

Best Tech Tips

Best Tech Tips (/fr) / Divers (/fr/parts/40-miscellanea)

/ Référence de brochage ESP32 - Un guide complet (/fr/chapters/13091-esp32-pinout-reference--a-complete-guide)

Référence de brochage ESP32 - Un guide complet

(/fr/chapters/13091-esp32-pinout-reference--a-complete-guide)

Catégorie Divers (/fr/parts/40-miscellanea) | 📅 April 06, 2023 16:25

L'ESP32 est une carte à microcontrôleur dotée d'un certain nombre de broches GPIO à plusieurs fins. Chacune de ces broches est conçue pour des fonctions spécifiques. ESP32 comporte un plus grand nombre de broches par rapport aux cartes Arduino UNO ou ESP8266. Pour commencer à travailler avec ESP32, une connaissance suffisante de sa broche est essentielle. L'objectif de ce guide est de discuter de toutes les broches disponibles sur le tableau et de leurs fonctionnalités associées.

Ce guide de brochage pour ESP32 contient le contenu suivant :

1: Présentation de l'ESP32

- 1.1: Brochage ESP32
- 1.2: carte de version ESP32 36 broches
- 1.3: carte de version ESP32 36 broches
- 1.4: Quelle est la différence ?

2: broches GPIO ESP32

- 2.1: Broches d'entrée/sortie
- 2.2: Broches d'entrée uniquement
- 2.3: Broches d'interruption
- 2.4: Broches RTC

3: Broches ADC ESP32

- 3.1: Brochage ESP32 ADC
- 3.2: Broche ADC du canal 1
- 3.3: Broche ADC du canal 2
- 3.4: Comment utiliser ESP32 ADC
- 3.5: Limitation ADC sur ESP32

4: Broches CNA

5: Broches PWM

6: Broches SPI dans ESP32

7: Broches I2C

8: Broches I2S

9: UART

10: Broches tactiles capacitatives

11: Goupilles de cerclage ESP32

12: Épingles hautes au BOOT

13: Activer le code PIN (EN)

14: broches d'alimentation ESP32

15: Capteur à effet Hall ESP32

Avant d'aller de l'avant ici, nous avons résumé une brève introduction à la carte ESP32 IoT.

1: Présentation de l'ESP32

- ESP32 est une carte microcontrôleur basée sur l'IoT très populaire.
- La partie principale de cette carte microcontrôleur est une puce Tensilica Xtensa LX6 conçue par Espressif Systems.
- Il contient un processeur double cœur et chacun de ces cœurs peut être contrôlé séparément.
- Un total de 48 broches sont présentes dans la puce ESP32, mais toutes ces broches ne sont pas exposées aux utilisateurs.
- L'ESP32 est disponible en deux versions différentes: 30 broches et 36 broches.
- L'ESP32 peut monter jusqu'à une fréquence allant de 80 MHz à 240 MHz.
- Il contient un ULP spécial (Ultra Low Power Co-Processor) qui économise une grande quantité d'énergie en utilisant très moins d'énergie lorsque le processeur principal est éteint.
- Il contient le WiFi embarqué et un double module Bluetooth.
- L'ESP32 est moins cher que les autres microcontrôleurs.

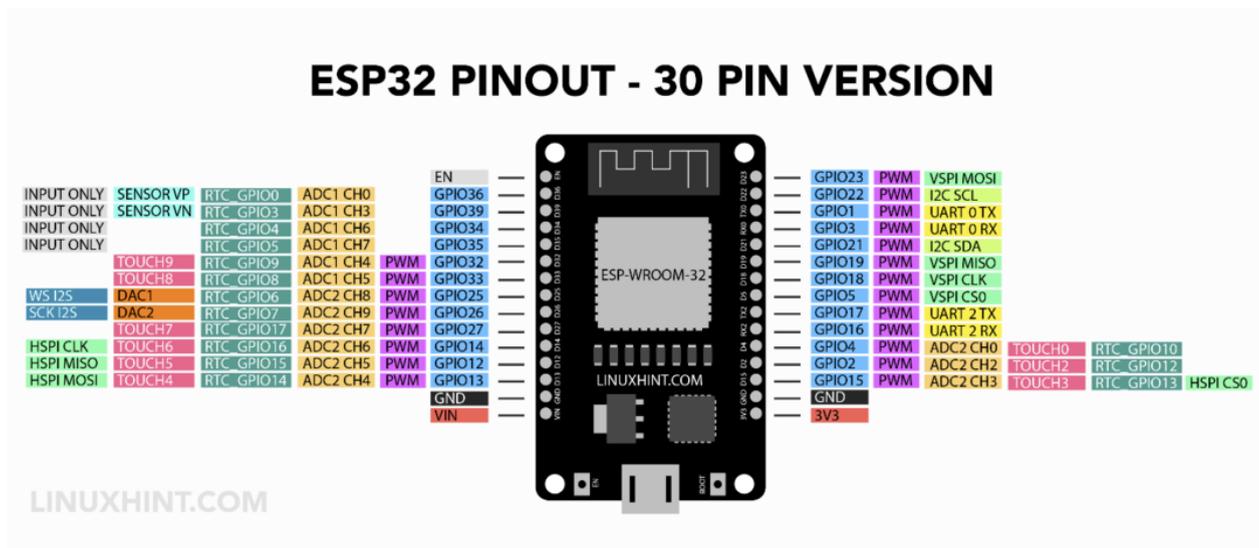
1.1: Brochage ESP32

Plusieurs variantes d'ESP32 sont disponibles sur le marché, aujourd'hui nous couvrirons le brochage détaillé du Variante à 30 broches fournie avec le microcontrôleur ESP32-WROOM-32 parfois également appelé WROOM32.

Un total de 48 broches sont disponibles dans les puces ESP32 parmi lesquelles 30 broches sont exposées à l'utilisateur tandis que d'autres sont intégrées à l'intérieur du microcontrôleur; certaines cartes contiennent également six broches intégrées flash SPI supplémentaires qui résument le nombre total de broches à 36.

1.2: carte de version ESP32 30 broches

L'image ci-dessous représente le brochage détaillé de la variante ESP32 à 30 broches contenant tous ses périphériques dont nous discutons un par un en détail.



Certains périphériques principaux à l'intérieur de l'ESP32 sont :

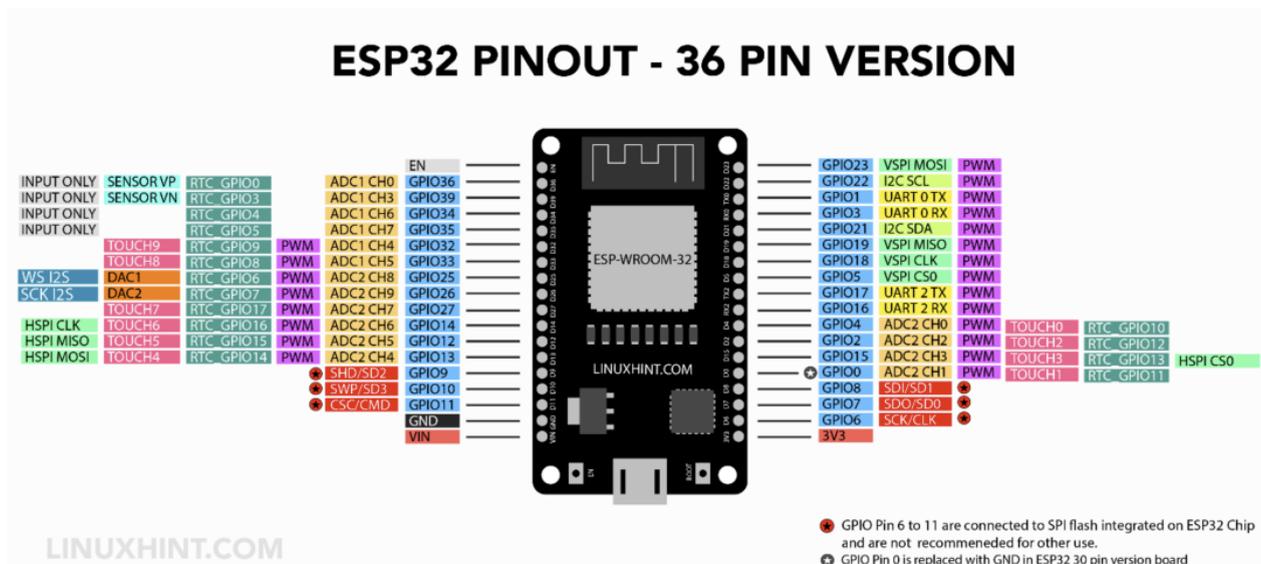
- Total 48 broches*
- 18 broches ADC 12 bits
- Deux broches DAC 8 bits
- 16 canaux PWM

- 10 broches tactiles capacitives
- 3 UART
- 2 I2C
- 1 CAN
- 2 I2S
- 3SPI

* La puce ESP32 contient un total de 48 broches dont seulement 30 broches sont disponibles pour l'interfaçage externe (dans certains cartes 36 qui incluent 6 broches SPI supplémentaires) les 18 broches restantes sont intégrées à l'intérieur de la puce pour la communication but.

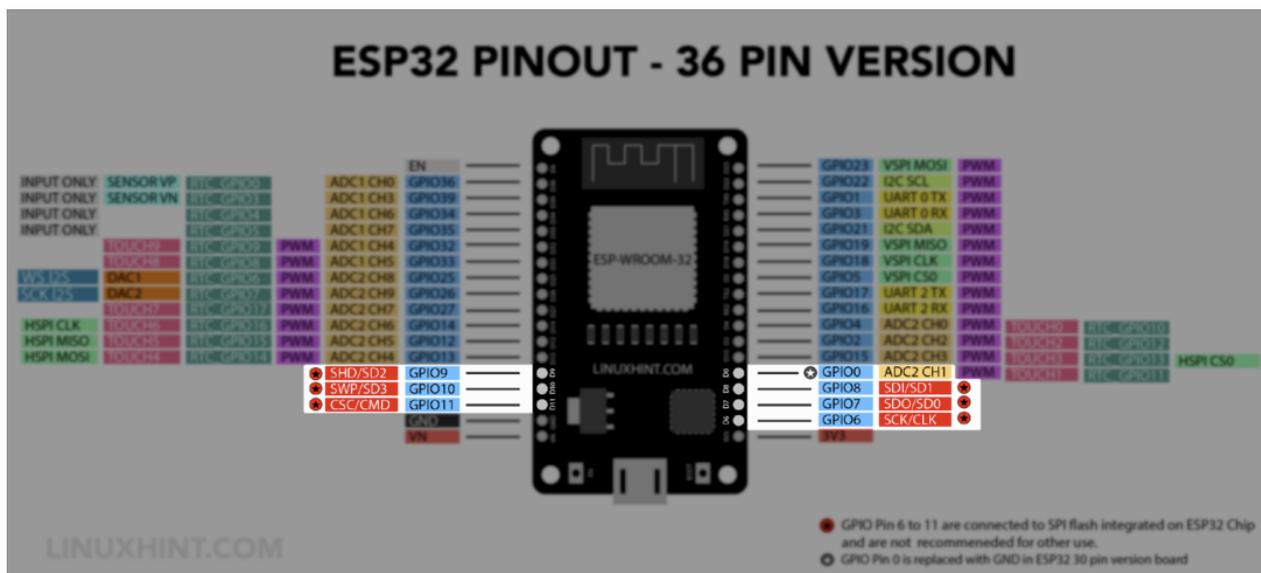
1.3: carte de version ESP32 36 broches

Voici une image d'une carte ESP32 ayant un total de 36 broches.



