Utiliser les pins de la carte ESP32

1. Objectifs

- Connaitre les pins de la carte **ESP32**

2. Description des pins de la carte ESP32

- Découvrez les pins de l'ESP32 afin d'utiliser correctement ces pins GPIO . Même s'il existe de nombreuses variations de cartes ESP32, le fonctionnement des pins reste le même. Par contre, il se peut que certaines broches ne soient pas accessibles en fonction de votre modèle.
- L'ESP32 dispose de nombreuses broches ou pins qui peuvent être utilisées comme entrée ou sortie numérique. L'ESP32 dispose par ailleurs de résistances pull-up et pull-down qui permet de simplifier les circuits.
- Pin ou GPIO(Les ports `GPIO` (anglais : **G**eneral **P**urpose **I**nput/**O**utput, littéralement Entrée-sortie à usage général) sont des ports d'entrées-sorties très utilisés dans le monde des microcontrôleurs)

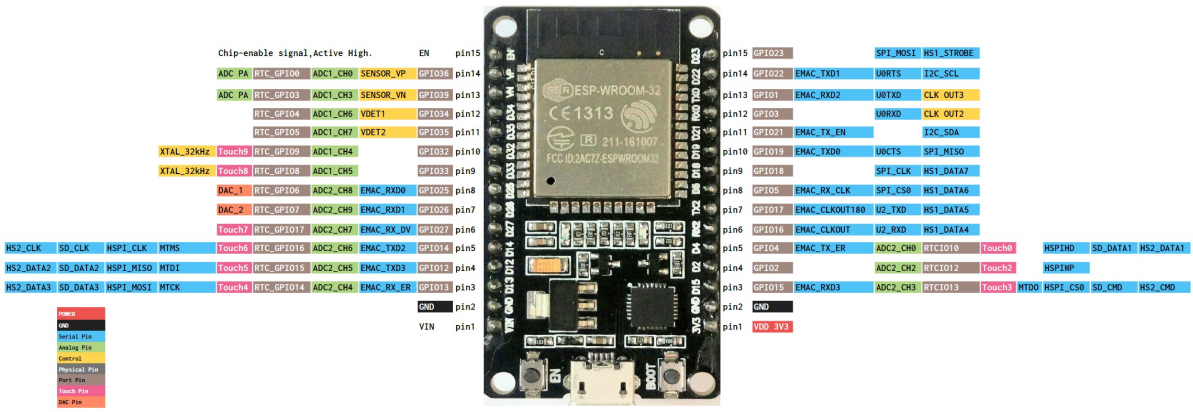
> *Le but est de vous informer des limitations de certains pins GPIO de l'ESP32 pour éviter d'avoir de mauvaises surprises lors de vos projets ou d'avoir un bug incompréhensible dans votre programme.*

- *Le nombre de broche du connecteur d'extension **GPIO** varie d'un fabricant à l'autre. Tout d'abord, certaines broches sont réservées pour l'accès à la mémoire flash via le bus SPI. Ensuite, chaque fabricant choisi les broches qu'il laisse à disposition (expose) de l'utilisateur (développeur) en fonction de sa propre conception et des équipements intégrés à la carte de développement (écran OLED, capteur, caméra, lecteur de carte micro SD...).*

3. Identifier les broches d'entrée sortie numérique de l'ESP32

- L'ESP32 dispose de nombreuses broches pouvant être utilisées comme entrée et / ou sortie numérique. Certaines broches sont réservées ou sont utilisées par le processeur par exemple pour accéder à la mémoire flash via le bus SPI (broches 6 à 11), port série pour téléverser le programme ou le débogage.

apcpedagogie.com



4. Fonctionnement particulier de certaines broches de l'ESP32

- Le but est de vous informer des limitations de certains pins GPIO de l'ESP32 pour éviter d'avoir de mauvaises surprises lors de vos projets ou d'avoir un bug incompréhensible dans votre programme.
- Les cartes de développement basées sur un ESP32 possèdent en général **33 broches** hormis celles pour l’alimentation. Certains pins GPIO ont des fonctionnements un peu particuliers :
- Si votre carte **ESP32** possède les pins **GPIO6**, **GPIO7**, **GPIO8**, **GPIO9**, **GPIO10**, **GPIO11** , il ne faut surtout pas les utiliser car ils sont reliés à la mémoire flash de l'ESP32 : **si vous les utilisez l'ESP32 ne fonctionnera pas.**

- **Analog to Digital Converter (ADC)**

  - ou encore **Convertisseur analogique-numérique (ADC)**
  - L'ESP32 possède **2 ADC** séparés : l’**ADC1** avec 8 canaux et l’**ADC2** avec 10 canaux, pour un total de 18 entrées analogique.
  - Par exemple, il y a sur le pin GPIO34, le canal numéro 6 de l'ADC1 (ADC1_CH6) . Seulement en pratique, il y en a beaucoup moins de disponibles à cause du Wi-Fi.

apcpedagogie.com

Correspondance pour l'ADC1

ADC1_CH0	ADC1_CH1	ADC1_CH2	ADC1_CH3
GPIO 36	GPIO 37	GPIO 38	GPIO 39
ADC1_CH4	**ADC1_CH5**	**ADC1_CH6**	**ADC1_CH7**
GPIO 32	GPIO 33	GPIO 34	GPIO 35

Correspondance pour l'ADC2 (rarement utilisé)

