

Alexandra Dulong

Communication à distance

243-510-MA

Projet : 'Streaming' vidéo avec le ESP32cam sur un Serveur IP

Travail présenté à

La communauté GitHub

Collège Maisonneuve

Département des Technologies du Génie Électrique

Décembre 2020

GUIDE - 'Streaming' vidéo avec le ESP32cam sur un Serveur IP

Mise en contexte

Le projet consiste à créer dans un premier temps un système de surveillance caméra simple et à petit budget. L'image captée par la caméra est transmise sur le réseau Internet et peut être observée via un appareil connecté au réseau. Mon deuxième objectif pour le projet est d'intégrer la fonction de reconnaissance faciale qu'offre un logiciel compatible avec le ESP32cam.

Le module de caméra utilisé pour le projet est le ESP32cam. Il est muni d'une caméra 2M pixel. Ce module, lorsque programmé, est capable de se connecter au Wifi et de transmettre ces images captées via un serveur Web (IP). Pour accéder à ces images captées en direct, il suffit de se connecter au navigateur web avec n'importe quels appareils connectés au réseau et de rechercher l'adresse IP du système.

Mon objectif principal est de pouvoir visualiser une image en direct, captée par le ESP32cam, sur mon ordinateur ou sur mon cellulaire. Mon objectif secondaire est de me familiariser avec la fonctionnalité de reconnaissance faciale. Lorsque mon visage, préalablement mémorisé, est capté par la caméra, une lumière verte s'allume. Par contre, si le visage capté n'était pas le mien ou que la caméra ne perçoit aucun visage, la lumière rouge est allumée.

Liste de matériel

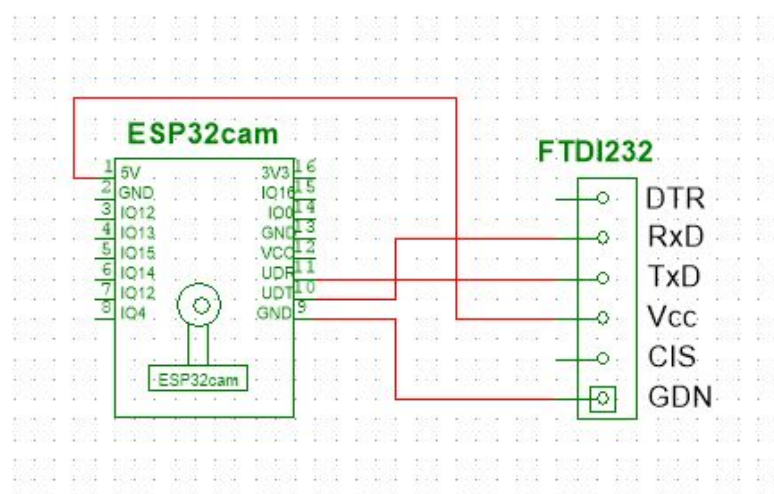
- ESP32 Caméra (10\$)
- Convertisseur FT232RL FTDI USB vers TTL (5\$)
- Convertisseur 5V
- Câble microUSB
- Ordinateur
- Logiciel Arduino IDE

Guide de réalisation

Système de surveillance caméra

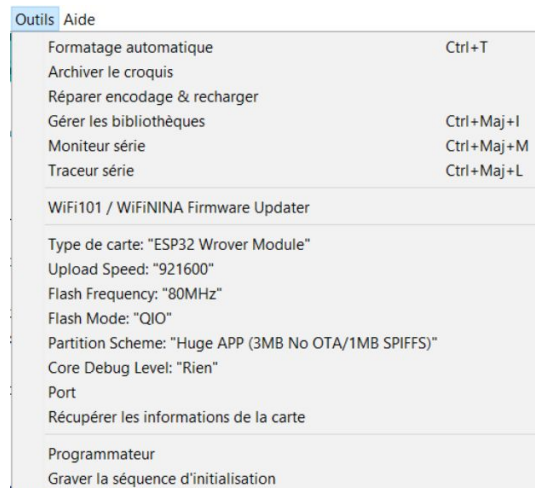
Étape 1 :

Faire le montage pour programmer le ESP32cam avec le Convertisseur FTDI 232 (USB vers l'interface série du module).