1.4: Différence entre la version ESP32 30 broches et la version ESP32 36 broches

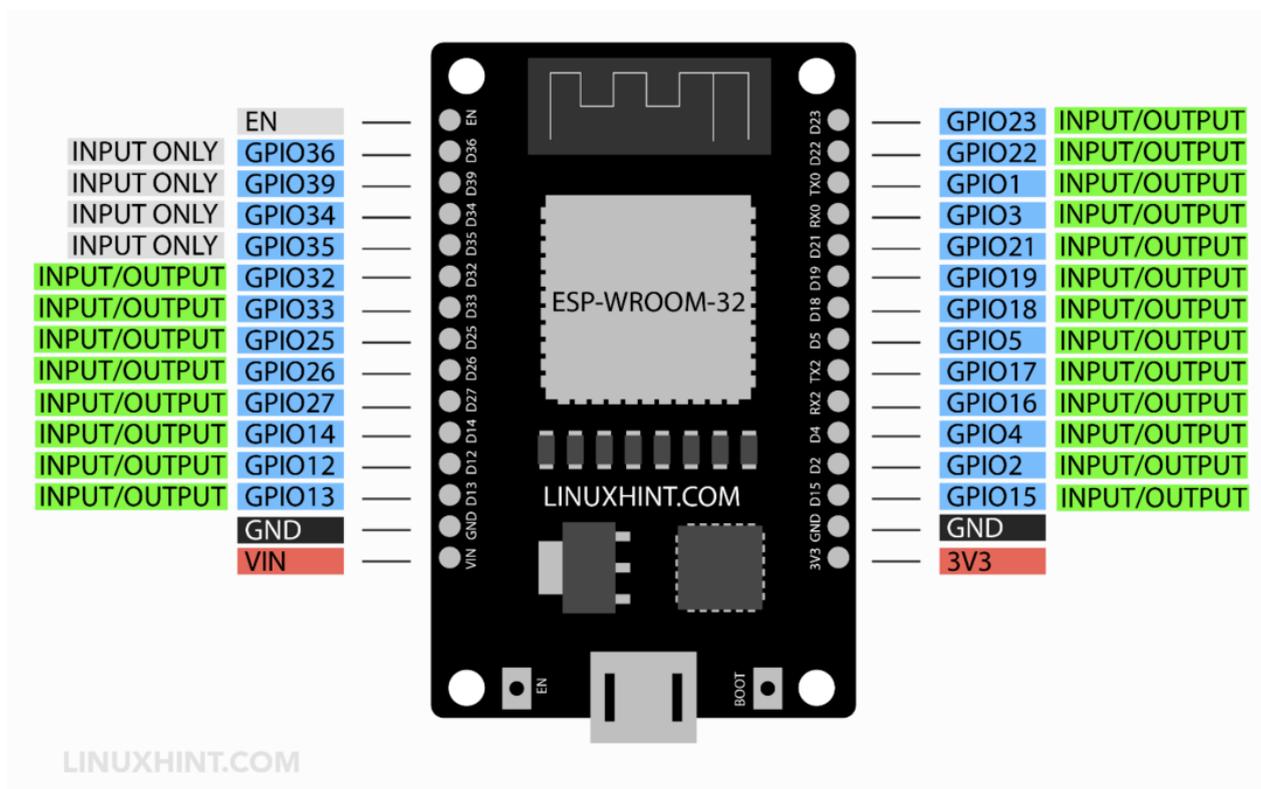
Les deux cartes ESP32 partagent les mêmes spécifications, la seule différence majeure ici est 6 broches supplémentaires qui sont exposées dans la carte ESP32 (36 broches) sont des broches intégrées flash SPI et deuxièmement, le GPIO 0 est remplacé par la broche GND dans la carte ESP32 (30 broches), ce qui entraîne l'absence de **Touchez 1** et **ADC2 CH1** broche.



2: broches GPIO ESP32

Comme mentionné précédemment, ESP32 a un total de 48 broches dont seulement 30 broches sont accessibles aux

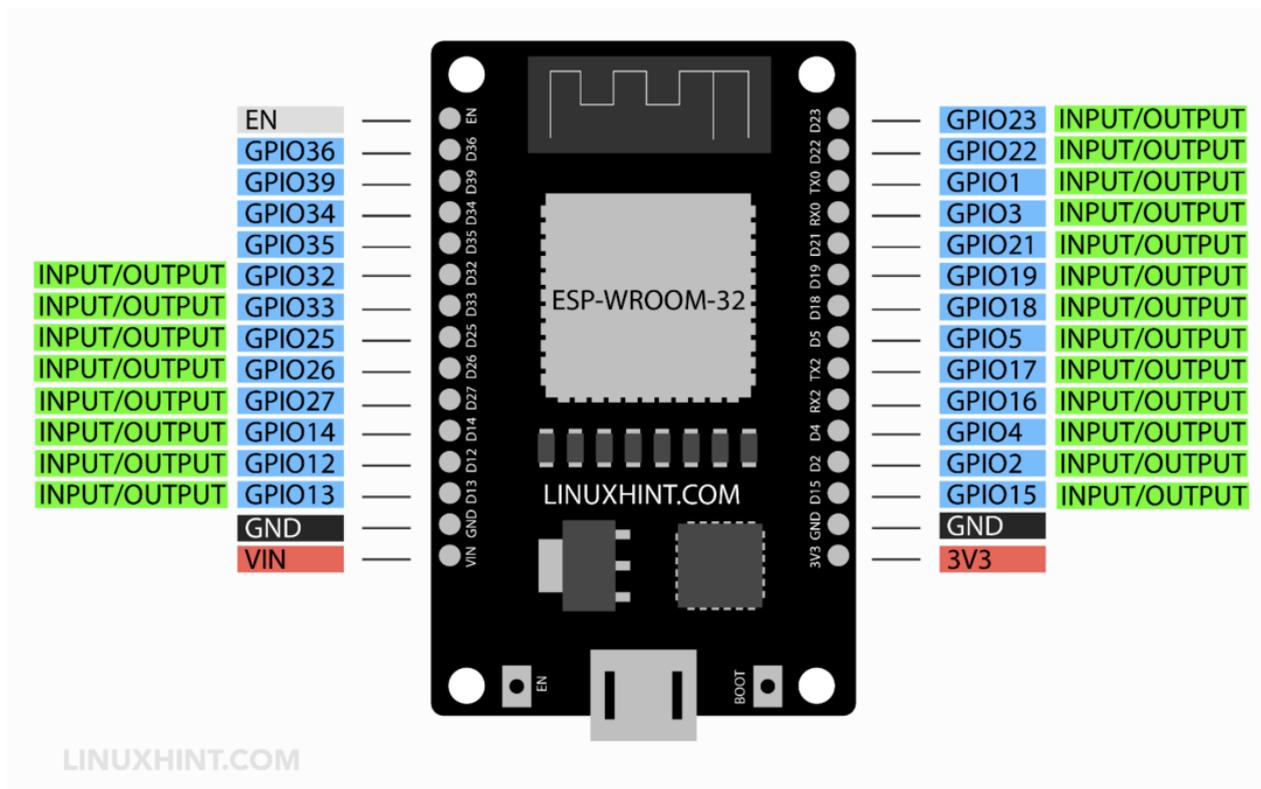
utilisateurs. Chacune de ces 30 broches d'entrée-sortie à usage général a une fonction spécifique et peut être configurée à l'aide d'un registre spécifique. Il existe différentes broches GPIO comme UART, PWM, ADC et DAC.



Sur ces 30 broches, certaines sont alimentées tandis que d'autres peuvent être configurées à la fois en entrée et en sortie alors que certaines broches sont uniquement en entrée.

2.1: Broches d'entrée/sortie

Presque toutes les broches GPIO peuvent être configurées comme entrée et sortie, à l'exception des 6 broches flash de l'interface périphérique série (SPI) qui ne peuvent pas être configurées à des fins d'entrée ou de sortie. Ces 6 pins SPI sont disponibles sur la carte version 36 pins.



Le tableau ci-dessous explique l'état des broches ESP32 GPIO pouvant être utilisées comme entrée et sortie :

Ici **D'ACCORD** signifie que la broche correspondante peut être utilisée comme entrée ou sortie.

BROCHE GPIOAISIR		SORTIR	Description
GPIO 0	Tiré vers le haut	D'ACCORD	Sortie PWM au démarrage
GPIO 1	Broche de transmission	D'ACCORD	Sortie de débogage au démarrage
GPIO 2	D'ACCORD	D'ACCORD	LED embarquée
GPIO 3	D'ACCORD	Broche de réception	Élevé au démarrage
GPIO 4	D'ACCORD	D'ACCORD	–
GPIO 5	D'ACCORD	D'ACCORD	Sortie PWM au démarrage
GPIO 6	–	–	Broche Flash SPI
GPIO 7	–	–	Broche Flash SPI
GPIO 8	–	–	Broche Flash SPI
GPIO 9	–	–	Broche Flash SPI
GPIO 10	–	–	Broche Flash SPI
GPIO 11	–	–	Broche Flash SPI
GPIO 12	D'ACCORD	D'ACCORD	Échec du démarrage à haute traction
GPIO 13	D'ACCORD	D'ACCORD	–
GPIO 14	D'ACCORD	D'ACCORD	Sortie PWM au démarrage
GPIO 15	D'ACCORD	D'ACCORD	Sortie PWM au démarrage
GPIO 16	D'ACCORD	D'ACCORD	–
GPIO 17	D'ACCORD	D'ACCORD	–
GPIO 18	D'ACCORD	D'ACCORD	–
GPIO 19	D'ACCORD	D'ACCORD	–
GPIO 21	D'ACCORD	D'ACCORD	–
GPIO 22	D'ACCORD	D'ACCORD	–
GPIO 23	D'ACCORD	D'ACCORD	–
GPIO 25	D'ACCORD	D'ACCORD	–
GPIO 26	D'ACCORD	D'ACCORD	–
GPIO 27	D'ACCORD	D'ACCORD	–
GPIO 32	D'ACCORD	D'ACCORD	–
GPIO 33	D'ACCORD	D'ACCORD	–
GPIO 34	D'ACCORD	Entrée uniquement	
GPIO 35	D'ACCORD	Entrée uniquement	
GPIO 36	D'ACCORD	Entrée uniquement	
GPIO 39	D'ACCORD	Entrée uniquement	