ADC2_CH0	ADC2_CH1	ADC2_CH2	ADC2_CH3	ADC2_CH4
GPIO 4	GPIO 0	GPIO 2	GPIO 15	GPIO 13
ADC2_CH5	**ADC2_CH6**	**ADC2_CH7**	**ADC2_CH8**	**ADC2_CH9**
GPIO 12	GPIO 14	GPIO 27	GPIO 25	GPIO 26

- **Digital to Analog Converter (DAC)**
  - L'ESP32 possède 2 sorties DAC de 8 bits pour convertir un signal numérique en un signal analogique (générer un sinus par exemple).
  - La structure de conception est composée de chaînes de résistances intégrées et d'un tampon. Ce double DAC prend en charge l'alimentation comme référence de tension d'entrée.
  - Les deux canaux DAC peuvent également prendre en charge des conversions indépendantes.
  - Les pins utilisés sont les suivants :`DAC1` **GPIO 25** et `DAC2` **GPIO 26**

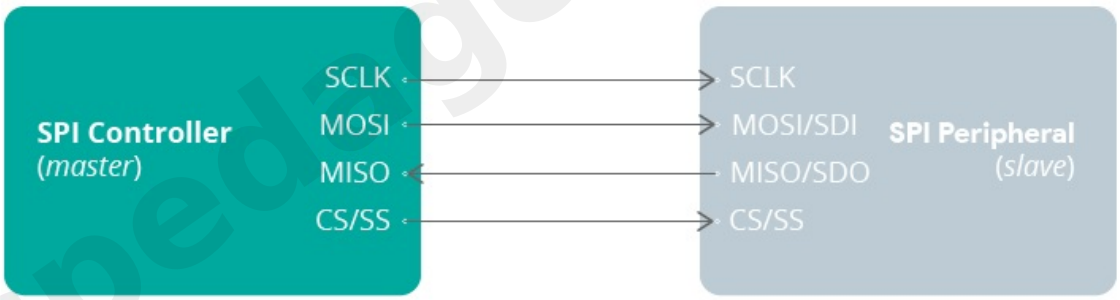
- **Senseur tactile (Touch Sensor) de l'ESP32**
  - Ou encore les capteurs capacitifs sur l'ESP32.
  - L'ESP32 dispose de **10 GPIO** à détection capacitive, qui détectent les variations induites par le toucher ou l'approche des GPIO avec un doigt ou d'autres objets.
  - La nature à faible bruit de la conception et la haute sensibilité du circuit permettent tampons relativement petits à utiliser.
  - Des rangées de pastilles peuvent également être utilisées, de sorte qu'une plus grande zone ou plusieurs points peuvent être détecté.
  - Les capteurs capacitifs internes sont connectés aux pins GPIO suivants :

apcpedagogie.com

TOUCH0	TOUCH1	TOUCH2	TOUCH3	TOUCH4
GPIO 4	GPIO 0	GPIO 2	GPIO 15	GPIO 13
TOUCH5	**TOUCH6**	**TOUCH7**	**TOUCH8**	**TOUCH9**
GPIO 12	GPIO 14	GPIO 27	GPIO 33	GPIO 32

- **Le SPI (Serial Peripheral Interface) sur l'ESP32**

  - SPI signifie **S**erial **P**eripheral **I**nterface, et c'est un protocole de données série synchrone utilisé par les microcontrôleurs pour communiquer avec un ou plusieurs périphériques. Par exemple, votre carte ESP32 communiquant avec un capteur supportant le SPI ou avec un autre microcontrôleur.
  - Dans une communication SPI, il y a toujours un contrôleur (également appelé maître) qui contrôle les périphériques (également appelés esclaves).
  - Les données peuvent être envoyées et reçues simultanément. Cela signifie que le maître peut envoyer des données à un esclave et qu'un esclave peut envoyer des données au maître en même temps.



  - Vous ne pouvez avoir qu'un seul maître , qui sera un microcontrôleur (l'ESP32), mais vous pouvez avoir plusieurs esclaves. Un esclave peut être un capteur, un écran, une carte microSD, etc., ou un autre microcontrôleur. Cela signifie que vous pouvez avoir un ESP32 connecté à plusieurs capteurs, mais le même capteur ne peut pas être connecté à plusieurs cartes ESP32 simultanément.
  - L'ESP32 intègre 4 périphériques **SPI : SPI0, SPI1, SPI2** (communément appelé HSPI) et **SPI3** (communément appelé **VSPI**).
  - SP0 et SP1 sont utilisés en interne pour communiquer avec la mémoire flash intégrée, et vous ne devez pas les utiliser pour d'autres tâches.
  - Vous pouvez utiliser **HSPI** et **VSPI** pour communiquer avec d'autres appareils. **HSPI** et **VSPI** ont des signaux de bus indépendants et chaque bus peut piloter

apcpedagogie.com

jusqu'à trois esclaves **SPI**.

- Par défaut, la correspondance des pins est la suivante :

SPI	MOSI	MISO	SCK/CLK	CS/SS
VSPI	GPIO 23	GPIO 19	GPIO 18	GPIO 5
HSPI	GPIO 13	GPIO 12	GPIO 14	GPIO 15

Source:

- https://randomnerdtutorials.com/esp32-spi-communication-arduino/
- https://www.upesy.fr/blogs/tutorials/esp32-pinout-reference-gpio-pins-ultimate-guide#le-spi-sur-lesp32

Cours Carte de développement ESP32

Source : https://apcpedagogie.com/utiliser-les-pins-de-la-carte-esp32 · téléchargé le 08/11/2026 · © APC Pédagogie · apcpedagogie.com

apcpedagogie.com