



# La carte ESP32 Feather Huzzah



[Mise à jour le 24/4/2022]

- **Sources**

- [2022 - NEW GUIDE: Adafruit ESP32-S3 Feather](#)
- Sites [Espressif](#) et [Adafruit](#)
- [espressif/arduino-esp32](#) sur Github
- [ESP-IDF Programming Guide](#)

- **Bibliothèques**

- [Arduino Library List](#)

- **Lectures connexes**

- Wiki "[Matériels - Capteurs, afficheurs, préactionneurs, etc.](#)"
- Wiki "[Mettre en oeuvre un client MQTT sur un EP8266 \(ESP32\) Feather Huzzah ou MKR1010](#)"
- Wiki "[Le système de fichiers LittleFS \(ESP\)](#)" (Mini Serre)

- **Distributeur**

- [Gotronic](#)

## 1. Généralités

L'ESP32 est une mise à niveau de l'ESP8266. En comparaison, l'ESP32 dispose de plus de GPIO et d'entrées analogiques, de deux sorties analogiques, de plusieurs périphériques supplémentaires (comme l'UART), de deux cœurs pour ne pas avoir à céder au gestionnaire WiFi, d'un processeur beaucoup plus rapide, etc !

La carte Feather HUZZAH32, développée par Adafruit, est une carte de développement WiFi et Bluetooth BLE "tout-en-un" avec USB intégré et chargeur de batterie développée à partir du module officiel [WROOM32](#) d'Espressif .

## 2. Principales caractéristiques

- **SOC** : [ESP32](#)
- **Processeur** : 32-bit CPU @ 240MHz
- **Wifi** 802.11 b/g/n (WEP/WPA)

- **Bluetooth** : compatible BLE
- **Flash** : 4MB
- **SRAM** : 520KB
- **PWM/timer** : disponible sur l'ensemble des broches GPIO
- **I<sup>2</sup>C** : 2 (seulement 1 est configuré par défaut dans le support Feather Arduino IDE)
- **SPI** : 3 (seulement 1 est configuré par défaut dans le support Feather Arduino IDE)
- **UART** : 3 (seulement deux sont configurés par défaut dans le support Feather Arduino IDE, un UART est utilisé pour le transfert et le débogage)
- **I2S Audio** : 2
- **Entrée analogique** : 12 (**1,8V max**)
- **Oscillateur** : 32kHz
- **Alimentation** : 5V régulateur 3.3V (pic de courant 500mA max)
- **Chargeur Lipo intégré** : ( 3,7V - 4,2V) - 100mA
- **LED** rouge à usage général sur la broche 0
- **Bouton** de Reset
- **Consommation** : 60 à 200mA
- **Dimensions** : 51mm x 23mm x 8mm
- **Masse** : 7g

### 3. Brochage

- **Identification** et description des entrées / sorties sur le site [Adafruit](https://adafruit.com/products/4077).



### ADAFRUIT HUZZAH32 PIN DIAGRAM

**A13** not exposed. It's used for measuring the voltage on the battery. The voltage is divided by 2 so multiply the analogRead by 2.

**GPIO#12** Used for booting up. Adafruit suggests not using it or only using for output.

**ADC#2** does not work when WiFi is activated. The ESP32 internally uses ADC#2 for WiFi

**PWM** is possible on every GPIO pin

Positive voltage from USB jack, if connected. ~5V

Drive LOW to disable 3.3V regulator

Positive voltage from LiPoly battery, if connected. ~3.7V

|  |           |          |
|--|-----------|----------|
| I2C SDA  | GPIO #23  | GPIO #21 |
| I2C SCL  | GPIO #22  | GPIO #17 |
| TOUCH6   | A6 ADC#2  | GPIO #14 |
| TOUCH9   | A7 ADC#1  | GPIO #32 |
| TOUCH3   | A8 ADC#2  | GPIO #15 |
| TOUCH8   | A9 ADC#1  | GPIO #33 |
| TOUCH7   | A10 ADC#2 | GPIO #27 |
| TOUCH5   | A11 ADC#2 | GPIO #12 |
| TOUCH4   | A12 ADC#2 | GPIO #13 |
| Positive voltage from USB jack, if connected. ~5V            |           |          |
| Drive LOW to disable 3.3V regulator                          |           |          |
| Positive voltage from LiPoly battery, if connected. ~3.7V    |           |          |
| GPIO #21   |           |          |
| GPIO #17   |           |          |
| GPIO #16   |           |          |
| GPIO #19   |           |          |
| GPIO #18   |           |          |
| GPIO #5  |           |          |
| GPIO #4  |           |          |
| GPIO #36   |           |          |
| GPIO #39   |           |          |
| GPIO #34   |           |          |
| GPIO #25   |           |          |
| GPIO #26   |           |          |
| GND. Common ground for power and logic                       |           |          |
| No connect pin (not used)                                    |           |          |
| 3.3V output from on-board regulator. Can supply up to 500mA. |           |          |
| Drive LOW to reset (or press Reset button)                   |           |          |