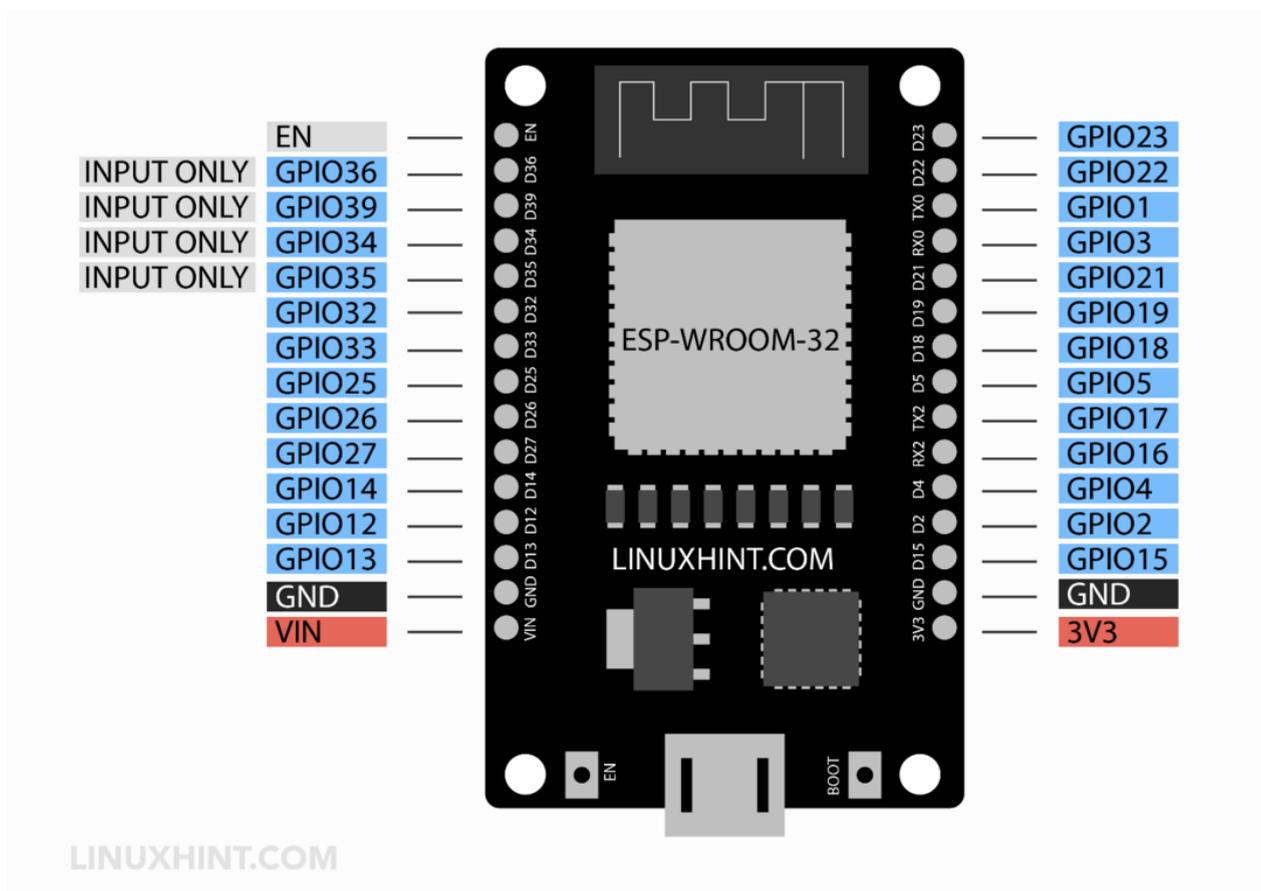
2.2: Broches d'entrée uniquement

Les broches GPIO 34 à 39 ne peuvent pas être configurées comme sortie car elles sont uniquement à des fins d'entrée. Cela est dû à l'absence de résistance interne pull up ou pull down et ne peut donc être utilisé que comme entrée.

De plus, GPIO 36(VP) et GPIO 39(VN) sont utilisés pour les préamplificateurs à très faible bruit dans ESP32 ADC.

Pour résumer, voici les broches d'entrée uniquement dans ESP32 :

- GPIO 34
- GPIO 35
- GPIO 36
- GPIO 39

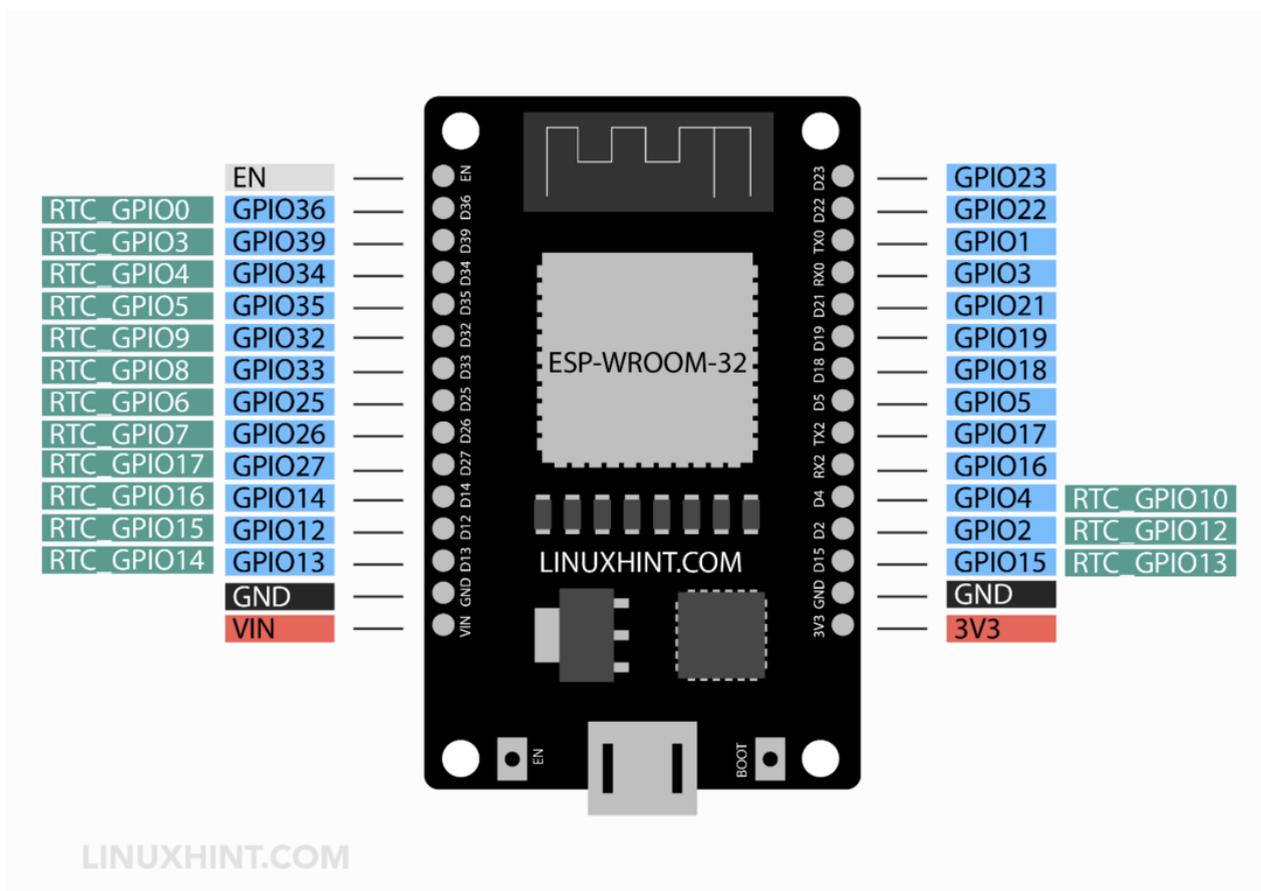


2.3: Broches d'interruption

Toutes les broches GPIO dans ESP32 peuvent prendre des interruptions externes. Cela permet de surveiller les changements à une interruption spécifique au lieu de surveiller en permanence.

2.4: Broches RTC

L'ESP32 possède également des broches GPIO RTC. Ces broches RTC permettent à ESP32 de fonctionner en mode veille profonde. Lorsque l'ESP32 est en mode veille profonde lors de l'exécution d'un coprocesseur ultra-basse consommation (ULP), ces broches RTC peuvent réveiller l'ESP32 de la veille profonde en économisant un grand pourcentage d'énergie.



Ces broches GPIO RTC peuvent agir comme une source d'excitation externe pour réveiller l'ESP32 du sommeil profond à un moment donné ou interrompre. Les broches GPIO RTC comprennent :

- RTC_GPIO0 (GPIO36)
- RTC_GPIO3 (GPIO39)
- RTC_GPIO4 (GPIO34)
- RTC_GPIO5 (GPIO35)
- RTC_GPIO6 (GPIO25)
- RTC_GPIO7 (GPIO26)
- RTC_GPIO8 (GPIO33)
- RTC_GPIO9 (GPIO32)
- RTC_GPIO10 (GPIO4)
- RTC_GPIO11 (GPIO0)
- RTC_GPIO12 (GPIO2)
- RTC_GPIO13 (GPIO15)
- RTC_GPIO14 (GPIO13)
- RTC_GPIO15 (GPIO12)
- RTC_GPIO16 (GPIO14)
- RTC_GPIO17 (GPIO27)

3: Broches ADC ESP32

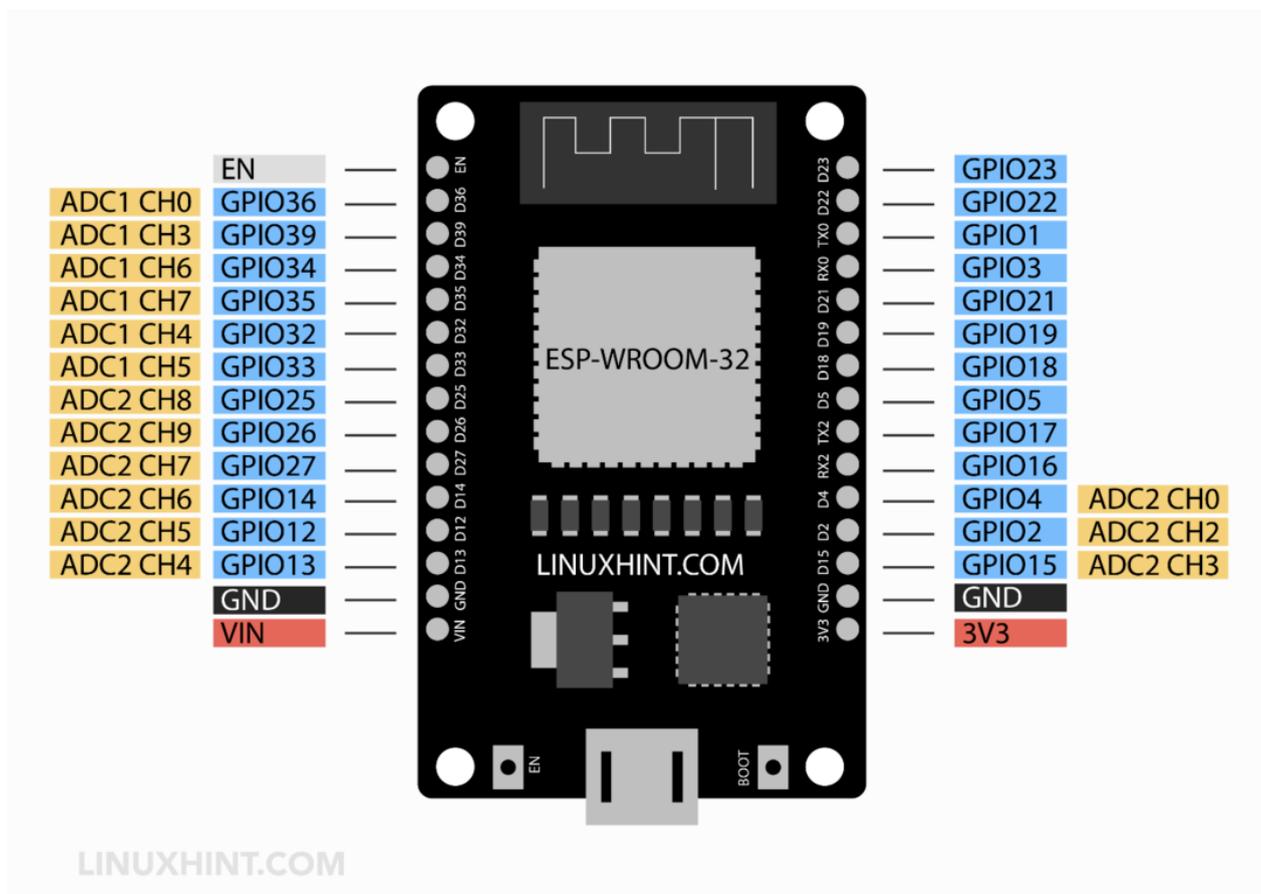
La carte ESP32 dispose de deux ADC 12 bits intégrés également appelés ADC SAR (Successive Approximation Registers). La carte ESP32 Les ADC prennent en charge 18 canaux d'entrée analogiques différents, ce qui signifie que nous pouvons connecter 18 capteurs analogiques différents pour prendre l'entrée de eux.

Mais ce n'est pas le cas ici; ces canaux analogiques sont divisés en deux catégories canal 1 et canal 2, ces deux canaux ont des broches qui ne sont pas toujours disponibles pour l'entrée ADC. Voyons ce que sont ces broches ADC avec d'autres.

3.1: Brochage ESP32 ADC

Comme mentionné précédemment, la carte ESP32 dispose de 18 canaux ADC. Sur 18 seulement 15 sont disponibles dans la carte DEVKIT V1 DOIT ayant un total de 30 GPIO.