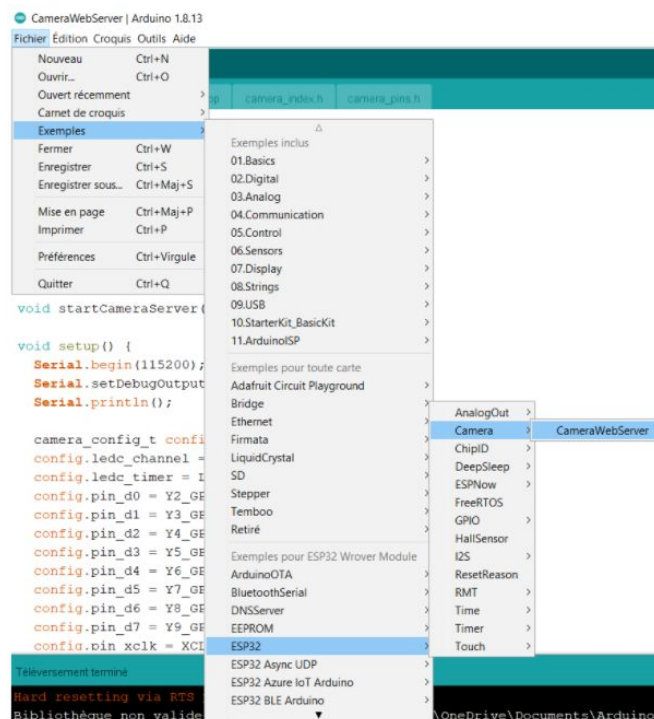
Étape 2 :

Ouvrez le logiciel Arduino IDE et assurez-vous d'avoir les configurations et la bonne carte de développement sélectionnée. Les paramètres à configurer se retrouvent dans l'onglet Outils.



Étape 3 :

Sélectionnez le code d'exemple offert par le logiciel Arduino et faites une analyse des fonctions du code.

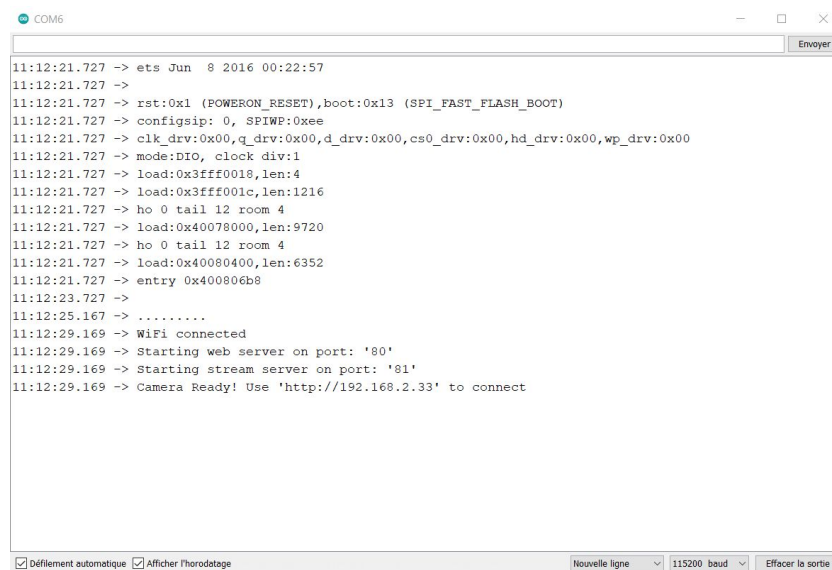


Étape 4 :

Sélectionnez le bon modèle : " #define CAMERA_MODEL_AI_THINKER ". Entrez vos informations de connexion au réseau WiFi (ssid ; password). Assurez-vous de connecter la broche GPIO 0 au GND et appuyez sur le bouton RESET. Programmez le ESP32cam avec le code modifié. Le code pour la caméra modèle AI_THINKER est déposé dans un fichier texte. C'est le code nommé : " surveillance caméra".

Étape 5 :

Ouvrez le moniteur série du logiciel Arduino IDE. Déconnectez la broche GPIO 0 du GND et appuyez sur RESET. Le module ESP32cam devrait se connecter au wifi et afficher à l'écran l'adresse IP nécessaire pour se connecter au serveur.

