

# Esp32 Cam et Tasmota

<https://cgomesu.com/blog/Esp32cam-tasmota-webcam-server/>

<https://makerhelp.fr/tasmota-comment-activer-le-serveur-rtsp-sur-lesp32-cam/>

[https://templates.blakadder.com/ai-thinker\\_ESP32-CAM.html](https://templates.blakadder.com/ai-thinker_ESP32-CAM.html)



## Ouvrez un terminal et installez les paquets requis:

```
sudo apt update
sudo apt install wget python3 python3-pip
```

## Installer esptool.py via pip3:

```
pip3 install esptool
```

## Découvrez si esptool.py peut être trouvé dans votre utilisateur \$PATH.

```
whereis esptool.py
```

Si esptool.py n'a pas été trouvé, cela signifie que votre utilisateur .local/bin n'est pas dans votre \$PATH. Ajoutez-le comme suit:

```
echo "export PATH=\"$HOME/.local/bin:$PATH" | tee -a \"$HOME/.bashrc" > /dev/null
```

Connectez votre caméra ESP32 à l'adaptateur USB à TTL/série en mode flash:

## ESP32-CAM wiring: Firmware flash mode (add GPIO0-GND jumper)



Attention. Assurez-vous que votre adaptateur USB vers TTL a VCC en mode 5V et dans l'ESP32, le câble VCC est connecté à la broche 5V. Vérifiez le câblage avant de passer à autre chose.

Connectez l'adaptateur à un port USB sur votre ordinateur et vérifiez le nouvel appareil /dev/:

```
ls -l /dev/ttyUSB*
```

Ajouter votre \$USER au même groupe que /dev/ttyUSB\*(c'est généralement dialoutmais si différent, changer dans la commande ci-dessous) et tty:

```
sudo usermod -aG dialout,tty ${USER}
```

Déconnectez-vous et revenez. (Si vous continuez à rencontrer des problèmes d'autorisation, essayez plutôt de redémarrer. Vous pouvez vérifier les autorisations de votre utilisateur avec `id ${USER}`.)

### Flashing Tasmota32 serveur webcam

Nous sommes maintenant prêts à flasher le firmware de Tasmota. Pour référence, les informations officielles sont disponibles à l'adresse <https://tasmota.github.io/docs/ESP32>.

### Créer un tasmota32Dir dans /opt:

```
cd /opt
```

```
sudo mkdir tasmota32
```

### Changer la propriété du nouveau répertoire à l'utilisateur actuel plutôt que root:

```
sudo chown ${USER}:${USER} tasmota32/
```

Télécharger le `tasmota32-webcam.factoryb.bi.factoryn` binaire et les binaires ESP32 Tasmota nécessaires du dépôt officiel de Github via `wget`. (Ce qui suit a été mis à jour le 12 août 2021.) Les binaires sont maintenant disponibles dans un dépôt différent de celui d'avant, à savoir `arendst/Tasmota-firmware` et `arendst/Tasmota-firmware`, et actuellement, le nouveau dépôt a une seule branche (`main`). Il existe deux versions de la `tasmota32-webcam.factory.bin`, un de la `release` et un autre de la `development` parties du projet Tasmota32.

Mon conseil est d'essayer la version stable d'abord, puis le développement si vous avez des problèmes, à moins qu'il n'y ait une note dans `Changelog` qui dit le contraire

Pour télécharger le binaire de version stable, utilisez la commande suivante:

```
wget -P /opt/tasmota32/  
https://ota.tasmota.com/tasmota32/release/tasmota32-webcam.factory.bin
```

Alternativement, pour télécharger le binaire de développement, utilisez la commande suivante:

```
wget -P /opt/tasmota32/  
https://ota.tasmota.com/tasmota32/tasmota32-webcam.factory.bin
```

Assurez-vous que votre caméra ESP32 est connectée à votre ordinateur en mode flash (sauter `GPIO0-GND`). Maintenant, trouvez le port USB que votre appareil utilise `/dev/` et le mettre à la variable environnementale `ESP_PORT`, comme suit:



Attention. Bien que pratique, la commande suivante suppose qu'il existe un seul adaptateur USB à série connecté à votre ordinateur. Si ce n'est pas le cas, régler manuellement `ESP_PORT` à l'emplacement que votre adaptateur USB utilise actuellement. Vous pouvez trouver le port via `ls /dev/ttyUSB*` et tester un par un jusqu'à ce que vous trouviez celui utilisé par l'adaptateur. Alternativement, il suffit de déconnecter tous les autres adaptateurs USB à série pour cette procédure et continuer.

```
ESP_PORT=$(ls /dev/ttyUSB*)
```

Veuillez noter que cela ne fonctionne que si vous continuez à utiliser la même coquille dans laquelle `ESP_PORT` a été définie. Si vous vous déconnectez ou même fermez le terminal actuel, vous devrez redéfinir `ESP_PORT` pour continuer à l'utiliser.

**Vous pouvez vérifier cela `ESP_PORT` a été correctement définie par `echo`, comme suit:**

```
echo $ESP_PORT
```

qui devrait produire quelque chose comme ceci:

```
/dev/ttyUSB0
```

Effacez le firmware actuel (ou toutes les données) de votre ESP32-cam.



Attention. La procédure suivante effacera toutes les données sur la caméra ESP32.

```
esptool.py --port $ESP_PORT erase_flash
```



Attendre jusqu'à esptool.py est fait. Ensuite, appuyez sur le bouton de réinitialisation de la caméra ESP32. Maintenant, vérifiez ça \$ESP\_PORT est de nouveau disponible.