Jetez un œil à votre carte et identifiez les broches ADC comme nous les avons mises en évidence dans l'image ci-dessous :



3.2: Broche ADC du canal 1

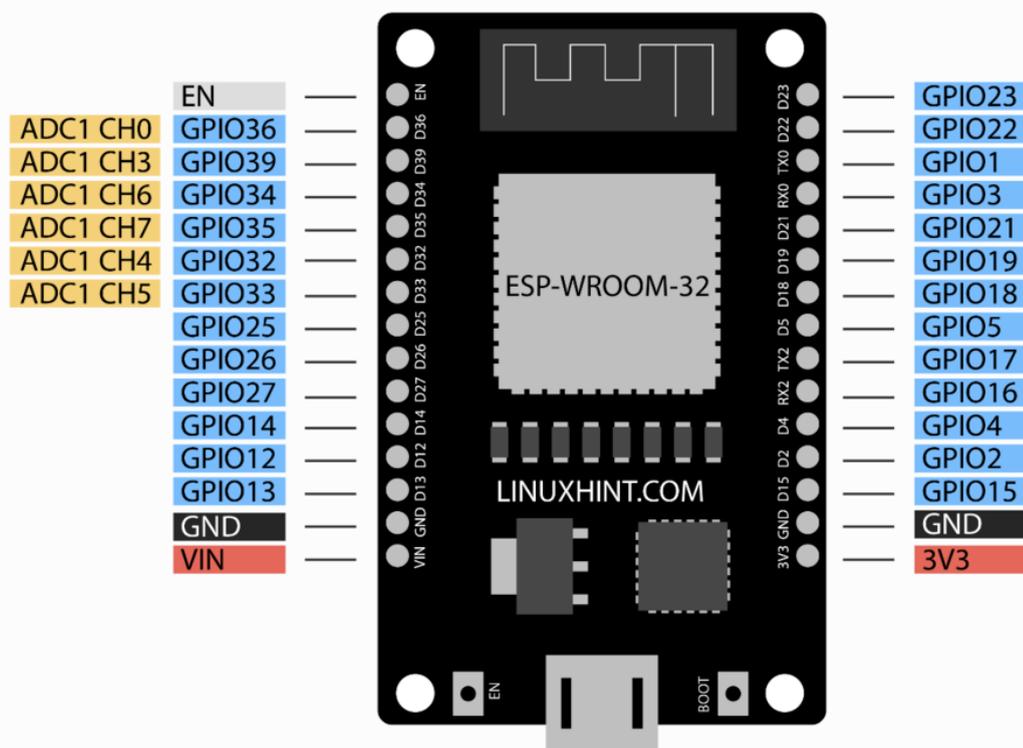
Voici le mappage des broches donné de la carte ESP32 DEVKIT DOIT. ADC1 dans ESP32 a 8 canaux, mais la carte DOIT DEVKIT ne prend en charge que 6 canaux. Mais je vous garantis que ceux-ci sont encore plus que suffisants.

ADC1BROCHE GPIO ESP32

- CH0 36
- CH1 37* (NA)
- CH2 38* (NA)
- CH3 39
- CH4 32
- CH5 33
- CH6 34
- CH7 35

*Ces broches ne sont pas disponibles pour l'interfaçage externe; ceux-ci sont intégrés dans les puces ESP32.

L'image suivante montre les canaux ESP32 ADC1 :



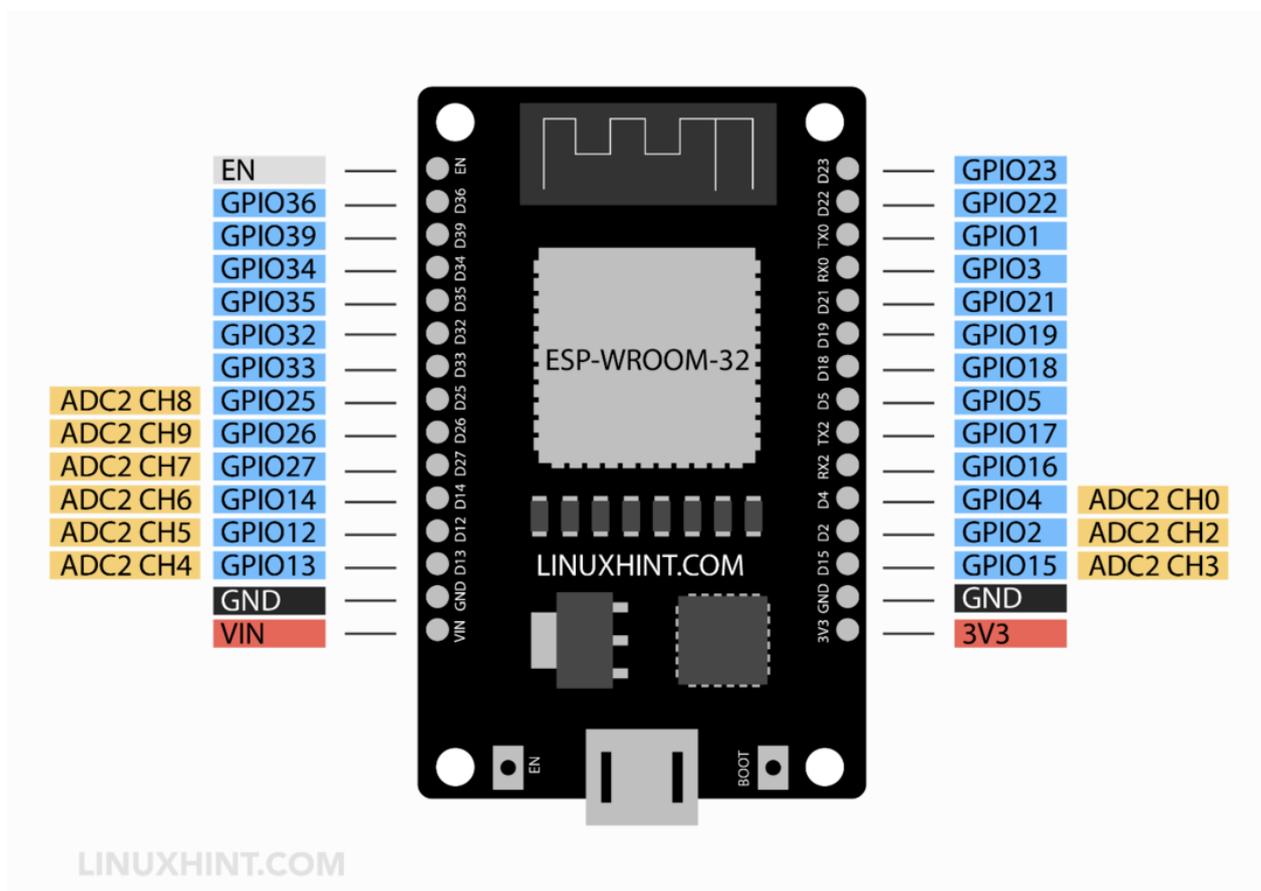
3.3: Broche ADC du canal 2

Les cartes DEVKIT DOIT ont 10 canaux analogiques en ADC2. Bien que l'ADC2 dispose de 10 canaux analogiques pour lire les données analogiques, ces canaux ne sont pas toujours disponibles. L'ADC2 est partagé avec les pilotes WiFi intégrés, ce qui signifie qu'au moment où la carte utilise le WIFI, ces ADC2 ne seront pas disponibles. La solution à ce problème consiste à utiliser ADC2 uniquement lorsque le pilote Wi-Fi est désactivé.

ADC2BROCHE GPIO ESP32

- CH0 4
- CH1 0 (NA en version 30 broches ESP32-Devkit DOIT)
- CH2 2
- CH3 15
- CH4 13
- CH5 12
- CH6 14
- CH7 27
- CH8 25
- CH9 26

L'image ci-dessous montre le mappage des broches du canal ADC2.



3.4: Comment utiliser ESP32 ADC

L'ADC ESP32 fonctionne de la même manière qu'Arduino, à la seule différence qu'il dispose d'un ADC 12 bits. Ainsi, la carte ESP32 cartographie les valeurs de tension analogiques allant de 0 à 4095 en valeurs discrètes numériques.



- Si la tension donnée à ESP32 ADC est nulle, un canal ADC, la valeur numérique sera nulle.
- Si la tension donnée à ADC est maximale signifie 3,3 V, la valeur numérique de sortie sera égale à 4095.
- Pour mesurer une tension plus élevée, nous pouvons utiliser la méthode du diviseur de tension.

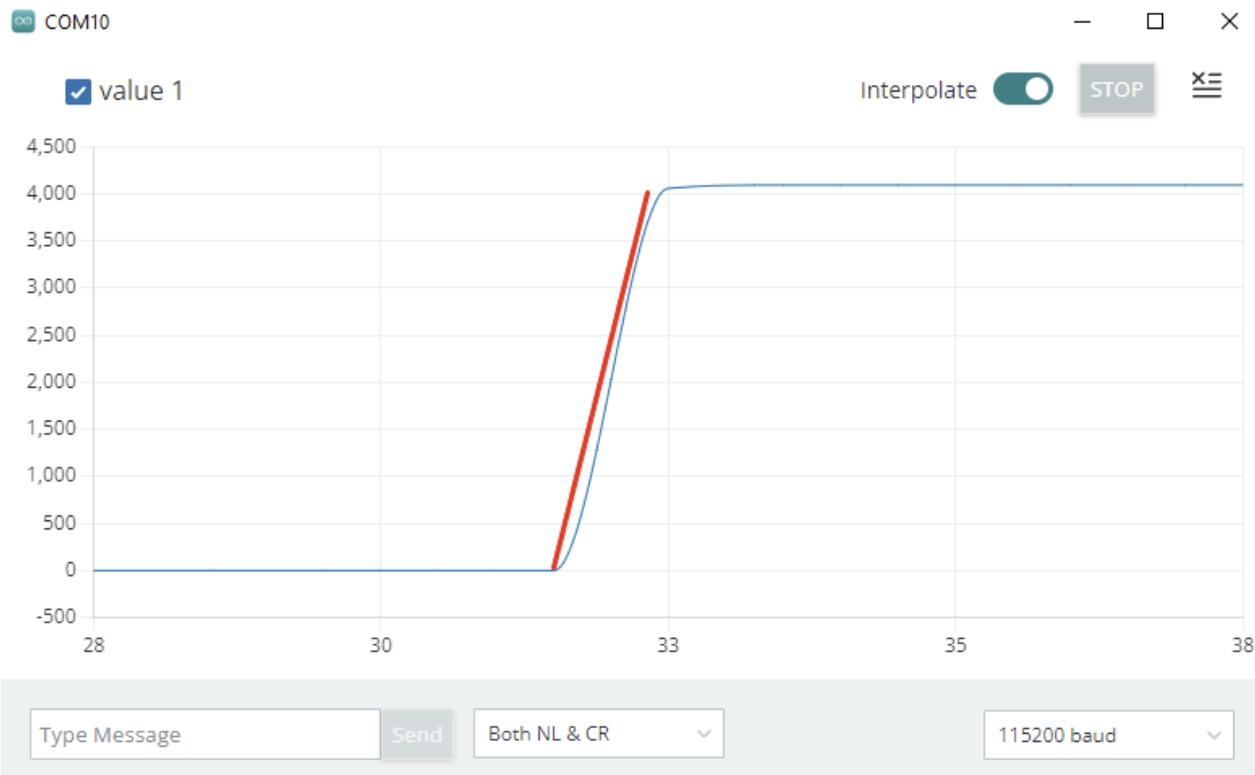
Note: ESP32 ADC est défini par défaut sur 12 bits, mais il est possible de le configurer en 0 bit, 10 bits et 11 bits. L'ADC 12 bits par défaut peut mesurer la valeur $2^{12}=4096$ et la tension analogique varie de 0V à 3,3V.

