

PDM-Labo | Première Mise en Service du module Raspberry CM3+

Sommaire

Résumé	1
1. Flasher la mémoire eMMC pour y installer Raspbian (via un ordinateur Ubuntu).....	1
1.1. Connexions matérielles.....	1
1.2. Installation de <code>rpiboot</code>	2
1.3. Dernières connexions et démarrage	2
1.4. Installation de Raspbian Buster	2
2. Démarrer sous Raspbian	3
3. Connecter un écran, clavier et souris.....	3
4. Connecter à internet	3
4.1. Utiliser un câble réseau et un module Ethernet over USB	3
4.2. Utiliser un dongle Wifi	3
5. Activer SSH.....	4
6. Se connecter via SSH	4
7. Configurer & Activer I2C & SPI	4
Ressources	5
Licence	5

Résumé

Document décrivant comment flasher la mémoire eMMC du module Raspberry CM3+, y installer et démarrer Raspbian, puis se connecter à internet, configurer l'accès via SSH et enfin activer les interfaces I2C et SPI.

1. Flasher la mémoire eMMC pour y installer Raspbian (via un ordinateur Ubuntu)

1.1. Connexions matérielles

- Installer le CM3+ dans la carte IO (voir vidéo [Youtube](#))
- Vérifier que le jumper J4 est en position **EN**

- Connecter la carte IO avec le périphérique Hôte USB (l'ordinateur ubuntu) via un câble micro USB (connecteur J15). NE PAS ALIMENTER LA CARTE IO A CE STADE.

1.2. Installation de `rpiboot`

Dans votre terminal sous ubuntu.

```
git clone --depth=1 https://github.com/raspberrypi/usbboot
cd usbboot

sudo apt update && sudo apt install libusb-1.0-0-dev

make

sudo ./rpiboot
```

1.3. Dernières connexions et démarrage

- Exécuter la commande `lsblk` et prendre connaissance de la liste des périphériques dans `/dev/`
- Finir de relier l'ordinateur avec la carte IO via le câble micro USB (connecteur J15)
- Alimenter la carte IO avec un câble micro USB (connecteur J2)
- `rpiboot` va détecter le CM3+ et envoyer le code de boot pour permettre l'accès à la eMMC (mémoire flash du CM3+)
- Une fois la commande terminée, un nouvel appareil est accessible : en général sur `/dev/sda` ou `/dev/sdb`

1.4. Installation de Raspbian Buster

- Télécharger [Raspbian Buster with desktop and recommended software](#)

```
// remplacer le nom du fichier .img à écrire et l'adresse de la carte IO
sudo dd if=raw_os_image_of_your_choice.img of=/dev/sdX bs=4MiB

// exemple
sudo dd if=2019-09-26-raspbian-buster-full.img of=/dev/sdb bs=4MiB
```

- La commande peut prendre du temps pour écrire l'OS sur le CM3+
- Une fois l'écriture terminée, déconnecter et reconnecter la carte IO
- Deux partitions doivent alors apparaître in `/dev`

```
/dev/sdX    <- Device
/dev/sdX1   <- First partition (FAT)
/dev/sdX2   <- Second partition (Linux filesystem)
```

2. Démarrer sous Raspbian

Deux possibilités pour démarrer le CM3+ sur le nouvel OS écrit sur la mémoire eMMC :

- le jumper J4 doit être en position DIS (disable) pour éviter un démarrage via USB
- déconnecter le câble micro USB du connecteur J15

Déconnecter et reconnecter l'alimentation de la carte IO (connecteur J2 - POWER IN) doit faire démarrer le CM3+ avec la mémoire eMMC.

3. Connecter un écran, clavier et souris.

Avec un multiplexeur USB, connecter clavier et souris à la carte IO.

Connecter un écran également à la carte IO.

Finir la configuration (langue, clavier, mot de passe principal) du CM3+ lors de son démarrage. Redémarrer.

4. Connecter à internet

4.1. Utiliser un câble réseau et un module Ethernet over USB

4.2. Utiliser un dongle Wifi

Module utilisé : **Wifi USB Adapter - N300 - EW-761UAn V2**

1. Vérifier que le dongle est bien reconnu avec `lsusb`
2. Installer `vim` avec `apt install -y vim` si, comme moi, vous n'utilisez pas l'éditeur `nano`
3. Editer le fichier de configuration du réseau Wifi : `sudo vim /etc/wpa_supplicant/wpa_supplicant.conf`

```
network={
  ssid="YOUR_NETWORK_SSID"
  psk="YOUR_NETWORK_PASSWORD"
}
```

4. Configurer le dongle pour éviter le mode veille : `sudo vim /etc/modprobe.d/8192cu.conf`

```
# Disable power management
options 8192cu rtw_power_mgnt=0 rtw_enusbss=0
```

5. Redémarrer le CM3+ avec `sudo reboot`

5. Activer SSH

```
sudo systemctl enable ssh
sudo systemctl start ssh
```

6. Se connecter via SSH

```
ssh pi@CM3_IP_ADDRESS

ssh -o "StrictHostKeyChecking=no" -o "UserKnownHostsFile=/dev/null" pi@CM3_IP_ADDRESS

// mot de passe par défaut est 'raspberrry'
```

7. Configurer & Activer I2C & SPI

```
sudo apt-get install -y python-smbus
sudo apt-get install -y i2c-tools
sudo raspi-config
```

- Puis activer I2C et SPI dans **Interfacing Options** via l'interface graphique qui s'affiche.
- Redémarrer avec `sudo reboot`
- Vérifier la liste des interfaces disponibles avec `ls /dev/i2c* /dev/spi*`

Activer des interfaces SPI supplémentaires

- Ouvrir le fichier avec `sudo vim /boot/config.txt` et y ajouter à la fin la ligne suivante

```
dtoverlay=spi1-3cs
```

- Redémarrer avec `sudo reboot` to the bottom of and rebooting. You'll then see the addition of some `/dev/spidev1.x` devices:
- De nouvelles interfaces doivent être apparues en tapant à nouveau `ls /dev/i2c* /dev/spi*`

Ressources

- [Flashing eMMC](#)
- [Download Raspbian](#)
- [Compute Module Datasheet](#)

Licence

```
<p style="text-align:center;">
```

```
  Ce document contributif est régi par les termes de la Licence <br/> <a  
href="https://creativecommons.org/licenses/by-sa/4.0/deed.fr">Creative Commons  
Attribution <br/> Partage dans les Mêmes Conditions 4.0 International</a>
```

```
</p>
```

```

```

```
<a href="https://agir.lowtech.fr"><p style="text-align:center;">CC-BY-SA 4.0 - Agir  
Low-Tech</p></a>
```

```
<a href="https://agir.lowtech.fr"><p style="text-  
align:center;">https://agir.lowtech.fr</p></a>
```