**Flash le tasmota32-webcam.factory.bin serveur de webcam binaire et les binaires Tasmota requis à l'ESP32-cam.**

```
esptool.py --chip esp32 \  
--port $ESP_PORT \  
--before default_reset \  
--after hard_reset \  
write_flash -z \  
--flash_mode dout \  
--flash_size detect \  
0x0 /opt/tasmota32/tasmota32-webcam.factory.bin
```



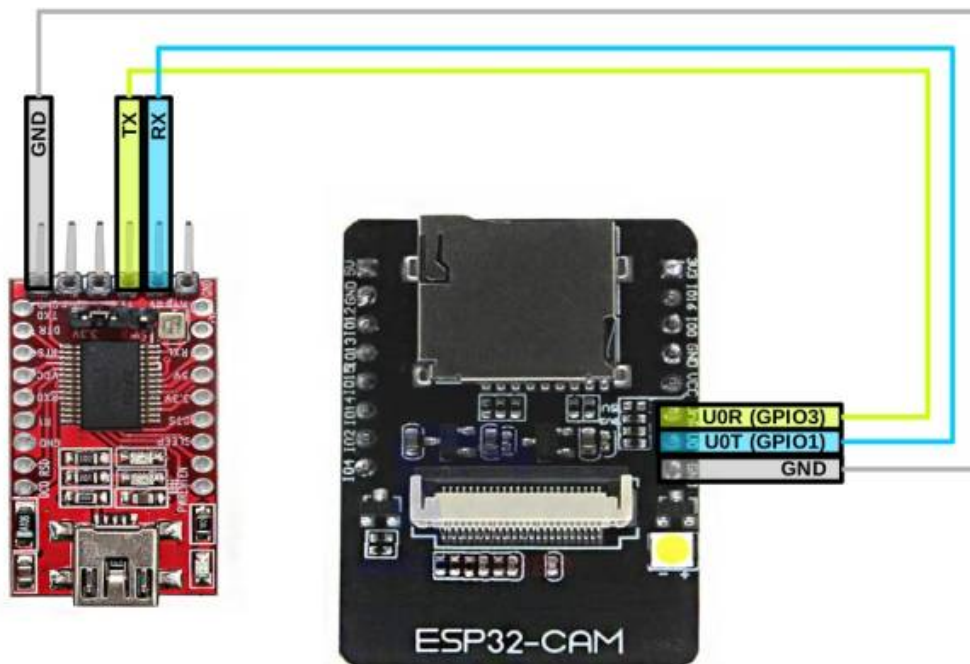
Attendre jusqu'à esptool.py est complètement fait avant de passer à autre chose. Clignoter un firmware peut prendre quelques minutes à compléter. Si vous rencontrez des problèmes tout en clignotant, essayez un taux de baud différent (-b) que le défaut 115200, comme -b 921600. La FAQ Tasmota peut vous aider à résoudre ce problème et d'autres.



Si esptool.py Pends à Connecting..., puis appuyez sur le bouton Redémarrer (RST) sur votre module ESP-cam.

Attendre jusqu'à esptool.py est fait. Ensuite, déconnectez votre adaptateur USB en TTL de votre ordinateur et retirez le sautoir en mode flash (GPIO0-GND) de la caméra ESP32.

### ESP32-CAM wiring: Serial monitor




Vous êtes maintenant prêt à alimenter votre nouveau dispositif Tasmota ESP32-cam à l'aide d'une alimentation indépendante. Dans les sections suivantes, nous verrons comment surveiller l'appareil via l'adaptateur TTL, qui est facultatif, puis comment alimenter l'appareil et le configurer, qui sont tous deux nécessaires.

## Console série

Cette section est optionnelle mais fortement recommandée pour faciliter les problèmes de puissance de dépannage, la spécification incorrecte des composants, les redémarrages aléatoires, et ainsi de suite. N'hésitez pas à passer au câblage autonome si vous n'avez pas envie de configurer une connexion série pour surveiller votre carte pendant que vous la configurez.

La configuration d'une connexion série à votre appareil Tasmota ESP32-cam vous permet de surveiller son état via une connexion filaire à votre ordinateur. Ceci est utile pour résoudre les problèmes qui se produisent avant que vous puissiez accéder à l'interface Web de l'appareil (par exemple, incapable de se connecter à son point d'accès, boucles de démarrage) et pendant les étapes de configuration initiales, car il n'y aura pas de marqueurs physiques de l'état de l'appareil sur lesquels se fier.

Pour créer une console série pour votre appareil, vous aurez besoin d'un (a) adaptateur USB vers TTL et d'un (b) émulateur de terminal. Si vous avez suivi ce guide, vous devriez déjà avoir un adaptateur USB à TTL, qui devrait maintenant être connecté à la carte ESP32-cam comme suit:

 Pour établir un moniteur série, nous utiliserons une application GNU appelée screen. Tout d'abord, ouvrez un terminal et vérifiez s'il est installé sur votre ordinateur ou non:

Last update: 2026/03/04 16:52 start:arduino:esp32b:espcam2:tasmota <https://www.fablab37110.chanterie37.fr/doku.php?id=start:arduino:esp32b:espcam2:tasmota&rev=1772639546>

---

From: <https://www.fablab37110.chanterie37.fr/> - **Castel'Lab le Fablab MJC de Château-Renault**

Permanent link: <https://www.fablab37110.chanterie37.fr/doku.php?id=start:arduino:esp32b:espcam2:tasmota&rev=1772639546>

Last update: **2026/03/04 16:52**

