

B Configuration de la caméra, Procédures B

B.1 Procédures de configuration

B.1.1 Préparation du microcontrôleur

Lors du montage d'une nouvelle caméra, vous devez préparer le micro-contrôleur Particle Boron. Dans chaque paquet de particle Boron, vous trouverez le microcontrôleur ainsi qu'une antenne dans des sachets séparés. La première étape est de raccorder l'antenne au port u.fl sur la face supérieur du microcontrôleur tel que sur l'image suivante :



FIGURE 24 – Particle Boron assemblé

Ensuite insérer une carte SIM au format NANO dans le port dédié à cet usage sur la surface inférieure du microcontrôleur :



FIGURE 25 – Carte SIM correctement insérée

Si ce n'est pas déjà fait, vous devez préparer l'environnement de travail VSCode pour vous permettre de forcer l'usage d'une version de device OS, pour possiblement recompiler des programmes, pour activer et renommer un Boron et pour télécharger le programme de la caméra dans celui-ci. Il est suggéré de réaliser les étapes dans L'ordre suivant :

1. Configurez l'environnement ;
2. Mettez à jour le système d'exploitation du Boron (Particle update) ;
3. Installez le programme de configuration de l'APN et le programme standard (Particle flash -usb SetTelusAPN.bin et tinkr) ;
4. Activez et renommez le Boron (Particle identify, device add et device rename) ;
5. Ajustez la focale ;
6. Installez le bon device OS ainsi que DEHCam.bin.

Le tout est expliqué en détail dans les sections qui suivent.

B.1.2 Configuration de l'environnement VS Code et GitHub

La caméra autonome est programmée en "C"/"C++" à l'aide du logiciel gratuit VSCode développé par Microsoft et qui est une branche de visual studio. Afin de faciliter la programmation de l'appareil, il faut configurer l'environnement propre à "Particle". Après l'installation de VSCode, installez le "Particle Workbench"⁹.

Après l'installation terminée, lancez VSCode et dites "oui" à l'installation de dépendances et de "Toolchain". Ceci va précharger des outils et le logiciel d'exploitation du contrôleur¹⁰. Assurez vous d'installer la bonne version du toolchain (1.5.2 pour la première version de DEHCam). Le dépôt du code et des documents dits "open-source" sont disponibles sur la plateforme "GitHub" à l'URL suivant :

<https://github.com/MELCC-DEH/DEHCam>

vous pouvez cloner le dossier sur votre poste en cliquant sur le bouton "Code" de la page et en sélectionnant "Clone" ou "download ZIP". Il est recommandé de le déployer sur le disque dur local et dans un chemin de répertoire sans caractères d'espaces.

Dans VSCode, cliquez sur fichiers, puis ouvrir dossier et sélectionnez le répertoire racine du dossier que vous venez de télécharger.

B.1.3 Configuration d'un nouvel appareil

Raccordez le microcontrôleur à l'ordinateur portable grâce à un câble micro-USB à USB type A. Laissez windows installer les pilotes.

Dans VSCode, appuyez sur ctrl-shift + P et sélectionnez la commande "Particle : Launch CLI". Une console apparaîtra.

Activez le mode "DFU" dans le micro-contrôleur en maintenant le bouton "MODE" enfoncé et en appuyant et relâchant le bouton "RESET". Continuez de maintenir le bouton "MODE" jusqu'à ce que la lumière clignote jaune. Relâchez le bouton "MODE". Laissez À nouveau Windows installer les pilotes si il y a lieu.

Dans la console, entrez la commande "particle update", la console va mettre à jour le système d'exploitation du micro-contrôleur. La dernière version stable du Device OS sera installée sur le microcontrôleur.

Dans le dossier de projet de la caméra, assurez vous d'avoir le fichier "setTelusAPN.bin" dans le répertoire

9. <https://www.particle.io/workbench#installation>

10. <https://prerelease-docs.particle.io/tutorials/developer-tools/workbench/>

"DEHCam/Extras/DEHCam_Tools/Firmware/"

Réactivez le mode DFU et exécutez la commande :

```
>particle flash --usb C:/<chemin d'accès>/setTelusAPN.bin
```

Ceci configurera l'appareil pour utilisation de la carte SIM externe et programmera l'APN. La lumière bleue s'allume lorsque l'opération est terminée. Il est possible de changer les paramètres (dont l'APN) en recompilant ce dernier. Voir en Annexe E.
Réactivez à nouveau le mode DFU et entrez la commande suivante :

```
>particle flash --usb tinker
```

Ceci installe un programme bidon pour connecter le contrôleur au réseau cellulaire pour la première fois. Le contrôleur est connecté au réseau cellulaire lorsque la lumière est cyan ou constamment allumée verte. Le processus de connexion est en cours si elle scintille verte. Il est possible qu'il soit nécessaire de redémarrer l'appareil suite à une première connexion (bouton "RESET").

Suite à cette procédure il faut lier le contrôleur au Compte Particle¹¹. Voir section suivante ou 5.12 pour configuration Cloud.

B.1.4 Activation du micro-contrôleur (localement)

Afin d'avoir accès aux fonctionnalités du cloud particle, il faut "réclamer" notre appareil. Exécutez la commande :

```
Particle Login
```

Entrez les identifiants liés à votre compte administratif principal.
Lancez ensuite la console Particle CLI à l'aide du menu ctrl-shift + P dans VSCode.

Maintenez le bouton "MODE" jusqu'à ce que la lumière clignote bleue, entrez ensuite la commande :

```
Particle identify
```

Notez le champ "Device ID" en le copiant dans le presse-papier.