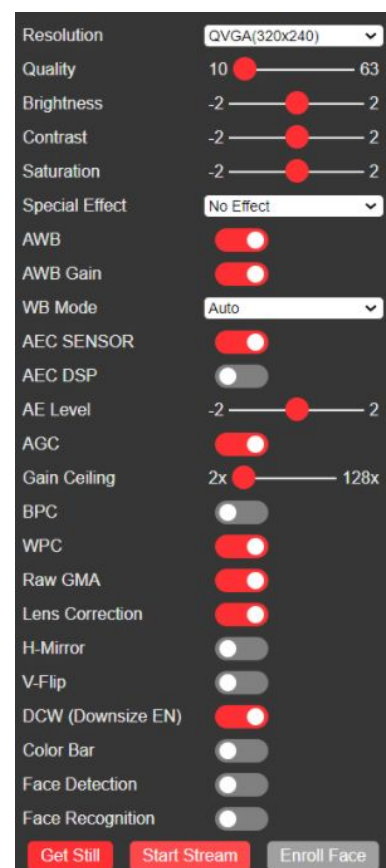


```
11:12:21.727 -> ets Jun 8 2016 00:22:57
11:12:21.727 ->
11:12:21.727 -> rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
11:12:21.727 -> configsip: 0, SPIWP:0xee
11:12:21.727 -> clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
11:12:21.727 -> mode:DIO, clock div:1
11:12:21.727 -> load:0x3fff0018,len:4
11:12:21.727 -> load:0x3fff001c,len:1216
11:12:21.727 -> ho 0 tail 12 room 4
11:12:21.727 -> load:0x40078000,len:9720
11:12:21.727 -> ho 0 tail 12 room 4
11:12:21.727 -> load:0x40080400,len:6352
11:12:21.727 -> entry 0x400806b8
11:12:23.727 ->
11:12:25.167 -> .....
11:12:29.169 -> WiFi connected
11:12:29.169 -> Starting web server on port: '80'
11:12:29.169 -> Starting stream server on port: '81'
11:12:29.169 -> Camera Ready! Use 'http://192.168.2.33' to connect
```

Étape 6 :

Testez le code, préalablement chargé sur le ESP32cam, avec un appareil connecté au réseau pour l'accès aux images en direct. Rendez-vous sur votre navigateur de réseau préféré et entrez l'adresse IP pour accéder au serveur. Voici l'interface de l'application :

Dans le menu de gauche, appuyez sur "start stream" afin de pouvoir observer les images captées. L'interface devrait ressembler à l'image ci-dessous.



Étape 7:

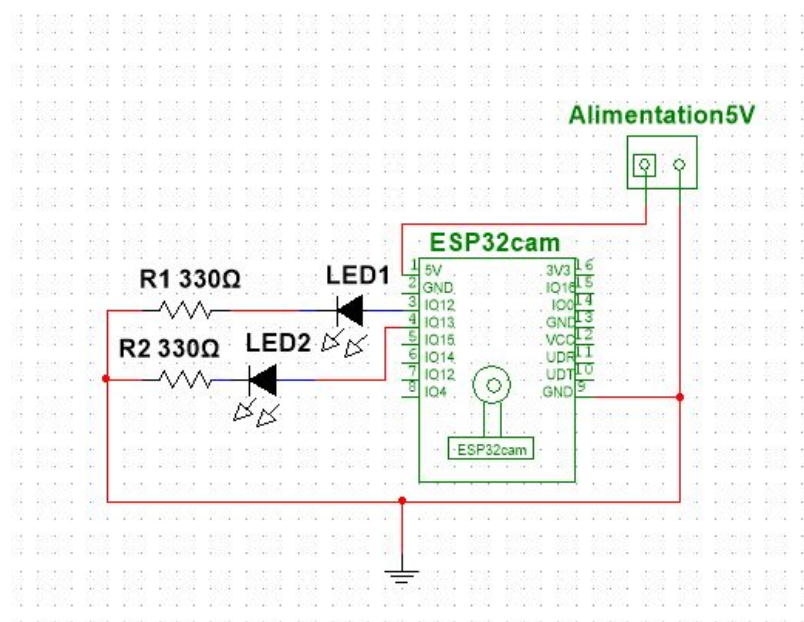
Concevoir, avec un câble USB et un bloc d'alimentation, une alimentation régulée de 5V afin de pouvoir utiliser la caméra à un autre endroit que connecter à l'ordinateur. Déconnectez le ESP32cam du module FTDI232 et connectez la broche Vcc et GND au câble conçu pour réguler le 5V nécessaire. Testez à nouveau le fonctionnement. Vous pouvez maintenant utiliser votre caméra ailleurs que sur votre ordinateur.



