## Procédure de mise en service d'un MMDVM en partant de zéro

Il existe peut-être une révision plus récente de ce document. Avant toute chose, pensez à <u>télécharger la dernière révision</u>. Tous les commentaires sont les bienvenus. Vous pouvez les déposer sur mon blog <u>en cliquant ici</u>.

Ce document recense l'intégralité des actions réalisées pour mettre en service mes relais MMDVM. Cependant, malgré toute l'attention et le temps passé à le rédiger, il peut ne plus être totalement applicable lorsque vous le consulterez. J'ai par exemple dû procéder à une importante mise à jour lorsque j'ai voulu mettre un nouveau MMDVM en service, en 2022. Le système d'exploitation de la carte **Raspberry Pi** ainsi que le logiciel chargé dans la carte **Nucleo STM32F446** et l'écran **Nextion** évoluent, ce qui donne lieu à des incompatibilités ou contraint à utiliser d'autres méthodes pour mettre l'ensemble en œuvre.

Ne soyez pas rebutés par la quantité de choses à effectuer. Dites-vous qu'il ne vous reste presque plus qu'à lire et appliquer ce qui est écrit. Imaginez le temps que ça vous prendrait en partant de zéro, sans la moindre infortmation.

L'interface utilisée dans mon cas est celle publiée par F5UII / Christian : <u>https://www.f5uii.net/mmdvm-nucleo-stm32-pcb-shield/</u> Je dispose de quelques circuits imprimés inutilisés. Si ça vous intéresse <u>laissez-moi un message sur mon blog</u>. On trouve quelques cartes **Nucleo STM32F446** à partir d'une trentaine d'euros sur Aliexpress, mais cela reste assez rare. Le prix le plus couramment pratiqué est plutôt autour de 50€. Quelqu'un a du se rendre compte qu'il n'était pas nécessaire de les brader, ou la pénurie mondiale de composants est passée par là, car les deux cartes que j'ai acheté la-bas en 2019 et 2021 m'ont coûté un peu moins de 21€ chacune !

# Présentation rapide du concept « MMDVM »

Le terme MMDVM est l'acronyme de Multi Mode Digital Voice Modem : Modulateur Démodulateur Multi-Modes Vocaux Numériques. La force de ce système est de permettre l'utilisation d'équipements radio fonctionnant initialement uniquement en FM, afin de réaliser un relais radio supportant de nombreux modes numériques : DMR, C4FM, D-STAR, NXDN, DAPNET (POCSAG) et évidemment FM.

Un MMDVM est composé des éléments suivants :

- Un équipement radio dédié à la réception des transmissions distantes, pouvant fournir à une interface externe un signal analogique non filtré véhiculant le signal démodulé reçu\* ;
- Un équipement radio dédié à l'émission, à destination des équipements radio distants, capable de transmettre un signal audio analogique « large bande » via son entrée non filtrée\* ;
- Une interface analogique placée entre les équipements radio et la carte processeur :
  - Amplification ou atténuation des signaux audio analogiques (celui reçu et celui émis), ainsi que leur remise en forme ;
  - Interface avec les signaux TOR (Tout Ou Rien) des équipements radio : PTT, indication de détection de porteuse, signal analogique indiquant la force du signal reçu (RSSI : Received Signal Strength Indication) ;
  - Indications visuelles par l'intermédiaire de LED et éventuellement d'un afficheur de type alphanumérique ou graphique.
  - Microcontrôleur (carte Arduino Mega) ou processeur (carte STM32) assurant la conversion analogique/numérique et numérique/analogique.
- Une carte **Raspberry Pi** assurant le traitement des signaux audio numériques, les fonctions de base d'un relais (balise en morse par exemple) et l'interface avec les serveurs Internet d'interconnexion des relais MMDVM.
- \* Si le relais est utilisé en mode « Hotspot » (simplex), on n'utilise qu'une seule radio en tant qu'émetteur et récepteur.

Des équipements intégrant sur une même carte électronique la majorité de ces fonctions sont aussi disponibles. Leur faible puissance d'émission restreint cependant leur périmètre de couverture radio au sein du domicile de l'opérateur.

Il n'est pas possible d'utiliser la suite de logiciels MMDVM en tant que convertisseur DMR vers FM analogique par exemple. Lorsqu'un relais MMDVM est utilisé en mode DMR, il reçoit une émission DMR, la traite en interne, et la retransmet uniquement en DMR. La fonction « Multi Modes » est donc assurée de façon exclusive, soit un mode à la fois, sans conversion d'un mode à l'autre. Certains serveurs d'interconnexion de relais MMDVM offrent des interfaces entre modes numériques, mais le relais MMDVM lui-même ne gère qu'un des modes à la fois sur ses interfaces de réception et d'émission.

# Installation du système d'exploitation de la carte Raspberry Pi

La carte Raspberry Pi nécessite un système d'exploitation pour fonctionner, comme tout ordinateur. Le choix de celui-ci est laissé libre tant qu'il est basé sur le système Linux, car les logiciels permettant la fonction MMDVM seront compilés et installés par la suite. Afin de nous simplifier la vie ici, nous allons utiliser un des variants du système d'exploitation mis à disposition par la fondation Raspberry Pi. Nous allons nous orienter vers une version sans interface graphique, puisque toutes les actions seront réalisées sous forme de lignes de commande à saisir au clavier sur une interface « terminal ». Ne soyez pas rebutés de suite par cette façon de faire, tout sera expliqué en détails dans ce document. Ce choix de système d'exploitation léger permet d'utiliser une génération de carte Raspberry Pi ancienne, et une carte mémoire SD de capacité relativement faible.

## Chargement du système d'exploitation

Comme indiqué précédemment, nous utiliserons **Raspberry Pi OS**, qui est conçu par l'équipe qui a créé la carte Raspberry Pi. Son téléchargement et le stockage sur la carte mémoire SD sont simplifiés grâce à l'outil mis à disposition par cette équipe, disponible pour Windows, macOS et Linux, et accessible à l'adresse suivante : <u>https://www.raspberrypi.com/software/</u>.

Si vous avez déjà installé le système d'exploitation **Raspbian** dans le passé, vous avez pour habitude d'utiliser le compte utilisateur défini par défaut, à savoir l'utilisateur **pi**, avec le mot de passe **raspberry**.