3.5: Limitation ADC sur ESP32

Voici quelques limitations de l'ESP32 ADC :

- ESP32 ADC ne peut pas mesurer directement une tension supérieure à 3,3 V.
- Lorsque les pilotes Wi-Fi sont activés, ADC2 ne peut pas être utilisé. Seuls 8 canaux d'ADC1 peuvent être utilisés.
- L'ESP32 ADC n'est pas très linéaire; ça montre **non-linéarité** comportement et ne peut pas faire la distinction entre 3,2 V et 3,3 V. Cependant, il est possible de calibrer ESP32 ADC. Ici est un article qui vous guidera pour calibrer le comportement de non-linéarité ESP32 ADC.

Le comportement de non-linéarité d'ESP32 peut être vu sur le moniteur série d'Arduino IDE.

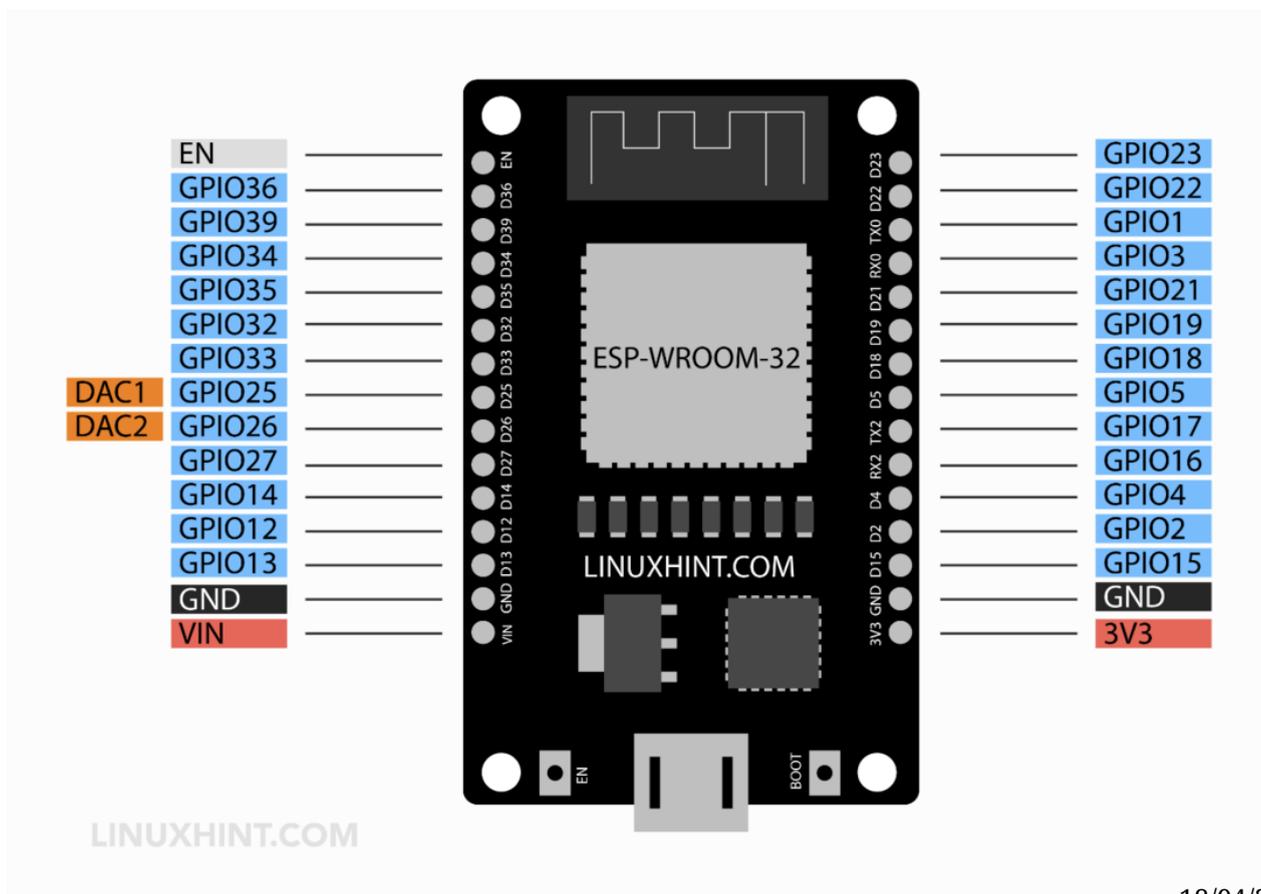


4: Broches CNA

ESP32 dispose de deux à bord **CNA 8 bits** (Convertisseur numérique analogique). En utilisant les broches ESP32 DAC, tout signal numérique peut être transformé en analogique. L'application des broches DAC inclut la tension et le contrôle PWM.

Voici les deux broches DAC de la carte ESP32.

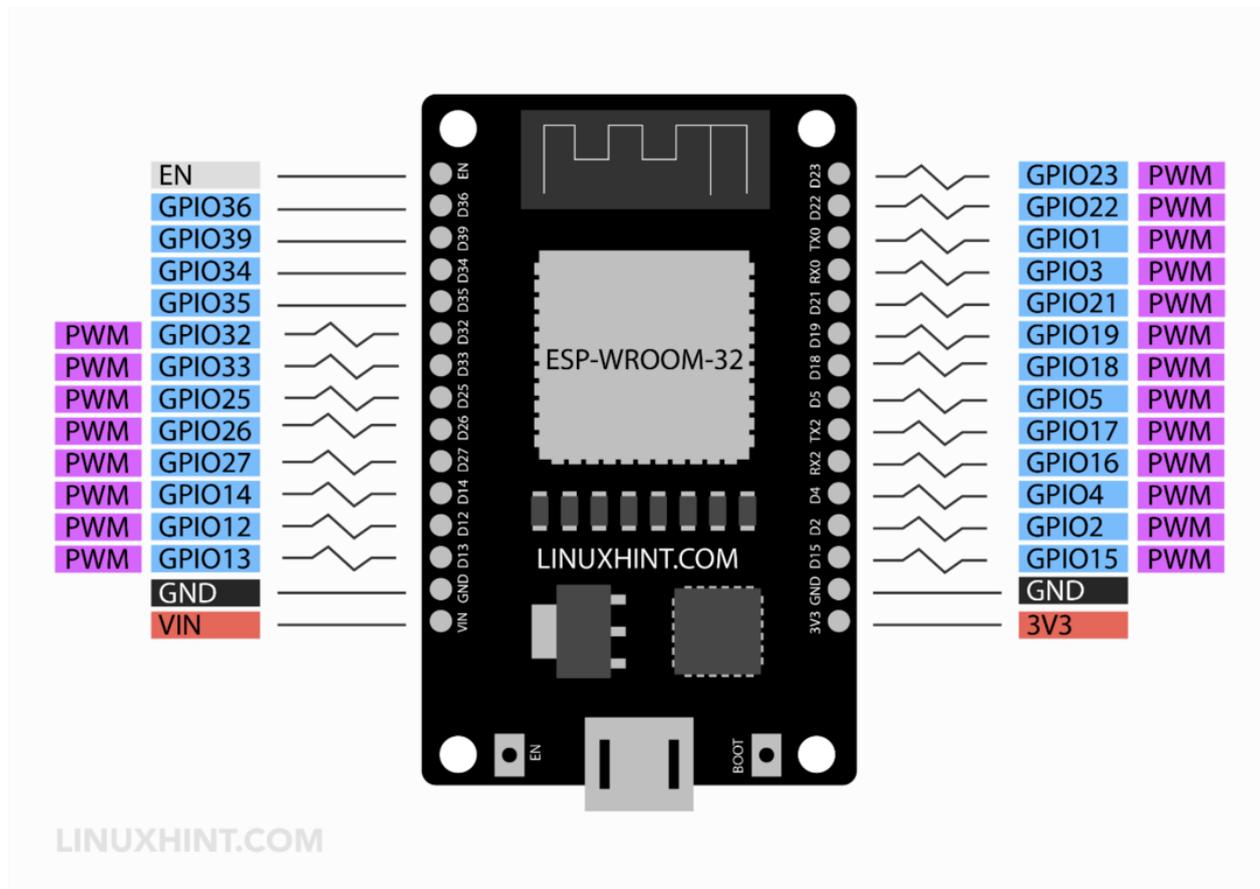
- DAC_1 (GPIO25)
- DAC_2 (GPIO26)



5: Broches PWM

La carte ESP32 contient 16 canaux de modulation de largeur d'impulsion (PWM) indépendants qui peuvent émettre différents signaux PWM. Presque tous les GPIO peuvent générer un signal PWM, mais l'entrée ne contient que des broches **34,35,36,39** ne peuvent pas être utilisées comme broches PWM car elles ne peuvent pas émettre de signal.

Note: Dans ESP32 36 broches, les 6 broches intégrées flash SPI intégrées (GPIO 6, 7, 8, 9, 10, 11) ne peuvent pas être utilisées comme PWM.



Lisez ici un guide complet pour débutants pour contrôler Broches ESP32 PWM utilisant Arduino IDE.

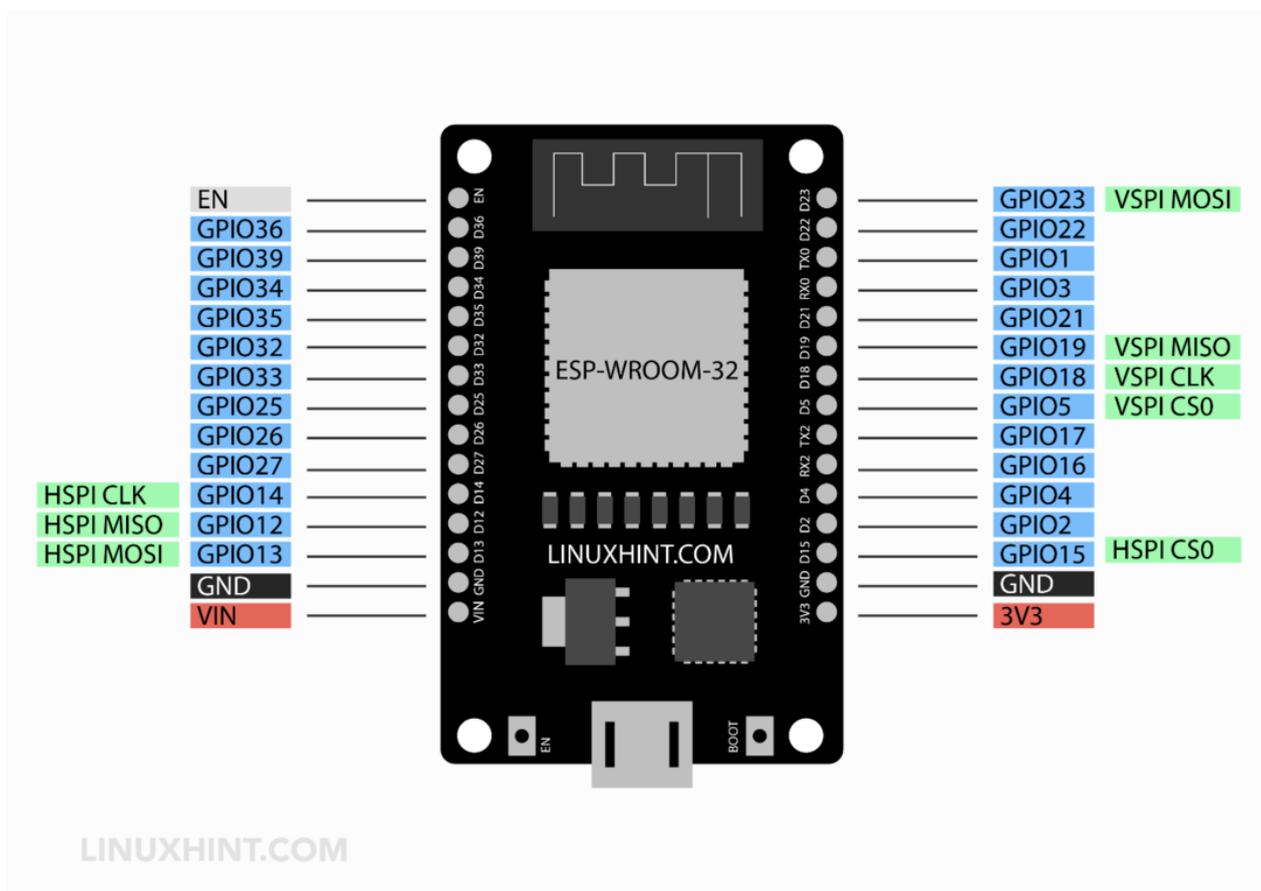
6: Broches SPI dans ESP32

L'ESP32 dispose de quatre périphériques SPI intégrés dans son microcontrôleur :

- **SPI0** : Ne peut pas être utilisé en externe uniquement pour la communication interne.
- **SPI1** : Ne peut pas être utilisé en externe avec des appareils SPI. Uniquement pour la communication de la mémoire interne
- **SPI2** : SPI2 ou HSPI peuvent communiquer avec des appareils et des capteurs externes. Il a des signaux de bus indépendants avec chaque bus capable de contrôler **3** appareils esclaves.
- **SPI3** : SPI3 ou VSPI peuvent communiquer avec des appareils et des capteurs externes. Il a des signaux de bus indépendants avec chaque bus capable de contrôler **3** appareils esclaves.