Système de surveillance caméra avec la fonctionnalité de reconnaissance faciale

Étape 1 :

Reprenez toutes les étapes élaborées précédemment pour concevoir le système de caméra surveillance sur serveur IP. Pour ajouter la fonctionnalité d'allumer une lumière verte si votre visage est reconnu et une lumière rouge s'il n'y a pas de visage ou si le visage capté n'est pas le vôtre, il faut d'abord faire le montage. Voici le schéma :

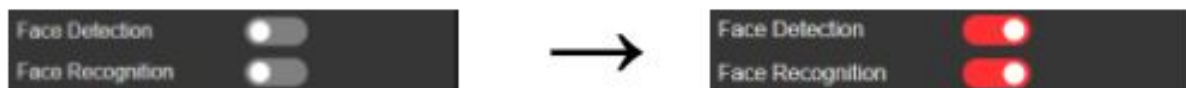


Étape 2 :

Copiez et collez le code "reconnaissance faciale" dans votre projet du logiciel Arduino IDE avec les mêmes configurations que la partie surveillance caméra. La procédure de téléchargement est la même.

Étape 3 :

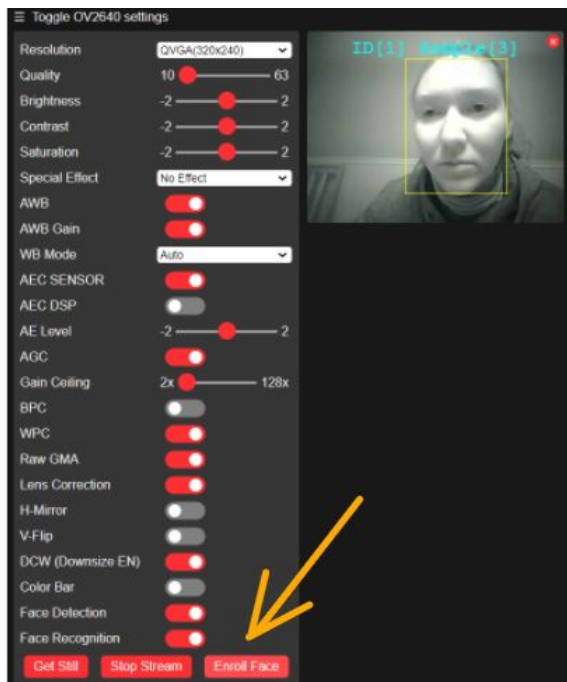
Rendez-vous sur votre navigateur de réseau préféré et entrez l'adresse IP pour accéder au serveur. Activez la fonction de reconnaissance faciale et de détection faciale telle qu'illustrée dans l'image suivante :



Testez cet ajout à l'image captée par la caméra. Assurez-vous qu'un visage humain est filmé. Un carré rouge encadrera celui-ci s'il est détecté et un message d'alerte apparaîtra puisque votre figure n'est pas enregistrée. Voici le résultat :