La sécurité est maintenant accrue, afin que les cartes **Raspberry Pi** accessibles depuis Internet ne puissent plus être « piratées » facilement en utilisant ce compte défini par défaut. Avec les versions récentes du système d'exploitation dédié à la **Raspberry Pi**, il appartient à l'utilisateur de définir lui-même l'identifiant et le mot de passe. Il n'existe donc plus de compte prédéfini par défaut. Comme nous utiliserons cette carte sans y connecter d'écran, de clavier et de souris, il sera impossible de définir ce compte utilisateur si celui-ci n'a pas été choisi avant le chargement du système d'exploitation sur la carte SD, via le logiciel **Raspberry Pi Imager**. Suivez donc avec la plus grande attention les étapes décrites ci-dessous.

## Installation de Raspberry Pi Imager sous Windows

Rien de bien compliqué ici, il suffit de confirmer l'installation et de valider toutes les étapes :

## Cliquer sur **Install** :



L'installation s'exécute :

😙 Raspberry Pi Imager		
Installing Please wait while Raspberry Pi Imager is being	g installed.	
Stopping Raspberry Pi Imager		
Show details		
Raspberry Pi Imager	Rack Next >	
	C Dark	

À l'issue de celle-ci, une nouvelle fenêtre s'affiche. Laisser la case Run Raspberry Pi Imager cochée et cliquer sur Finish :



## Paramétrage de Raspberry Pi Imager et chargement du système d'exploitation sur la carte SD

Dans ces étapes, je définis les identifiants qui étaient jusqu'ici ceux par défaut sur ce système d'exploitation, car certaines étapes décrites dans ce document sont antérieures à cette nouvelle règle de sécurité. Cela me permet d'éditer uniquement les nouvelles actions à effectuer, sans procéder à des modifications importantes de ce document qui dépasse les 100 pages. De plus, mon MMDVM est connecté à un modem 4G dont l'abonnement pour particuliers ne permet pas d'ouvrir de port pour donner un accès à la console Linux de ma carte Raspberry Pi depuis Internet. C'est une bonne sécurité, mais cela rend le MMDVM totalement incontrôlable à distance s'il est installé sur un point haut autre que le domicile du responsable.

Si vous définissez d'autres identifiants pour l'utilisateur, vous devrez saisir ceux-ci à la place de **pi** et **raspberry** pour les étapes décrites dans le reste de ce document.

Le logiciel Raspberry Pi Imager est exécuté automatiquement :



Cliquer sur Raspberry Pi OS Lite (32-bit) :



Cliquer sur CHOISISSEZ L'OS, puis sur Raspberry Pi OS (other) :



Cliquer sur le bouton **CHOISISSEZ LE...** pour définir la lettre du disque affectée à votre carte SD :



Cliquer sur le bouton de paramètres tout en bas à droite 🥸 :



Faire descendre la barre de défilement vertical afin de définir les identifiants utilisateur (ici le mot de passe sera **raspberry**) :



Cocher la case **Enable SSH** afin d'autoriser les connexions IP à la console Linux :



Cliquer sur **SAVE** pour quitter l'édition des paramètres, puis sur **ÉCRIRE** pour lancer l'installation sur la carte SD :



Respberry Pi Imager v1.7.2

Accepter la mise en garde concernant la perte des données :

Une fois le chargement terminé sur la carte SD, cliquer sur **CONTINUER** et la placer dans le support de la carte Raspberry PI



Attendre la fin de l'écriture des données sur la carte SD :



## Avant de commencer

## Activation de l'accès à la console par IP via le protocole SSH

La majorité des étapes décrites ci-dessous est effectuée en utilisant un client SSH. Ce logiciel permet d'exécuter des commandes sur la carte Raspberry Pi par l'intermédiaire de son lien réseau. Pour des raisons de sécurité, l'accès SSH n'est pas activé par défaut sur le système d'exploitation Raspbian. Si l'activation n'a pas été faite lors du chargement du système d'exploitation décrit dans le paragraphe précédent, il vous faudra créer un fichier vierge nommé ssh dans la racine de la carte SD, à partir de Windows. Depuis la racine de la carte SD, faire un clic droit et aller sur **Nouveau** dans le menu déroulant, puis cliquer sur **Document texte** :