La plupart des cartes ESP32 sont livrées avec des broches SPI préassignées pour SPI2 et SPI3. Cependant, s'ils ne sont pas attribués, nous pouvons toujours attribuer des broches SPI dans le code. Voici les broches SPI trouvées dans la plupart des cartes ESP32 qui sont préaffectées :

Interface	SPIMOSI	MISO	SCLK	CS
VSPI	GPIO 23	GPIO 19	GPIO 18	GPIO 5
HSPI	GPIO 13	GPIO 12	GPIO 14	GPIO 15



Les broches SPI mentionnées ci-dessus peuvent varier en fonction du type de carte. Nous allons maintenant écrire un code pour vérifier les broches ESP32 SPI à l'aide de l'IDE Arduino.

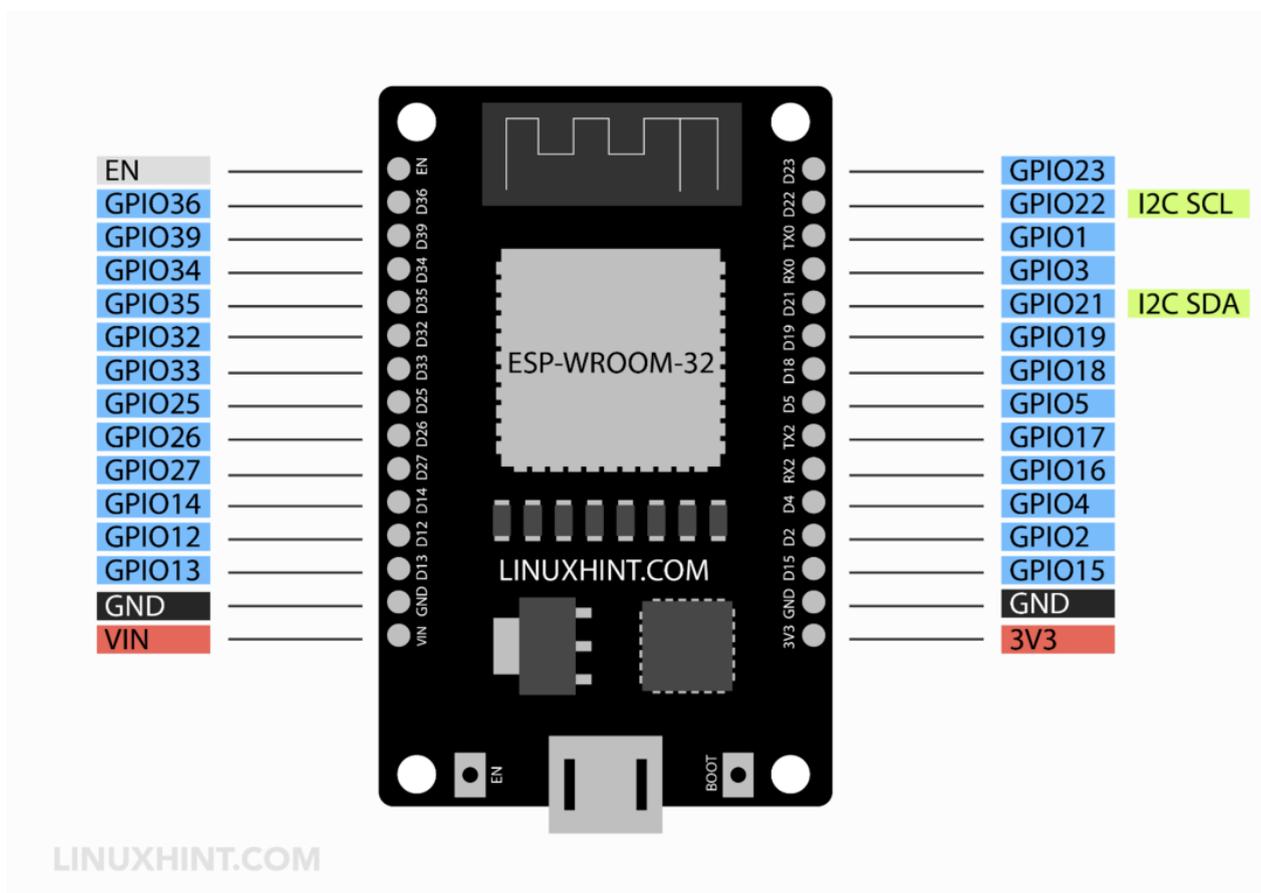
Pour un didacticiel complet sur l'interface périphérique série, cliquez sur [ici](#).

7: Broches I2C

La carte ESP32 est livrée avec un seul bus I2C qui prend en charge jusqu'à 120 appareils I2C. Par défaut, deux broches SPI pour SDA et SCL sont définies respectivement sur GPIO 21 et 22. Cependant en utilisant la commande **wire.begin (SDA, SCL)** nous pouvons configurer n'importe quel GPIO comme interface I2C.

Les deux broches GPIO suivantes sont définies par défaut pour I2C :

- GPIO21 - SDA (broche de données)
- GPIO22 - SCL (broche de synchronisation d'horloge)



8: Broches I2S

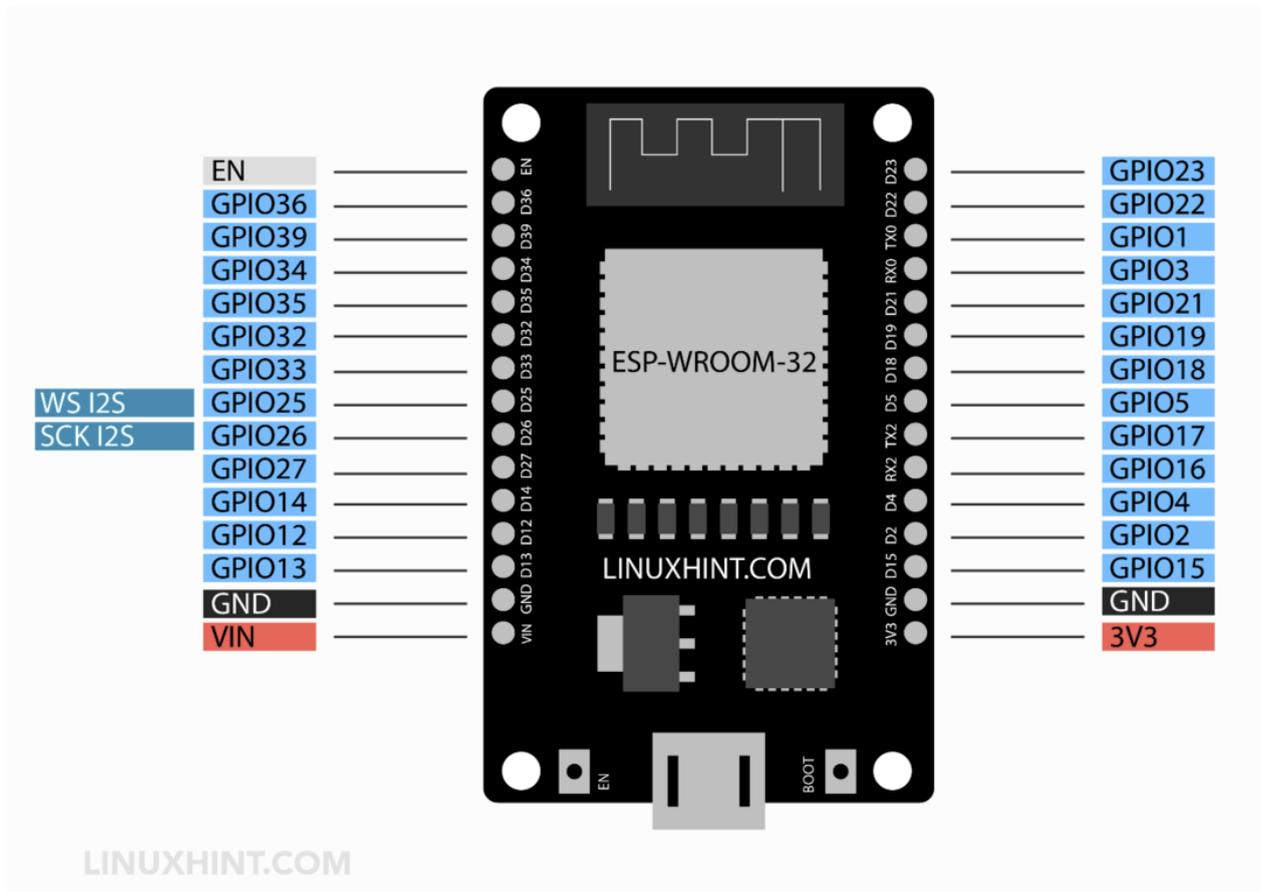
I2S (Inter-IC Sound) est un protocole de communication synchrone qui transmet en série des signaux audio entre deux appareils audio numériques.

L'ESP32 dispose de deux périphériques I2S, chacun d'eux fonctionnant en mode de communication semi-duplex mais nous pouvons également les combiner pour fonctionner en mode duplex intégral.