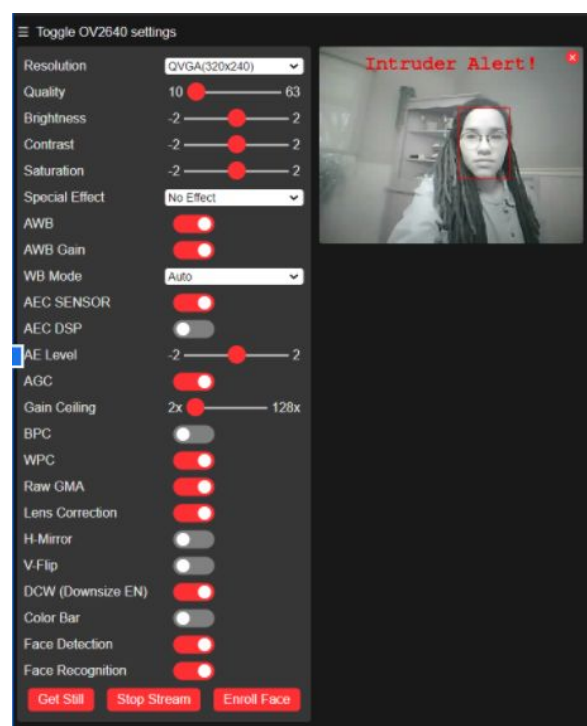
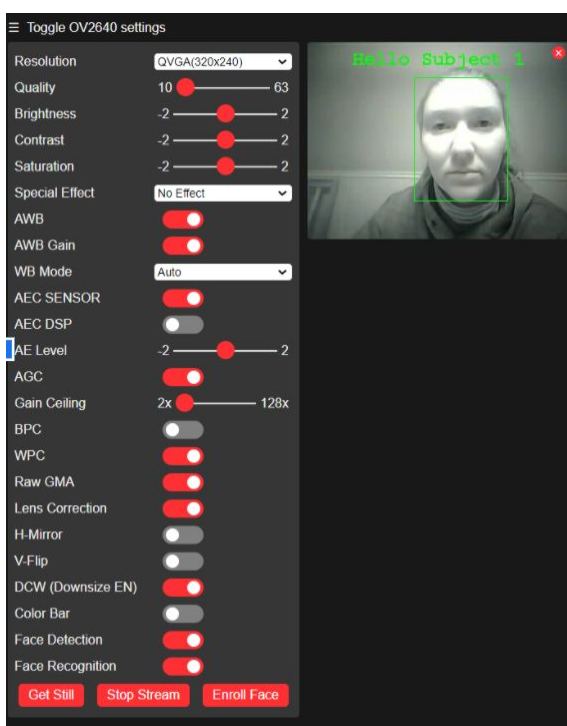


Étape 4 :



Enregistrer votre visage dans le système. Activez la fonction "enroll face" dans le menu à gauche de l'écran. Le système prendra 5 photos de votre visage qui serviront d'échantillons pour déterminer si le visage capté sera le vôtre.

Testez cet ajout de détection de votre visage en particulier. Filmez d'abord votre visage, un carré vert devrait apparaître autour de votre portrait pour signaler que c'est bien celui échantillonné plus tôt. Demandez à une autre personne de venir tester votre projet, le visage capté par la caméra ne devrait pas être reconnu et un carré rouge devrait encadrer le visage. Les lumières verte et rouge devraient s'allumer en conséquence. Voici le résultat à l'écran du périphérique :



Étape BONUS :