Kenoris       Nom       Modifié le       Type       Talle         Bureau       froupAx.dat       froupAx.dat       21/03/2021 11:26       Fichier DAT       9         Emplacements récents       froupAx.dat       Affichage       1       21/03/2021 11:26       Fichier DAT       9         Bibliothèques       froupAx.dat       Affichage       1       21/03/2021 11:26       Fichier DAT       9         Bibliothèques       froupAc.dat       tartatk.eff       Start&da.eff       21/03/2021 11:26       Fichier DAT       6         Groupe residentiel       start&d.eff       coller       21/03/2021 11:25       Fichier DAT       6         © Ordinateur       Golge resourci       Coller       21/03/2021 11:25       Fichier ELF       2177         © Disque local (C.)       froup, x.dat       froup, s.dat       Coller le de/plecement       Ctrl-Z       21/03/2021 11:25       Fichier DAT       11         Disque local (C.)       froup, d.dat       froup, d.dat       froup, d.dat       Coller       21/03/2021 11:25       Fichier DAT       11         Bibliothéques       froup, d.dat       froup, d.dat       froup, d.dat       froup, d.dat       11       21/03/2021 11:25       Fichier DAT       11       21/03/2021 11:25       Fichier DA	Organiser 🔻 Partager avec 🔻 Graver	Nouveau dossier							8== 🕶 🔳	
Bureau       foup5x.det       21/03/2021 11:26       Fichier DAT       9         Bureau       foup4c.det       11/03/2021 11:26       Fichier DAT       9         Bureau       foup4c.det       Actualiser       12/03/2021 11:26       Fichier DAT       6         Bureau       foup4c.det       Actualiser       12/03/2021 11:26       Fichier DAT       6         Groupe résidentiel       start4c.def       Coller       21/03/2021 11:25       Fichier ELF       775         Groupe résidentiel       foup,2.det       Goler le raccourci       Coller       21/03/2021 11:25       Fichier ELF       2172         Disque local (C)       foup,2.det       foup,2.det       Open Folder as PyCharm Community Edition Project       21/03/2021 11:25       Fichier DAT       11         Disque local (C)       foup,2.det       foup.p1:1:25       Fichier DAT       4       21/03/2021 11:25       Fichier DAT       4         Bureau       foup.cd.det       foup.cd.det       Goler le raccourci <th>🔶 Favoris</th> <th>Nom</th> <th></th> <th></th> <th></th> <th></th> <th>Modifié le</th> <th>Туре</th> <th>Taille</th> <th></th>	🔶 Favoris	Nom					Modifié le	Туре	Taille	
<sup>1</sup> Emplacements récents <sup>1</sup> LUENCE broadcom <sup>1</sup> Affichage <sup>1</sup> LUENCE broadcom <sup>1</sup> LUA3/2021 11:26 <sup>1</sup> Fichier DAT <sup>1</sup> LUA3/2021 11:26 <sup>1</sup> Fichier ELF <sup>1</sup> ZU33/2021 11:26 <sup>1</sup> Fichier ELF <sup>1</sup> ZU33/2021 11:25 <sup>1</sup> Fichier DAT	E Bureau	fixup4x.dat					21/03/2021 11:26	Fichier DAT	9 Ko	
Image: TeleChargements       froup4db.dat       Trier par       21/03/2021 11:26       Fichier DAT       9         Image: TeleChargements       froup4db.dat       froup4db.dat       7       8       21/03/2021 11:26       Fichier DAT       4         Image: TeleChargements       froup4db.dat       froup4db.dat       7       8       21/03/2021 11:26       Fichier DAT       4         Image: TeleChargements       froup4db.dat       froup4db.dat       7       6       21/03/2021 11:26       Fichier DAT       6         Image: TeleChargements       froup4db.dat       froup4db.dat       7       6       21/03/2021 11:26       Fichier DAT       6         Image: TeleChargements       froup4db.dat       froup4db.dat       Personnaliser ce dosiser       21/03/2021 11:26       Fichier ELF       7075         Image: TeleChargements       Coller       Coller le raccourci       Culv3/2021 11:25       Fichier BN       22       21/03/2021 11:25       Fichier BN       22       21/03/2021 11:25       Fichier DAT       11       21/03/2021 11:25       Fichier DAT       11       21/03/2021 11:25       Fichier DAT       11       21/03/2021 11:25       Fichier DAT       4       4       4       4       4       4       4       4       4       4	Emplacements récents	LICENCE.broadcom		Affichage	÷	1	21/03/2021 11:26	Fichier BROADCOM	2 Ko	
image: bibliothèques       image: data       Regrouper par       >       21/03/2021 11:26       Fichier DAT       4         image: data       image: data       image: data       Actualiser       21/03/2021 11:26       Fichier DAT       6         image: data       image: data       image: data       image: data       21/03/2021 11:26       Fichier DAT       6         image: data       image: data       image: data       image: data       21/03/2021 11:26       Fichier DAT       6         image: data       image: data       image: data       image: data       21/03/2021 11:26       Fichier DAT       7         image: data       image: data       image: data       image: data       image: data       21/03/2021 11:25       Fichier BAT       1         image: data       image: data       image: data       image: data       image: data       21/03/2021 11:25       Fichier BAT       11         image: data       image: data       image: data       image: data       image: data       image: data       11         image: data       image: data       image: data       image: data       11       image: data       11 <td>〕 Téléchargements</td> <td>fixup4db.dat</td> <td></td> <td>Trier par</td> <td>+</td> <td>ι.</td> <td>21/03/2021 11:26</td> <td>Fichier DAT</td> <td>9 Ko</td> <td></td>	〕 Téléchargements	fixup4db.dat		Trier par	+	ι.	21/03/2021 11:26	Fichier DAT	9 Ko	
Bibliothèques       □ foup4 dat 1 star4.eff       Actualiser       21.03/202111:25       Fichine DAT       6         G'oroupe résidentiel       1 star4.eff       Coller       21.03/202111:25       Fichine ELF       3636         Ordinateur       Dostocde.bin       Coller déplacement       Ctrl-2       21.03/202111:25       Fichine DAT       6         Ordinateur       Dostocde.bin       Edile le raccourci       21.03/202111:25       Fichine DAT       111         Annuler le déplacement       Ctrl-2       21.03/202111:25       Fichine DAT       111         Annuler le déplacement       Ctrl-2       21.03/202111:25       Fichine DAT       111         Dost (G)       Fixupdb.dat       Partager avec       21.03/202111:25       Fichine DAT       111         Toroup dat       Fichine DAT       111       Fichine DAT       111         Dost (G)       Fixupdb.dat       Partager avec       21.03/202111:25       Fichine DAT       4         1 start_vd.eff       Fichine CLF       Partager avec       110       103/202111:25       Fichine DAT       4         1 start_vd.eff       Fichine DAT       Fichine DAT       8       103/20211:25       Fichine DAT       4         1 start_vd.eff       Fichine DAT       Fichine	🥽 Bibliothèques	fixup4cd.dat		Regrouper par	•	ι.	21/03/2021 11:26	Fichier DAT	4 Ko	