Entrez la commande :

```
particle device add <DEVICE ID>
```

En ajoutant à la fin de la commande, le "Device ID" que vous avez noté. Vous devriez voir un message indiquant : "Successfully claimed device" avec votre "Device ID" après.

Enfin, renommez l'appareil à l'aide de la commande suivante :

11. <https://support.particle.io/hc/en-us/articles/360039741113>

```
particle device rename <DEVICE_ID> "<No_BDH>"
```

Remplacez le No_BDH par le numéro de BDH de la caméra en conservant les guillemets et la section "DEVICE_ID" par le "Device ID" de l'appareil.

B.1.5 Ajustement de la focale

L'ajustement d'usine du focus est généralement adéquat pour le scénario d'utilisation de notre caméra. Dans le cas où le focus n'est pas adéquat, il faut d'abord installer un firmware permettant de visionner l'image de la caméra en "temps réel". Ce fichier fonctionne avec le device OS 2.0.1 il se situe dans :

```
"../DEHCam/Extras/DEHCam_Tools/Firmware/OV2640_test_201.bin"
```

suivez la procédure de la section 4.2.6 avec ce répertoire pour changer le programme. Lorsque le programme s'exécute sur le Boron (lumière blanche qui scintille), raccorder le câble usb si ce n'est déjà fait et exécutez le logiciel "ArduCAM_Host_V2.exe" situé dans le répertoire suivant :

```
"../DEHCam/Extras/DEHCam_Tools/ArduCAM_Host_V2.0_Windows/"
```

Attention cette application n'est pas très stable et peut planter fréquemment. Sélectionnez le bon port COM et 115200 bps puis "Open". Vous pouvez sélectionner plusieurs résolutions différentes, étant donné que le transfert se fait par lien série, de plus grande résolution prendrons un temps significatif de transfert pouvant atteindre 10-15 secondes. Pour avoir de nouvelles images en continu, sélectionnez le mode "continuous". N'oubliez pas de remettre la bonne version de device OS et le bon firmware dans le contrôleur à la fin de votre manipulation. Voir section suivante.

B.1.6 Téléversement du microcode principal dans le micro-contrôleur (localement)

À partir de l'environnement configuré sur VSCode, sélectionnez la bonne version d'OS, le bon appareil (boron) et entrez le no BDH, identifiez vous au besoin (Particle login). Lancez le menu ctrl-shift + p et sélectionnez :

```
Particle: Flash application and device OS (local)
```

L'appareil doit être raccordé par USB, la séquence va activer automatiquement le mode DFU.

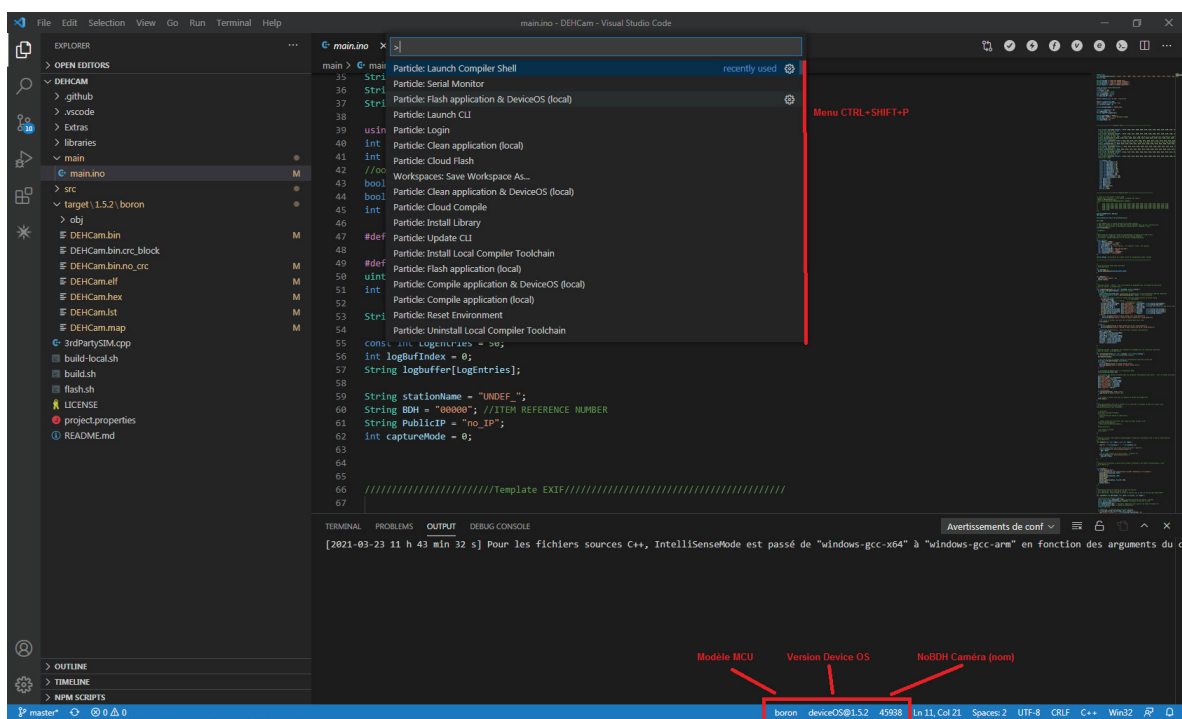


FIGURE 26 – Environnement VSCode + Particle workbench