beaucoup plus simple et direct. Une fois que vous connectez le module UART à votre système, vous verrez une entrée pour cela sous le chemin **/dev/cu.usbserial-xxxx** (le xxxx est un nombre, qui était 0001 dans mon cas).

Ouvrez une fenêtre de terminal et tapez la commande suivante pour accéder à la console série :

```
screen /dev/cu.usbserial-0001 115200 -L
```

- Super ! Vous serez maintenant invité à entrer votre nom d'utilisateur et votre mot de passe pour votre Pi. Entrez-les et vous aurez maintenant accès au terminal de votre Pi via UART.



```
screen /dev/ttyUSB0:0001:115200-L
[ 7.394935] EXT4-fs (mmcblk0p2): re-mounted. Quota mode: none.
[ 7.408676] systemd[1]: Mounted sys-kernel-tracing.mount - Kernel Trace File System.
[ 7.419248] systemd[1]: Finished fake-hwclock.service - Restore / save the current clock.
[ 7.430836] systemd[1]: Finished kmod-static-nodes.service - Create List of Static Device Nodes.
[ 7.443347] systemd[1]: modprobe@configfs.service: Deactivated successfully.
[ 7.451507] systemd[1]: Finished modprobe@configfs.service - Load Kernel Module configfs.
[ 7.462975] systemd[1]: modprobe@dm_mod.service: Deactivated successfully.
[ 7.471132] systemd[1]: Finished modprobe@dm_mod.service - Load Kernel Module dm_mod.
[ 7.481823] systemd[1]: modprobe@drm.service: Deactivated successfully.
[ 7.489766] systemd[1]: Finished modprobe@drm.service - Load Kernel Module drm.
[ 7.499779] systemd[1]: modprobe@efi_pstore.service: Deactivated successfully.
[ 7.508318] systemd[1]: Finished modprobe@efi_pstore.service - Load Kernel Module efi_pstore.
[ 7.518665] systemd[1]: Started systemd-journald.service - Journal Service.
[ 7.670792] systemd-journald[288]: Received client request to flush runtime journal.
[ 7.694102] systemd-journald[288]: File /var/log/journal/9b13a55a95d8455aba22173c866abaec/system.journal corrupted or uncleanly shut down, renaming and replacing.

Debian GNU/Linux 12 pi4 ttyS0

pi4 login: raspberrypi
Password:
Linux pi4 6.1.0-rpi7-rpi-v8 #1 SMP PREEMPT Debian 1:6.1.63-1+rpt1 (2023-11-24) aarch64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Jan 23 20:19:30 EST 2024 on tty1
raspberrypi@pi4:~$
```

Cela marque la fin de ce tutoriel. Vous connaissez maintenant les différentes manières d'activer UART sur votre Raspberry Pi et comment utiliser l'interface UART sur votre Pi.

## Téléchargez mon antisèche !

Format PDF pratique, avec les 74 commandes à retenir pour maîtriser son Raspberry Pi.

[Télécharger maintenant](#)

### Ces articles pourraient vous intéresser:

- [25 Idées de Projets Domotiques Incroyables sur Raspberry Pi](#)
- [15 meilleures distributions à utiliser sur Raspberry Pi](#)
- [Mon livre: Maîtriser Raspberry Pi en 30 jours](#)

## Conclusion

Dans cet article, vous avez appris ce qu'est UART et les différentes approches pour activer cette Interface Série sur le Raspberry Pi. Résumons les méthodes que nous avons explorées :

- Via l'**application de configuration Raspberry Pi**.
- Via le **terminal du Pi**.
- **En modifiant les fichiers de démarrage pour activer UART**.

Ensuite, nous avons vu cette interface en action lorsque nous l'avons utilisée pour accéder à la Console Série. Cependant, il existe bien d'autres applications pour cette interface. Par exemple, vous pouvez connecter des capteurs basés sur UART avec votre Pi et les utiliser pour diverses applications.

En général, les possibilités sont infinies pour cette interface de communication simple mais puissante couplée avec le Raspberry Pi.

Ce tutoriel ne marche plus à 100%? [Prévenez-moi](#) pour que je le mette à jour!



**Patrick Fromaget**

Je suis l'auteur principal et le créateur de RaspberryTips.

Mon but est de vous aider sur tous vos problèmes qui concernent Raspberry Pi en publiant des guides détaillés et des tutoriels rapides.

Dans la vraie vie, je suis administrateur système (Linux principalement) avec une solide expérience dans le développement web.

← PRÉCÉDENT

Les Raspberry Pi ont-ils besoin d'antivirus ?

SUIVANT →

Comment installer des polices sur un Raspberry Pi ?

## A lire également

**2 Méthodes Simples pour Faire du Dual Boot sur Raspberry Pi**

**3 façons simples de désactiver IPv6 sur Raspberry Pi**

## Laisser un commentaire

*Votre adresse e-mail ne sera pas publiée. Les champs obligatoires sont indiqués avec \**

Commentaire \*

☐ Enregistrer mon nom, mon e-mail et mon site dans le navigateur pour mon prochain commentaire.

Laisser un commentaire

## Bienvenue



Bonjour, je suis Patrick, le créateur de ce site.

Je souhaite vous aider sur tous vos problèmes qui concernent Raspberry Pi en publiant des guides détaillés et des tutoriels rapides.

Dans la vraie vie, je suis administrateur système (Linux principalement) avec une solide expérience dans le développement web.

Je suis fan du Raspberry Pi et partage avec vous toutes mes découvertes !

À propos du site

## Mon antisèche



Téléchargez mon antisèche au format PDF avec toutes les commandes pour Raspberry Pi ! (c'est cadeau)

[Télécharger maintenant](#)

[À propos](#) [Écrivez pour RaspberryTips](#) [Signaler un problème](#)

[Politique de confidentialité](#) [Contact](#) [English website](#) [Versión española](#)

© 2024 RaspberryTips

Ce site est détenu et exploité par la société [WebInpact](#), 12 Rue de la Part-Dieu, 69003 Lyon.

Raspberry Pi est une marque déposée de la Fondation Raspberry Pi.

RaspberryTips.fr participe au programme d'affiliation Amazon, un programme de publicité pour les affiliés.

En tant que Partenaire Amazon, je réalise un bénéfice sur les achats remplissant les conditions requises.

Ce site participe également à d'autres programmes d'affiliation et est rémunéré pour le trafic et les affaires de ces sociétés.