Bibliothèques       is tarté.deff       21/03/2021.11:26       Fichier E.F       2912         Groupe résidentiel       istarté.deff       Coller       21/03/2021.11:25       Fichier E.F       3630         Ordinateur       boot (céle in tarté.deff       Coller / tarté.deff       Coller / tarté.deff       21/03/2021.11:25       Fichier E.F       21/03         Ordinateur       boot (céle in tarté.deff       Coller / tarté.deff       Coller / tarté.deff       21/03/2021.11:25       Fichier E.F       21/03         Disque local (C)       fixupdb.dt       Gopen folder as PyCharm Community Edition Project       21/03/2021.11:25       Fichier DAT       111         Partager avec       YumMerge       21/03/2021.11:25       Fichier DAT       111         21/03/2021.11:25       Fichier DAT       4       YumMerge       21/03/2021.11:25       Fichier DAT       111         21/03/2021.11:25       Fichier DAT       4       YumMerge       21/03/2021.11:25       Fichier DAT       4         10 tarté.deff       Nouveau       YumMerge       21/03/2021.11:25       Fichier DAT       4         10 tarté.deff       Propriétés       Propriétés       Propriétés       21/03/2021.11:25       Fichier DAT       4         10 tarté.deff       Propriétés       Propriétés		fixup4.dat		Actualiser		ι.	21/03/2021 11:26	Fichier DAT	6 Ko	
Groupe résidentiel       21/03/2021 11:25       Fichier EF       3636         Groupe résidentiel       1 startéd.eff       1 startéd.eff       21/03/2021 11:25       Fichier EF       775         Ordinateur       Doctocobe.inn       Gopp. Folder as PyCharm Community Edition Project       21/03/2021 11:25       Fichier EF       717         Jorgue local (C)       Fixup_ed.odat       Image bitmap       21/03/2021 11:25       Fichier EF       21/03/2021 11:25       Fichier DAT       11         Partager avec       21/03/2021 11:25       Fichier DAT       11       21/03/2021 11:25       Fichier DAT       48         Graup de attr_uelf       Nouveau       Image bitmap       21/03/2021 11:25       Fichier DAT       48         Tatr_uelf       Nouveau       Image bitmap       775       3618       775         startef       COPYINGinux       Image bitmap       775       21/03/2021 11:25       Fichier DAT       48	🗃 Bibliothèques	start4x.elf		Deservation of deservation		ι.	21/03/2021 11:26	Fichier ELF	2 912 Ko	
Groupe résidentiel       atafdc.def       Coller       21/03/2021.11:25       Fichier EF       775         Ordinateur       bootode.bin       Goller le raccourci       21/03/2021.11:25       Fichier EF       21/03/2021.11:25       Fichier DAT       11         Partager avec       VinMerge       21/03/2021.11:25       Fichier DAT       4		start4db.elf		Personnaliser ce dossier		ι.	21/03/2021 11:26	Fichier ELF	3 636 Ko	
Ordinateur       21/03/2021 11:25       Fichier EF       21/77         Disque local (C:)       foup_xdat       Gopen folder as PyCharm Community Edition Project       21/03/2021 11:25       Fichier BAT       111         > boot (G:)       foup_dat       Gopen folder as PyCharm Community Edition Project       21/03/2021 11:25       Fichier DAT       111         > boot (G:)       foup_dat       Gopen folder as PyCharm Community Edition Project       21/03/2021 11:25       Fichier DAT       111         > boot (G:)       foup_dat       WinMerge       21/03/2021 11:25       Fichier DAT       111         21/03/2021 11:25       Fichier DAT       111       21/03/2021 11:25       Fichier DAT       111         > boot (G:)       foup_dat       foup_dat       WinMerge       21/03/2021 11:25       Fichier DAT       111         > data t_xelf       Nouveau       it       it       Bossier       3618         > data t_cdeff       start_deff       Start_cdeff       it       6000000000000000000000000000000000000	🖏 Groupe résidentiel	start4cd.elf		Coller		ι.	21/03/2021 11:25	Fichier ELF	775 Ko	
Ordinateur       Dototodebin       Annuler (e deplacement       Ctrl-Z       21/03/2021125       Fichier INN       52         Disque local (C)       fixup_xdat       Image integration       21/03/2021125       Fichier DAT       11         Dotot (G)       fixup_ddat       Image integration       21/03/2021125       Fichier DAT       11         Partager avec       21/03/2021125       Fichier DAT       4         Start_xbelf       Nouveau       Image integration       3618         start_del eff       Porpietés       21/03/2021125       Fichier DAT       4         Start_del eff       Propiétés       Image integration       705         start_del eff       Propiétés       Image integration       705         Start_del eff       Propiétés       Image integration       284         COPVIMS/inax       Image integration       705       286         bern2711-rpi-4-b.dtb       Image integration       705       286         bern2711-rpi-4-b.dtb       Image integration       72       284         bern2710-rpi-3-b.dtb       Image integration       72       286         bern2710-rpi-3-b.dtb       Image integration       72       72         bern2710-rpi-3-b.dtb       Image integration       <		start4.elf		Coller le raccourci		ι.	21/03/2021 11:25	Fichier ELF	2 177 Ko	
Image local (C)       Image local (C)       Image local (C)       Image local (C)         Image local (C)       Image local (C)       Image local (C)       Image local (C)         Image local (C)       Image local (C)       Image local (C)       Image local (C)         Image local (C)       Image local (C)       Image local (C)       Image local (C)         Image local (C)       Image local (C)       Image local (C)       Image local (C)         Image local (C)       Image local (C)       Image local (C)       Image local (C)         Image local (C)       Image local (C)       Image local (C)       Image local (C)         Image local (C)       Image local (C)       Image local (C)       Image local (C)         Image local (C)       Image local (C)       Image local (C)       Image local (C)       Image local (C)         Image local (C)       Image local (C)       Image local (C)       Image local (C)       Image local (C)       Image local (C)         Image local (C)       Image local (C)       Image local (C)       Image local (C)       Image local (C)       Image local (C)       Image local (C)       Image local (C)       Image local (C)       Image local (C)       Image local (C)       Image local (C)       Image local (C)       Image local (C)       Image local (C)       Image local (C)	Ordinateur	bootcode.bin		Annuler le déplacement	Ctrl+Z	ι.	21/03/2021 11:25	Fichier BIN	52 Ko	