Normalement, les deux broches DAC dans ESP32 sont utilisées pour la communication audio I2S. Voici les broches I2S dans ESP32 :

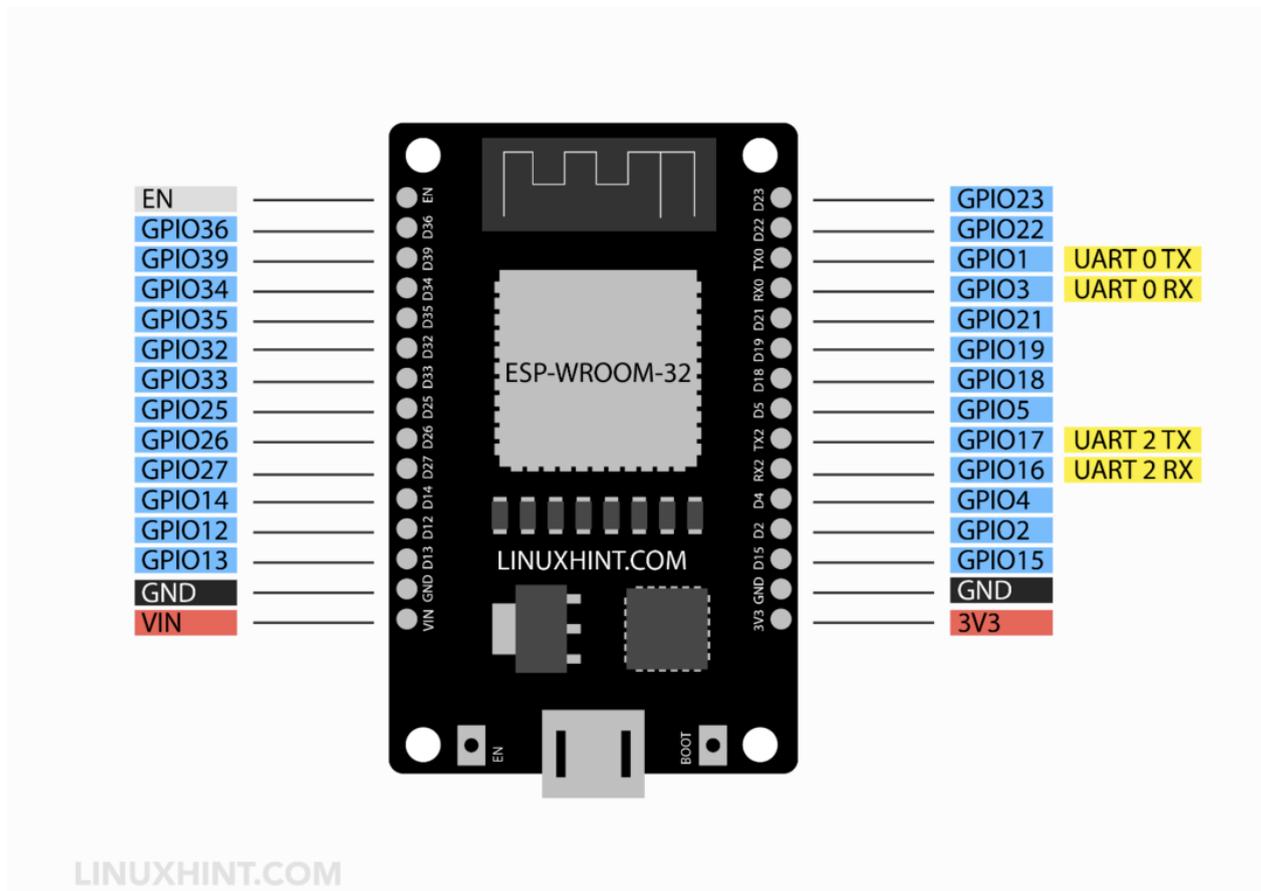
- GPIO 26 – Horloge série (SCK)
- GPIO 25 – Sélection de mot (WS)

Pour les broches I2S Serial Data (SD), nous pouvons configurer n'importe quelle broche GPIO.



9: UART

Par défaut, ESP32 possède trois interfaces UART qui sont UART0, UART1 et UART2. UART0 et UART2 sont utilisables en externe, mais l'UART1 n'est pas disponible pour l'interfaçage et la communication externes car il est connecté en interne à la mémoire flash SPI intégrée.



au convertisseur USB-série et est utilisée par ESP32 pour la communication série via le port USB. Si nous utilisons des broches UART0, nous ne pourrions pas communiquer avec le PC. Par conséquent, il n'est pas recommandé d'utiliser les broches UART0 en externe.

- UART2, d'autre part, n'est pas connecté en interne au convertisseur USB-série, ce qui signifie que nous pouvons l'utiliser pour l'interfaçage externe pour la communication UART entre les appareils et les capteurs.
- UART1, comme mentionné précédemment, est connecté en interne à la mémoire flash, n'utilisez donc pas les broches GPIO 9 et 10 pour la communication UART externe.

Note: La puce ESP32 a une capacité de multiplexage, ce qui signifie que différentes broches peuvent également être utilisées pour les communications comme nous pouvons configurer n'importe quelle broche GPIO dans ESP32 pour la communication UART1 en la définissant à l'intérieur de l'Arduino code.

Voici les broches UART d'ESP32 :

Autobus UART	Rx	Émission	Description
UART0	GPIO 3	GPIO 1	Peut être utilisé mais non recommandé car connecté en interne au convertisseur USB-série
UART1	GPIO 9	GPIO 10	Ne pas utiliser connecté à la mémoire Flash ESP32 interne SPI
UART2	GPIO 16	GPIO 17	Autorisé à utiliser

10: Broches tactiles capacitives

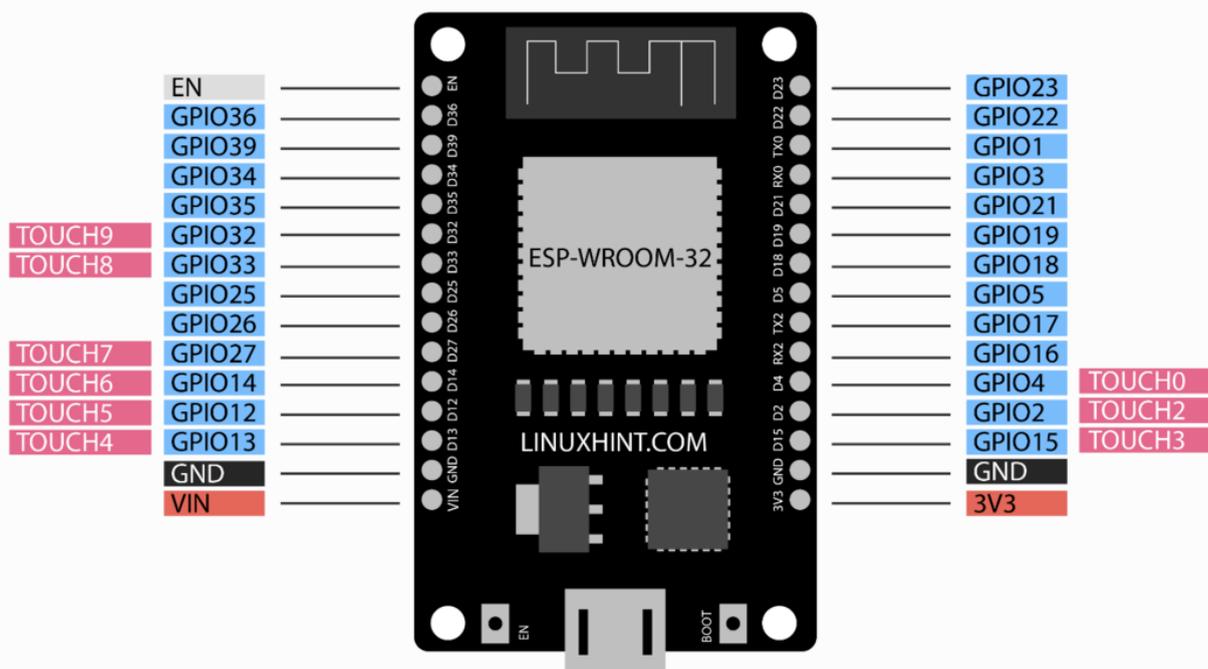
L'ESP32 possède 10 broches GPIO qui prennent en charge les capteurs tactiles capacitifs. En utilisant ces broches, tout changement de charge électrique peut être détecté. Ces broches agissent comme un pavé tactile tel que l'entrée de détection d'un doigt humain ou toute autre interruption tactile causée.

En utilisant ces broches, nous pouvons également concevoir une source de réveil externe pour ESP32 à partir du mode veille profonde.

Les broches tactiles incluent :

- Touch_0 (GPIO4)
- Touch_1 (GPIO0)
- Touch_2 (GPIO2)
- Touch_3 (GPIO15)
- Touch_4 (GPIO13)
- Touch_5 (GPIO12)
- Touch_6 (GPIO14)
- Touch_7 (GPIO27)
- Touch_8 (GPIO33)
- Touch_9 (GPIO32)

Voici les broches du capteur tactile dans la carte ESP32 :



LINUXHINT.COM

Appuyez sur_1 La broche est manquante dans cette version de la carte ESP32 (30 broches). **Appuyez sur_1** la broche est à (GPIO0) qui est présent dans l'ESP32 à 36 broches.

Voici un tutoriel sur Capteur tactile capacitif ESP32 avec Arduino IDE.

11: Goupilles de cerclage ESP32

ESP32 a des broches de cerclage qui peuvent mettre ESP32 dans différents modes comme le chargeur de démarrage ou le mode clignotant. Dans la plupart des cartes équipées de l'USB-Serial intégré, nous n'avons pas à nous soucier de ces broches car la carte elle-même met ESP32 en mode correct, soit en mode clignotant, soit en mode de démarrage.

Cependant, si ces broches sont en cours d'utilisation, vous pouvez rencontrer des problèmes pour télécharger un nouveau code, faire clignoter le micrologiciel ou réinitialiser la carte ESP32.

Vous trouverez ci-dessous les goupilles de cerclage ESP32 disponibles :

- GPIO 0 (doit être BAS pour entrer en mode de démarrage)
- GPIO 2 (doit être flottant ou BAS pendant le démarrage)
- GPIO 4
- GPIO 5 (doit être ÉLEVÉ pendant le démarrage)
- GPIO 12 (doit être BAS pendant le démarrage)
- GPIO 15 (doit être ÉLEVÉ pendant le démarrage)

12: Épingles hautes au BOOT

Certaines broches GPIO présentent un comportement inattendu lorsque les sorties sont connectées à ces broches car ces broches affichent un état HIGH ou génèrent un signal PWM une fois la carte ESP32 démarrée ou réinitialisée.

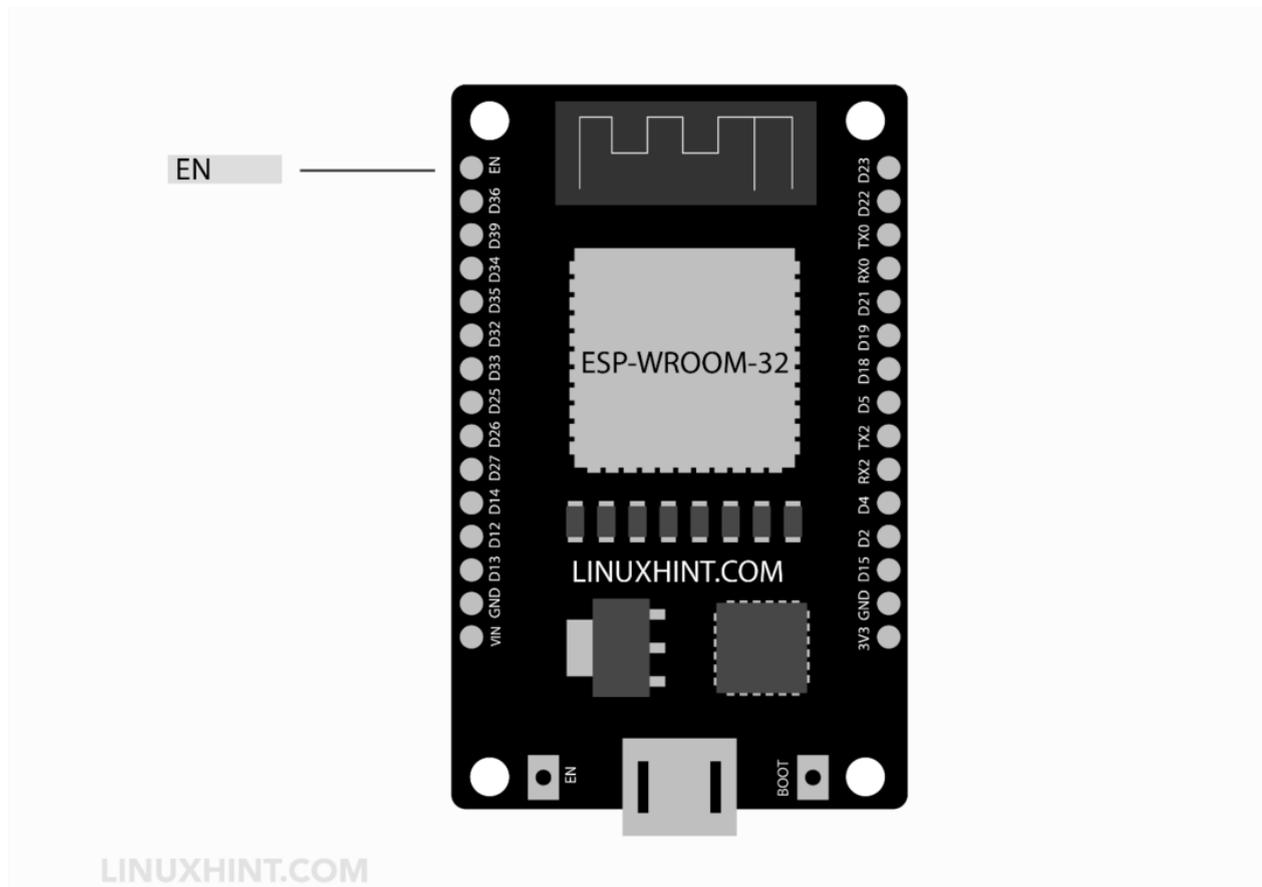
Ces broches sont :

- GPIO 1
- GPIO 3
- GPIO 5
- GPIO 6 à GPIO 11 (interfacé avec le flash SPI interne ESP32 - N'utilisez pas ces broches à d'autres fins).