By @jonfroehlich

| Rôle                | Nom sur la carte    | Broche | Shield Particle | Remarques                          |
|---------------------|---------------------|--------|-----------------|------------------------------------|
| GPIO et Analogique  | A0 (ADC2) ou (DAC2) | 26     | A0              |                                    |
|                     | A1 (ADC2) ou (DAC1) | 25     | A1              |                                    |
|                     | A2 (ADC1)           | 34     | A2              | Analog utilisable sous MicroPython |
|                     | A3 (ADC1)           | 39     | A3              | Analog utilisable sous MicroPython |
|                     | A4 (ADC1)           | 36     | A4              | Analog utilisable sous MicroPython |
|                     | A5 (ADC2)           | 4      | A5              |                                    |
|                     | A6 (ADC2)           | 14     | D2              |                                    |
|                     | A7 (ADC1)           | 32     | D3              | Analog utilisable sous MicroPython |
|                     | A8 (ADC2)           | 15     | D4              |                                    |
|                     | A9 (ADC1)           | 33     | D5              | Analog utilisable sous MicroPython |
|                     | A10 (ADC2)          | 27     |                 |                                    |
|                     | A11 (ADC2)          | 12     |                 |                                    |
|                     | A12 (ADC2)          | 13     |                 |                                    |
| GPIO                | 21                  | 21     |                 | E/S à usage général                |
| Série (asynchrone)  | RX                  | 16     | RX              |                                    |
|                     | TX                  | 17     | TX              |                                    |
| Bus I2C (Synchrone) | SCL                 | 22     | SCL             |                                    |
|                     | SDA                 | 23     | SDA             |                                    |
| Bus SPI (Synchrone) | SCK                 | 5      |                 |                                    |
|                     | MOSI                | 18     |                 |                                    |
|                     | MISO                | 19     |                 |                                    |

4. Accessoires



- **Adafruit Grove Shield FeatherWing** ([schéma](#)) pour Particle Mesh et tous les Feathers<sup>1)</sup>.
  - **Caractéristiques**
    - 8 connecteurs Grove :
      - 3 connecteurs analogiques, 2 broches analogiques par connecteur, 6 broches analogiques au total;
      - 2 connecteurs numériques, 2 broches numériques par connecteur, 4 broches numériques au total;
      - 2 connecteurs I2C
      - 1 connecteur UART
  - **Schéma** : téléchargeable [ici](#)
  - **Distributeurs** : [Mouser](#) et ([Adafruit](#))
  - **Table de correspondance**

| Shield | ESP | Commentaires   |
|--------|-----|--|
| D2     | 14  | GPIO 14 et entrée analogique A6 sur le CAN 2   |
| D3     | 32  | GPIO 32 et entrée analogique A7 sur l'ADC 1. Peut également être utilisée pour connecter un quartz 32 KHz. |
| D4     | 15  | GPIO 15 et entrée analogique A8 sur l'ADC 2  |
| D5     | 33  | GPIO 33 et entrée analogique A9 sur l'ADC 1. Peut également être utilisée pour connecter un quartz 32 KHz. |



- **Batterie Lipo 3,7V 500mA**
  - **Caractéristiques**
    - Tension: 3,7 Vcc
    - Intensité: 400 mAh
    - Courant de décharge maxi: 880 mA
    - Connecteur: 2 broches type JST

- Dimensions: 35,5 x 25,5 x 5,2 mm
- Longueur du câble: 100 mm
- Poids: 10 g
- **Distributeurs** : [Gotronic](#)

## 5. Préparation de l'IDE Arduino

### 5.1 Installation du support pour les cartes à "ESP32"

- L'installation du support ESP32 pour Arduino se fait en passant par l'édition des préférences (**Fichier** → **Préférences**).

Dans « URL de gestionnaire de cartes supplémentaires » ajoutez :

```
https://dl.espressif.com/dl/package_esp32_index.json
```

URL de gestionnaire de cartes supplémentaires

Remarque : une URL par ligne, pas de séparateur !

### 5.2 Installation du gestionnaire de carte

- Dans le menu « **Outils** », « **Type de carte** » et « **Gestionnaire de carte** », rechercher « **esp32 by Espressif Systems** » et installer le composant.

esp32 by Espressif Systems  
Cartes incluses dans ce paquet:  
ESP32 Dev Module, WEMOS LoLin32.  
[More info](#)

1.0.1

Installer

### 5.3 Spécificités de la carte Adafruit HUZZAH32 - ESP32 Feather

- Installer la carte à partir de [Check out the Espressif Arduino repository for details on how to install it](#)
- Installer le driver [SiLabs CP2104](#)

## 6. Simulateur

- Voir le simulateur ESP32 sur <https://wokwi.com/>

## Pour aller plus loin

- [ESP-IDF Programming Guide](#)
- [Intégration continue sans serveur et flux de mise à jour OTA pour les appareils IoT utilisant](#)

## Google Cloud Build et Arduino

1)

Sur l'ESP8266, il est nécessaire de connecter une résistance de 10k entre la broche CHPD et le 3,3V

From:

<https://webge.fr/dokuwiki/> - **WEBGE Wikis**

Permanent link:

<https://webge.fr/dokuwiki/doku.php?id=arduino:uc:esp32&rev=1659112659>

Last update: **2022/07/29 18:37**