bot (6) four_ddat four_ddat virMerge 21/03/2021 11:25 Fichier DAT 11 froup_ddat virMerge 21/03/2021 11:25 Fichier DAT 41 21/03/2021 11:25 Fichier DAT 44 21/03/2021 11:25 Fichier DAT 44 20/03/2021 11:25 Fichier DAT 44 20/03/2021 11:25 Fichier DAT 44 20/03/2021 11:25 Fichier DAT 44 20/03/2021 11:25 Fichier DAT 46 20/03/2021 11:25 Fichier DAT 47 20/03/202	🏭 Disque local (C:)	fixup_x.dat	PC	Open Folder as PyCharm Community Edition Project		ι.	21/03/2021 11:25	Fichier DAT	11 Ko	
foup-cddat     Windwerge     21/03/2021 11:25     Fichier DAT     4       foup-dat     Windwerge     21/03/2021 11:25     Fichier DAT     8       start_xceff     Nouveau     Image bitmap     3618       start_deleff     Propriétés     Image bitmap     757       start.db.eff     Propriétés     Image bitmap     755       start.db.eff     Propriétés     Image bitmap     755       start.db.eff     Desin OpenDocument     149       bern2711-rpi-40.dtb     Image bitmap     755       bern2711-rpi-40.dtb     Image bitmap     755       bern2711-rpi-40.dtb     Image bitmap     765       bern2711-rpi-40.dtb     Image bitmap     755       bern2711-rpi-40.dtb     Image bitmap     755       bern2711-rpi-40.dtb     Image bitmap     755       bern2711-rpi-40.dtb     Image bitmap     755       bern2710-rpi-3-b.dtb     Image bitmap <td< td=""><td>👝 boot (G:)</td><td>fixup_db.dat</td><td></td><td>Partager avec</td><td>,</td><td>ι.</td><td>21/03/2021 11:25</td><td>Fichier DAT</td><td>11 Ko</td><td></td></td<>	👝 boot (G:)	fixup_db.dat		Partager avec	,	ι.	21/03/2021 11:25	Fichier DAT	11 Ko	
foupdat       L2/03/20211225       Fichier DAT       8         start_xcif       Novreau       Dossier       3618         start_db.ef       Propriétés       Image bitmap       775         start_delf       Image bitmap       775         start_delf       Image bitmap       2884         COP/NKGInux       Image bitmap       2884         bcm2711-rpi-cm4.dtb       Image bitmap       2884         bcm2711-rpi-t4-b.dtb       Image bitmap       2884         bcm2710-rpi-3-b.dtb       Image bitmap		fixup_cd.dat	-	WinMerge		ι.	21/03/2021 11:25	Fichier DAT	4 Ko	
istart_xelf     Nouveau     istart_xelf     3618       istart_xelf     Propriétés     if and istart and and istart and and istart and		📄 fixup.dat	~	winneige		L	21/03/2021 11:25	Fichier DAT	8 Ko	
jstart.db.eff     Propriétés     Image bitmap     775       jstart.db.eff     Start.db.eff     Start.db.eff     284       COPVING/inax     Context     284       COPVING/inax     Context     284       DCOPVING/inax     Desin OpenDocument     19       bcm2711-rpi-40.dtb     Présentation OpenDocument     48       bcm2711-rpi-40.dtb     Context     284       bcm2711-rpi-40.dtb     Context     284       bcm2711-rpi-40.dtb     Context     284       bcm2710-rpi-3.btb     Context     284       bcm2710-rpi-3.btb     Document Rich Text     284       bcm2710-rpi-3-b.dtb     Document Rich Text     284		start_x.elf		Nouveau	+		Dossier		3 618 Ko	
istar_cd.eff     Image bitmap     775       istar_cd.eff     Image bitmap     2884       COPVING.inux     Image bitmap     2884       bcm2711-rpi-cm4.dtb     Image bitmap     2884       bcm2711-rpi-th.dtb     Image bitmap     2884       bcm2710-rpi-cm3.dtb     Image bitmap     275       bcm2710-rpi-3b.dtb     Image bitmap     2884		start_db.elf		Propriétés		2	Raccourci		4 683 Ko	
start.eff     2884       COPVING.linux     Image: Contact     19       bcm2711-rpi-cm4.dtb     Image: Contact     19       bcm2711-rpi-cm4.dtb     Image: Contact     48       bcm2711-rpi-4b.dtb     Image: Contact     48       bcm2711-rpi-4b.dtb     Image: Contact     48       bcm2711-rpi-4b.dtb     Image: Contact     48       bcm2711-rpi-4b.dtb     Image: Contact     28       bcm2711-rpi-4b.dtb     Image: Contact     48       bcm2710-rpi-3-b.dtb     Image: Contact     28       bcm2710-rpi-3-b.plus.dtb     Image: Contact     29		start_cd.elf	_				Image bitmap		775 Ko	
COPVING.inux         Image: Copving: Copving.inux         Image: Copving: Copving		start.elf					Contact		2 884 Ko	
bcm2711-rpi-cm4.db     Présentation OpenDocument     49       bcm2711-rpi-400.dtb     Classeur OpenDocument     48       bcm2711-rpi-4.b.dtb     Classeur OpenDocument     48       bcm2710-rpi-cm3.dtb     Texte OpenDocument     27       bcm2710-rpi-sb.dtb     Document Rich Text     28       bcm2710-rpi-3-b.dtb     Document texte     29       bcm2710-rpi-2.b.dtb     Document texte     29		COPYING.linux				ାର୍ଦ୍ଧ	Dessin OpenDocu	ment	19 Ko	
bcm2711-rpi-40.dtb         If Classeur OpenDocument         48           bcm2711-rpi-4b.dtb         If Teste OpenDocument         48           bcm2710-rpi-7m3.dtb         If Teste OpenDocument         27           bcm2710-rpi-3b.dtb         If Document Rich Test         28           bcm2710-rpi-3b-buls.dtb         Document teste         29           bcm2710-rpi-2b.dtb         Document compressé         27		bcm2711-rpi-cm4.d	tb				Présentation Oper	Document	49 Ko	
bcm2710-rpi-4-b.dtb         Texte OpenDocument         27           bcm2710-rpi-3-b.dtb         Document Rich Text         28           bcm2710-rpi-3-b.dtb         Document compressé         28           bcm2710-rpi-3-b.dtb         Document compressé         29		bcm2/11-rpi-400.dt	b				Classeur OpenDoc	ument	48 Ko	
bcm2710-rpi-cm3.db 2/ bcm2710-rpi-3-b.db 20 bcm2710-rpi-3-b.db 20 bcm2710-rpi-3-b.db 20 bcm2710-rpi-2-b.db 20		bcm2711-rpi-4-b.dt	b				Texte OpenDocum	ient	48 Ko	
bcm2710-rpi-3-batb         28           bcm2710-rpi-3-br]us.dtb         0cument texte         29           bcm2710-rpi-3-batb         0cssier compressé         27		bcm2/10-rpi-cm3.d	tb				Document Rich Te	xt	27 Ko	
berr210-rp-3-buto 22		bcm2/10-rpi-3-b.dt	0				Document texte		28 Ko	
DCm2/10-rpi-2-0.0t0		bcm2/10-rpi-3-b-pl	us.atb				Dossier compresse		29 Ko	
🕒 herre 2700 ari 2 habb		bcm2/10-rpi-2-b.dt	D L				Porte-documents		27 Ko	
ben2709 mi and the 2020 2011 111 Stebies DTP 25		bcm2709-rpi-2-0.dt	нь			_	21/02/2021 11:11	Fichier DTR	21 KO 25 Ko	