- GPIO 14
- GPIO 15

13: Activer le code PIN (EN)

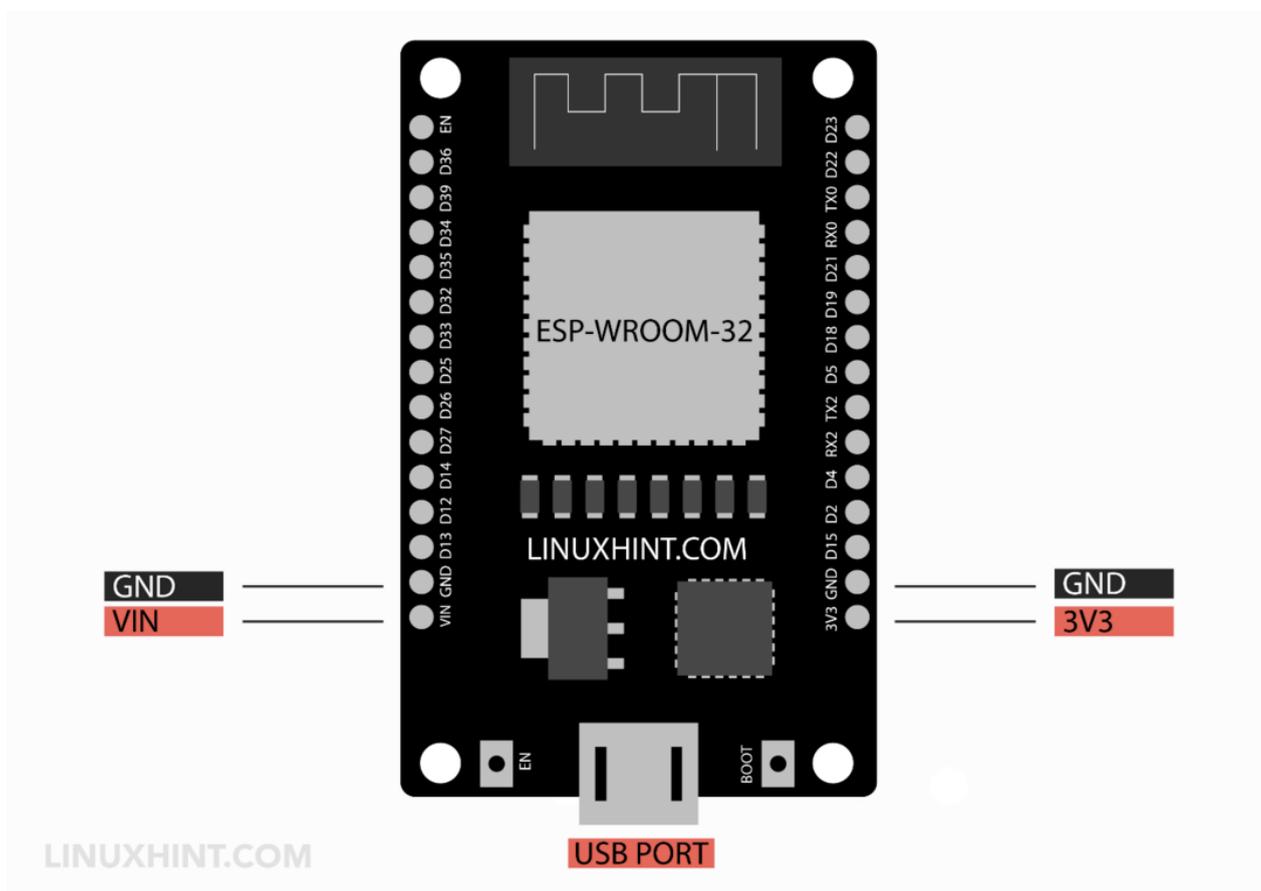
Cette broche est utilisée pour activer la carte ESP32. Grâce à cela, nous pouvons contrôler le régulateur de tension ESP32. Cette broche active la puce lorsqu'elle est tirée vers le HAUT et lorsqu'elle est tirée vers le BAS, l'ESP32 fonctionne à la puissance minimale.



En connectant la broche EN (activation) à GND, le régulateur de tension embarqué de 3,3 V désactive cela, ce qui signifie que nous pouvons utiliser un bouton-poussoir externe pour redémarrer l'ESP32 si nécessaire.

14: broches d'alimentation ESP32

L'ESP32 dispose de plusieurs sources d'alimentation. Principalement deux broches peuvent être utilisées pour alimenter ESP32 qui incluent la broche VIN (Vin) et la broche 3V3 (3.3V). La principale source d'alimentation de l'ESP32 utilise le câble USB. Les deux autres sources nécessitent un approvisionnement externe régulé.



L'ESP32 a un bord **Régulateur de tension** de sortie 3.3V qui prend l'entrée de deux sources USB et la broche VN après cela, il convertit la tension d'entrée (5V) en 3.3V pour le fonctionnement de l'ESP32.

Voici les trois sources d'alimentation pour ESP32 :

- Port USB: ne peut donner de l'alimentation qu'à l'ESP32
- VN PIN: Fonctionne en entrée et en sortie bidirectionnelles
- PIN 3V3: fonctionne en entrée et en sortie bidirectionnelles

Note: La broche 3V3 de l'ESP32 n'est pas connectée au régulateur de tension embarqué, il n'est pas recommandé de l'utiliser pour l'alimentation entrée car une légère augmentation de la tension entraînera un flux de courant plus important à partir de la borne de sortie du LDO régulateur (**AMS1117**) à l'entrée entraînant des dommages permanents du régulateur de tension ESP32.

Cependant, si vous avez une alimentation constante de 3,3 V, vous pouvez l'utiliser.

Deuxièmement, ne donnez pas plus de 9 V à la broche VN car ESP32 n'a besoin que de 3,3 V pour fonctionner; toutes les tensions restantes seront dissipées sous forme de chaleur.

Pour un guide plus détaillé sur les sources d'alimentation ESP32 et les exigences de tension, consultez ce tutoriel comment alimenter ESP32.

15: Capteur à effet Hall ESP32

ESP32 dispose d'un capteur à effet Hall intégré à l'aide duquel nous pouvons détecter les changements de champ magnétique et exécuter une sortie spécifique en conséquence.

Voici un tutoriel sur comment utiliser le capteur à effet Hall intégré ESP32 et imprimer les données lues sur le moniteur série.

Conclusion

Commencer avec ESP32 n'a jamais été facile, mais en utilisant cet article sur le brochage ESP32, tout le monde peut commencer avec une carte basée sur l'IoT en quelques minutes. Ici, cet article couvre tous les détails concernant le brochage ESP32. Chaque broche ESP32 est discutée en détail. Pour plus de tutoriels sur des broches spécifiques,

Catégories

- Adobe (/fr/parts/100-adobe)
- Disney+ (/fr/parts/101-disney)
- /fr/parts/102 (/fr/parts/102)
- Diapositives Google (/fr/parts/103-google-slides)
- Max (/fr/parts/104-max)
- Youtube (/fr/parts/105-youtube)
- Google Docs (/fr/parts/106-google-docs)
- Hulu (/fr/parts/107-hulu)
- Feuilles Google (/fr/parts/108-google-sheets)
- Google Drive (/fr/parts/109-google-drive)
- Commandes A Z (/fr/parts/10-a-z-commands)
- Pinterest (/fr/parts/110-pinterest)
- Calendrier Google (/fr/parts/111-google-calendar)
- Chromebook (/fr/parts/112-chromebook)
- Amazon Prime (/fr/parts/113-amazon-prime)

- Paramount+ (/fr/parts/114-paramount)
- Paon (/fr/parts/115-peacock)
- Travail à Distance (/fr/parts/116-remote-work)
- /fr/parts/117 (/fr/parts/117)
- Android (/fr/parts/11-android)
- écosystème Apple (/fr/parts/12-apple-ecosystem)
- Apple Tv (/fr/parts/13-apple-tv)
- Système D'exploitation Chrome (/fr/parts/14-chrome-os)
- Cloud Computing (/fr/parts/15-cloud-computing)
- Conseils Informatiques (/fr/parts/16-computer-tips)
- Sites Web Sympas (/fr/parts/17-cool-websites)
- Crypto Monnaie (/fr/parts/18-cryptocurrency)
- Science Des Données (/fr/parts/19-data-science)
- Téléchargements De Logiciels Gratuits (/fr/parts/20-free-software-downloads)
- Truc Amusant (/fr/parts/21-fun-stuff)

Dernier article de blog

JUL 30 Activer Bash Windows 10 – Indice Linux
(/fr/chapters/2597-enable-bash-windows-10--linux-hint)

JUL 30 Comment installer WINE pour la compatibilité Windows sur OpenSUSE – Linux Hint
(/fr/chapters/2593-how-to-install-wine-for-windows-compatibility-on-opensuse)

JUL 30 Comment utiliser Synology QuickConnect – Astuce Linux
(/fr/chapters/2610-how-to-use-synology-quickconnect--linux-hint)

Catégories

- > [Adobe \(/fr/parts/100-adobe\)](/fr/parts/100-adobe)
- > [Disney+ \(/fr/parts/101-disney\)](/fr/parts/101-disney)
- > [/fr/parts/102 \(/fr/parts/102\)](/fr/parts/102 (/fr/parts/102)
- > [Diapositives Google \(/fr/parts/103-google-slides\)](/fr/parts/103-google-slides)
- > [Max \(/fr/parts/104-max\)](/fr/parts/104-max)
- > [Youtube \(/fr/parts/105-youtube\)](/fr/parts/105-youtube)
- > [Google Docs \(/fr/parts/106-google-docs\)](/fr/parts/106-google-docs)
- > [Hulu \(/fr/parts/107-hulu\)](/fr/parts/107-hulu)
- > [Feuilles Google \(/fr/parts/108-google-sheets\)](/fr/parts/108-google-sheets)
- > [Google Drive \(/fr/parts/109-google-drive\)](/fr/parts/109-google-drive)

Dernier

- > [Correction rapide de l'erreur "fatal: pas de branche en amont" de Git \(/fr/chapters/16660-fix-git-s--fatal-no-upstream-branch--error-quickly-examp\)](/fr/chapters/16660-fix-git-s--fatal-no-upstream-branch--error-quickly-examp)
- > [Git Stash - Comment stocker les modifications dans Git \(/fr/chapters/16661-git-stash--how-to-stash-changes-in-git\)](/fr/chapters/16661-git-stash--how-to-stash-changes-in-git)
- > [Annuler la validation du dernier exemple de validation de Git \(/fr/chapters/16663-uncommit-git-s-last-commit-example\)](/fr/chapters/16663-uncommit-git-s-last-commit-example)

[Privacy \(/privacy\)](/privacy)

© 2025 Best Tech Tips (<https://ciksiti.com/>) All Rights Reserved.