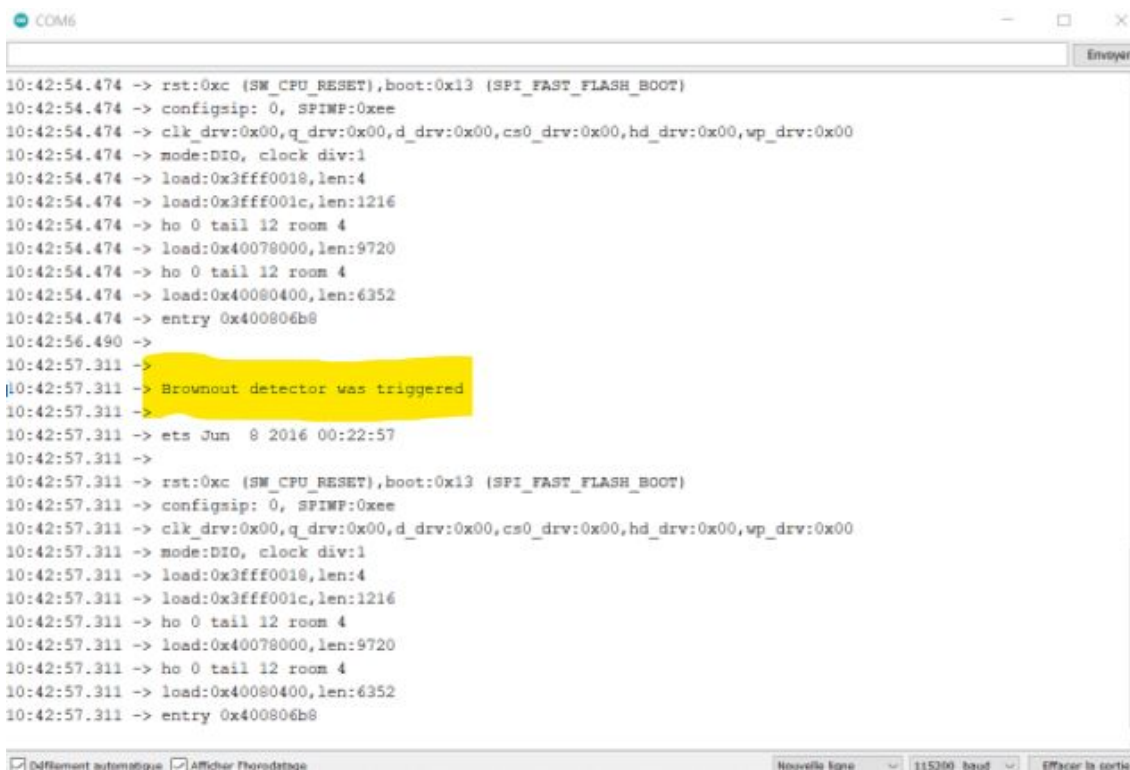
Félicitation vous venez de concevoir un système de reconnaissance faciale. Vous pouvez maintenant concevoir un boîtier pour la caméra à votre goût. Voici mon résultat final :



Erreur rencontré

“Brownout detector was triggered”

Lorsque le message d’erreur “Brownout detector was triggered” s’affiche en boucle sur le moniteur série, c’est qu’il y ait un problème matériel sur votre système.



The screenshot shows a serial monitor window titled 'COM6'. It displays a series of boot logs for an ESP8266. The logs show the device resetting (rst:0xc (SW_CPU_RESET)), booting from SPI flash (boot:0x13 (SPI_FAST_FLASH_BOOT)), and loading various components like the SPI flash, mode (DIO), and various drivers. The logs are repeated twice, indicating a loop. In the middle of the first loop, the message 'Brownout detector was triggered' is displayed, highlighted with a yellow background. The timestamp for this message is 10:42:57.311. The bottom of the window shows settings for 'Nouvelle ligne', '115200 baud', and 'Effacer la sortie'.

```
10:42:54.474 -> rst:0xc (SW_CPU_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
10:42:54.474 -> configspi: 0, SPIWP:0xee
10:42:54.474 -> clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
10:42:54.474 -> mode:DIO, clock div:1
10:42:54.474 -> load:0x3fff0010,len:4
10:42:54.474 -> load:0x3fff001c,len:1216
10:42:54.474 -> ho 0 tail 12 room 4
10:42:54.474 -> load:0x40078000,len:9720
10:42:54.474 -> ho 0 tail 12 room 4
10:42:54.474 -> load:0x40080400,len:6352
10:42:54.474 -> entry 0x400806b8
10:42:56.490 ->
10:42:57.311 ->
10:42:57.311 -> Brownout detector was triggered
10:42:57.311 ->
10:42:57.311 -> ets Jun  8 2016 00:22:57
10:42:57.311 ->
10:42:57.311 -> rst:0xc (SW_CPU_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
10:42:57.311 -> configspi: 0, SPIWP:0xee
10:42:57.311 -> clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
10:42:57.311 -> mode:DIO, clock div:1
10:42:57.311 -> load:0x3fff0010,len:4
10:42:57.311 -> load:0x3fff001c,len:1216
10:42:57.311 -> ho 0 tail 12 room 4
10:42:57.311 -> load:0x40078000,len:9720
10:42:57.311 -> ho 0 tail 12 room 4
10:42:57.311 -> load:0x40080400,len:6352
10:42:57.311 -> entry 0x400806b8
```

Pour remédier à ce problème, vérifiez par exemple l’état de votre câble USB. Le problème pourrait être :

- mauvaise qualité du breadboard
- mauvaise qualité du câble USB
- câble USB trop long
- mauvaise connexion au port USB de l’ordinateur
- Trop peu de puissance fournie par le câble USB