Remplacer le nom du fichier par **ssh**, sans aucune extension de fichier, et valider en appuyant sur la touche **Entrée** :

Validez cette action en cliquant sur **Oui** :



Éjectez proprement le disque SD depuis la barre des tâches de Windows, insérez à nouveau la carte SD dans le lecteur de la Raspberry Pi et mettez-la sous tension.

Retrouvez l'adresse IP de la carte Raspberry Pi depuis l'interface web de votre routeur ou de votre box Internet. Cette étape est très spécifique compte tenu de la multitude d'équipements réseau pouvant attribuer une adresse IP, et ne sera donc pas détaillée dans ce document. Questionnez votre moteur de recherche favori en incluant la référence de votre équipement dans les mots clés de recherche. Il est par ailleurs très vivement conseillé de définir un bail DHCP permanent pour la carte Raspberry Pi, afin que son adresse IP ne change pas à chaque mise sous tension. Ici encore, tout dépend de votre routeur ou votre box Internet.

## **Connexion** à la console

Une fois l'adresse IP connue, il faudra vous y connecter en utilisant un client SSH. J'utilise « Putty », qui est disponible gratuitement à cette adresse : <u>https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html</u> Une fois installé, utilisez l'icône disponible sur le bureau de Windows pour lancer le logiciel :



Sélectionnez le type de connexion **SSH**, saisissez l'adresse IP de la carte Raspberry Pi, laissez le port à 22, donnez un nom à cette connexion dans la case **Saved Sessions**, et cliquez sur **Save**. Vous accéderez ainsi rapidement à la Raspberry Pi lors des connexions suivantes :



On accède ensuite à la console SSH en double-cliquant sur le nom donné à la connexion :



Le système d'exploitation vous invite à vous identifier. Tapez **pi** à la suite de « login as : » et validez avec la touche **Entrée**. Lorsque la ligne « Password » s'affiche, tapez **raspberry**. Rien ne s'affiche à l'écran, mais c'est normal. Validez ensuite avec la touche **Entrée**.



Vous voilà connecté à la console Linux. C'est à partir de ce moment que vous pouvez exécuter toutes les commandes listées en italique dans ce document.

## Extension du système de fichiers

Les systèmes d'exploitation Raspberry OS récents se chargent automatiquement de redimensionner les partitions, afin d'utiliser la totalité de l'espace disponible sur la carte SD. Avant, il fallait utiliser la commande *sudo raspi-config* pour accéder à cette fonction.



## **Test de connexion Internet**

Comme il va falloir installer des fichiers et mises à jour assez rapidement, on s'assurera que la Raspberry Pi peut bien accéder à Internet en exécutant par exemple la commande suivante : *ping 8.8.8* 



Appuyer sur la touche **Ctrl** du clavier, puis en la maintenant appuyée, appuyer sur la touche **C**. Cela permet de mettre fin à une tâche en cours d'exécution.

Cette action sera notée **Ctrl+C** dans la suite de ce document. Il en va de même pour toutes les autres combinaisons de touches, comme **Ctrl+Z** par exemple.

## Navigation dans les répertoires

La commande permettant de changer de répertoire est à quelques détails près identique à celle utilisée sous DOS.

Dans l'exemple ci-dessous, je souhaite entrer dans le répertoire contenant les chemins d'accès aux périphériques de la carte Raspberry Pi :



Il est préférable de garder pour habitude d'utiliser des chemins absolus, c'est à dire qu'on considère toujours qu'on part de la racine de l'arborescence des répertoires. La racine est le niveau le plus bas de l'arborescence du « disque », comme <u>c:</u> sous Windows. L'objectif étant de ne pas se soucier de l'emplacement courant avant le changement de répertoire. Vous pourrez utiliser des chemins relatifs quand vous vous serez familiarisé avec l'environnement Linux. Un chemin relatif permet de n'indiquer que le nom d'un sous-répertoire présent dans le répertoire courant.

En cas de doute concernant le chemin courant, on peut le vérifier à l'aide de la commande pwd.

#### Rev. 04/06/2022 @ 22:29

#### http://blog.shibby.fr



Dans l'exemple ci-dessus, je vérifie que je suis bien allé dans le répertoire /dev avec la commande pwd.

Je me déplace ensuite dans la racine avec *cd* /.

Je me déplace ensuite dans le répertoire de l'utilisateur **pi** (le nom d'utilisateur par défaut sur le système d'exploitation Raspbian) avec la commande *cd* ~. On voit bien que cette commande nous emmène dans **/home/pi** grâce à la commande *pwd*.

Enfin, je me déplace dans le répertoire **/opt/** en utilisant son chemin absolu avec la commande *cd /opt/*, puis dans **MMDVMHost** en utilisant son chemin relatif avec la commande *cd MMDVMHost/*. On voit enfin que la commande définissant le chemin absolu complet nous emmène au même endroit : **cd /opt/MMDVMHost/**.

## Édition de fichiers depuis la console

Il sera nécessaire d'éditer des fichiers de configuration des modules logiciels du MMDVM, ou de créer des fichiers pour l'exécution automatique de ces modules par Linux. La façon la plus simple et la plus sûre d'effectuer ces actions consiste à utiliser un éditeur de texte depuis la console Linux. Ainsi, on ne perd pas le format spécifique des fichiers, notamment les caractères invisibles de sauts de ligne qui sont différents de ceux utilisés par Windows, ce qui évite d'être confronté à des messages d'erreur qui ne sont pas forcément explicites pour les personnes découvrant Linux. L'éditeur dont l'utilisation est la plus intuitive, et qui est présent nativement sur le système d'exploitation de la Raspberry Pi, s'appelle **nano**. L'utilisation de cet éditeur nécessite des droits « super utilisateur », car il permet de

modifier n'importe quel fichier, avec tous les risques de corruption de ceux-ci qui peuvent en découler. Ainsi, on exécutera ce logiciel en le précédant de la commande **sudo**, comme pour d'autres actions tout aussi sensibles décrites plus loin dans ce document. Exemple d'édition d'un fichier nommé « fichier\_de\_test » :



Si ce fichier n'existe pas, il sera créé automatiquement au moment où nous demanderons à **nano** de sauvegarder les modifications.



La sauvegarde est proposée au moment où on quitte le **nano**, avec la combinaison de touches **Ctrl+X**:

الع pi@raspberrypi: ~		
GNU nano 3.2	fichier_de_test	Modified 🔺
J'écris un truc, et si je dois aussi écrire des chiff placées au dessus des lettres sur mon clavier: pas ce	res je dois obligatoirement le faire en utilisant les t lles du pavé numérique!	ouches
J'appuie sur la touche SHIFT du clavier et j'appuie s	ur les chiffres :) 0123456789	
Save modified buffer? (Answering "No" will DISCARD c	hanges.)	
Y Yes N No <sup>^C</sup> Cancel		-

Il suffit enfin d'appuyer sur la touche **Y** pour confirmer la sauvegarde, **N** pour quitter sans sauvegarder, ou **Ctrl+C** pour revenir à l'édition du fichier sans effectuer de sauvegarde.

وع pi@raspberrypi: ~				_ <b>D</b> X
GNU nano 3.2	fichier_de	_test		Modified 🔺
J'écris un truc, et si je dois auss placées au dessus des lettres sur m	i écrire des chiffres je dois on clavier: pas celles du pavé	obligatoirement le faire é numérique!	en utilisant les touc	hes
J'appuie sur la touche SHIFT du cla	vier et j'appuie sur les chiff	fres :) 0123456789		
File Name to Write: fichier_de_test				
^G Get Help M-D I	OS Format M-A A	Append	M-B Backup File	
^C Cancel M-M N	ac Format M-P P	Prepend	^T To Files	-

Ici le nom de fichier est rappelé après avoir appuyé sur **Y**. Si on souhaite garder ce nom de fichier, il suffit d'appuyer sur la touche **Entrée**, sinon on peut le modifier avant d'appuyer sur la touche **Entrée**. On peut aussi annuler cette action à tout moment en appuyant sur **Ctrl+C**.

Vous connaissez maintenant les bases de l'utilisation de la console Linux, nous allons pouvoir passer aux choses sérieuses !

# **Conditions initiales**

Les étapes décrites dans ce document sont basées sur le système d'exploitation **Raspbian 11 Bullseye** « light », qui ne dispose pas d'interface graphique. L'objectif étant de ne pas consommer de ressources matérielles inutilement sur la carte Raspberry Pi. Le texte écrit ci-dessous en italique est correspond aux commandes à taper au clavier sur la console. Il faut ensuite les valider en appuyant sur la touche **Entrée**.

## Version du système d'exploitation

cat /etc/os-release

Pi@raspberrypi: ~	
pi@raspberrypi:~ \$ cat /etc/os-release	<b>^</b>
PRETTY_NAME="Raspbian GNU/Linux 11 (bullseye)"	
NAME="Raspbian GNU/Linux"	
VERSION_ID="11"	
VERSION="11 (bullseye)"	
VERSION_CODENAME=bullseye	
ID=raspbian	
ID_LIKE=debian	
HOME_URL="http://www.raspbian.org/"	
SUPPORT_URL="http://www.raspbian.org/RaspbianForums"	
BUG_REPORT_URL="http://www.raspbian.org/RaspbianBugs"	
pi@raspberrypi:~ \$	-

## Pré-requis

## Mise à jour du système d'exploitation

sudo apt-get update sudo apt-get upgrade -y

## Installation de GIT

sudo apt install git -y

## Installation de l'outil de compilation

sudo apt-get install gcc-arm-none-eabi -y

# Programmation de la carte interface NUCLEO STM32F466 avec le firmware MMDVM

L'ancienne méthode utilisée est déplacée en **Annexe 1** à la fin de ce document, à des fins d'information uniquement, car elle n'est plus applicable aux versions récentes du logiciel MMDVM développé par G4KLX. Il doit être possible d'adapter cette ancienne méthode pour continuer à compiler depuis Windows, mais la nouvelle méthode utilisant la Raspberry Pi est nettement plus simple finalement.

## Chargement des fichiers et compilation depuis la carte Raspberry Pi

La commande à exécuter pour charger les fichiers source est la suivante : sudo git clone https://github.com/G4KLX/MMDVM



Ouvrir le répertoire MMDVM fraîchement créé, avec la commande suivante : *cd MMDVM* 

Charger les librairies qui permettront de compiler le logiciel embarqué de la carte Nucleo STM32F446, avec cette commande :

sudo git clone <u>https://github.com/juribeparada/STM32F4XX\_Lib.git</u>

Revenir à la racide du répertoire MMDVM avec cette commande : *cd /home/pi/MMDVM* 

Éditer ensuite le fichier de configuration avec la commande suivante : *sudo nano Config.h* 



On définit ici les paramètres de fonctionnement de la carte interface du MMDVM. Ils dépendront donc de votre besoin. Pour information, voici le paramétrage que j'utilise sur la carte interface de F5UII :

- /\*
- \* Copyright (C) 2015,2016,2017,2018,2020 by Jonathan Naylor G4KLX
- \*
- \* This program is free software; you can redistribute it and/or modify
- \* it under the terms of the GNU General Public License as published by
- \* the Free Software Foundation; either version 2 of the License, or
- \* (at your option) any later version.
- \*
- \* This program is distributed in the hope that it will be useful,
- \* but WITHOUT ANY WARRANTY; without even the implied warranty of
- \* MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
- \* GNU General Public License for more details.
- \*
- \* You should have received a copy of the GNU General Public License
- \* along with this program; if not, write to the Free Software
- \* Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

\*/

#if !defined(CONFIG\_H)
#define CONFIG\_H

// Allow for the selection of which modes to compile into the firmware. This is particularly useful for processors // which have limited code space and processing power like the STM32F103, which is found on older/cheaper boards.

// Enable D-Star support.

Rev. 04/06/2022 @ 22:29

#define MODE\_DSTAR

// Enable DMR support.
#define MODE\_DMR

// Enable System Fusion support.
#define MODE\_YSF

// Enable P25 phase 1 support.
#define MODE\_P25

// Enable NXDN support, the boxcar filter sometimes improves the performance of NXDN receive on some systems.
#define MODE\_NXDN
#define USE\_NXDN\_BOXCAR

// Enable M17 support.
#define MODE\_M17

// Enable POCSAG support.
#define MODE\_POCSAG

// Enable FM support.
#define MODE\_FM

// Enable AX.25 support, this is only enabled if FM is also enabled.
//#define MODE\_AX25

// Allow for the use of high quality external clock oscillators // The number is the frequency of the oscillator in Hertz. // // The frequency of the TCXO must be an integer multiple of 48000. // Frequencies such as 12.0 Mhz (48000 \* 250) and 14.4 Mhz (48000 \* 300) are suitable. // Frequencies such as 10.0 Mhz (48000 \* 208.333) or 20 Mhz (48000 \* 416.666) are not suitable. // // For 12 MHz #define EXTERNAL\_OSC 12000000 // For 12.288 MHz // #define EXTERNAL\_OSC 12288000 // For 14.4 MHz // #define EXTERNAL\_OSC 14400000 // For 19.2 MHz // #define EXTERNAL\_OSC 1920000

// Select a baud rate for host communication. The faster speeds are needed for external FM to work. #define SERIAL\_SPEED 115200 // Suitable for most older boards (Arduino Due, STM32F1\_POG, etc). External FM will NOT work with this! // #define SERIAL\_SPEED 230400 // Only works on newer boards like fast M4, M7, Teensy 3.x. External FM might work with this // #define SERIAL\_SPEED 460800 // Only works on newer boards like fast M4, M7, Teensy 3.x. External FM should work with this

// Use pins to output the current mode via LEDs
#define MODE\_LEDS

// For the original Arduino Due pin layout
// #define ARDUINO\_DUE\_PAPA

#if defined(STM32F1)
// For the SQ6POG board
#define STM32F1\_POG
#else
// For the ZUM V1.0 and V1.0.1 boards pin layout
// #define ARDUINO\_DUE\_ZUM\_V10
#endif

// For the SP8NTH board
// #define ARDUINO\_DUE\_NTH

// For ST Nucleo-64 STM32F446RE board
#define STM32F4\_NUCLEO\_MORPHO\_HEADER
// #define STM32F4\_NUCLEO\_ARDUINO\_HEADER

// Use separate mode pins to switch external channel/filters/bandwidth for example
// #define MODE\_PINS

// For the VK6MST Pi3 Shield communicating over i2c. i2c address & speed defined in i2cTeensy.cpp // #define VK6MST\_TEENSY\_PI3\_SHIELD\_I2C

// Pass RSSI information to the host
#define SEND\_RSSI\_DATA

// Use the modem as a serial repeater for Nextion displays
#define SERIAL\_REPEATER

// Use the modem as an I2C repeater for OLED displays
// #define I2C\_REPEATER

// To reduce CPU load, you can remove the DC blocker by commenting out the next line #define USE\_DCBLOCKER

// Constant Service LED once repeater is running
// Do not use if employing an external hardware watchdog
// #define CONSTANT\_SRV\_LED

// Use the YSF and P25 LEDs for NXDN
#define USE\_ALTERNATE\_NXDN\_LEDS

// Use the D-Star and P25 LEDs for M17
#define USE\_ALTERNATE\_M17\_LEDS

// Use the D-Star and DMR LEDs for POCSAG #define USE\_ALTERNATE\_POCSAG\_LEDS

// Use the D-Star and YSF LEDs for FM
#define USE\_ALTERNATE\_FM\_LEDS

#if defined(STM32F1\_POG)
// Slower boards need to run their serial at 115200 baud
#undef SERIAL\_SPEED
#define SERIAL\_SPEED 115200
#endif

### #endif

Une fois le fichier édité, quitter **nano** en sauvegardant les modifications apportées, avec **ctrl+X** suivi d'**Entrée**.

Compiler ensuite le fichier binaire pour la carte **Nucleo STM32F446** en exécutant la commande suivante : *sudo make nucleo* 

Les étapes de compilation défilent à l'écran, jusqu'à la création du fichier binaire qui sera injecté dans la carte Nucleo STM32F446.

😰 pi@raspberrypi: ~/MMDVM	
xx.c -o obj f4/startup stm32f4xx.o	A
Compiled STM32F4XX_Lib/Device/startup/startup_stm32f4xx.c!	
arm-none-eabi-g++ obj_f4/AX25Demodulator.o obj_f4/AX25Frame.o obj_f4/AX25RX.o obj_f4/AX25Twist.o obj_f4/AX25TX.o obj_f4/CalDMR.o obj_f4/CalDMR.o obj_f4/CalDMR.o obj_f4/CalDMR.o obj_f4/CalDMR.o obj_f4/CalDMR.o obj_f4/CalDMR.o obj_f4/CalDMR.o obj_f4/DMRDMORX.o obj_f4/DMRDMOTX.o obj_f4/DMRIdleRX.o obj_f4/DMRRX.o obj_f4/DMRSIotRX.o obj_f4 o obj_f4/FMCTCSSRX.o obj_f4/FMCTCSSTX.o obj_f4/FMDownSampler.o obj_f4/FMKeyer.o obj_f4/FMNoiseSquelch.o obj_f4/FMTimeout.o obj_f4/FMTimer.o ob sy.o obj_f4/M17RX.o obj_f4/M17TX.o obj_f4/MMDVM.o obj_f4/NXDNRX.o obj_f4/NXDNTX.o obj_f4/P25RX.o obj_f4/P25TX.o obj_f4/POCSAGTX.o obj_f4/Seria o obj_f4/STMUART.o obj_f4/Utils.o obj_f4/YSFRX.o obj_f4/YSFTX.o obj_f4/misc.o obj_f4/stm32f4xx_adc.o obj_f4/stm32f4xx_dac.o obj_f4/stm3	arRX.o obj_f4/CalDStar 4/DMRSlotType.o obj_f4 j_f4/FMUpSampler.o obj lArduino.o obj_f4/Seri pio.o obj_f4/stm32f4xx -mfpu=fpv4-sp-d16 -mfl
arm-none-eabi-size bin/mmdvm f4.elf	
text data bss dec hex filename	
299408 1656 71388 372452 5aee4 bin/mmdvm_f4.elf	
arm-none-eabi-objcopy -O ihex bin/mmdvm_f4.elf bin/mmdvm_f4.hex	
Objcopy from ELF to IHEX complete!	
arm-none-eabi-objcopy -O binary bin/mmdvm f4.elf bin/mmdvm f4.bin	
Objcopy from ELF to BINARY complete!	
pi@raspberrypi:~/MMDVM \$	-

Il ne reste plus qu'à charger un de ces fichiers dans la carte **Nucleo STM32F446**. La carte **Nucleo STM32F446** peut maintenant être connectée sur un des ports USB de la carte **Raspberry Pi** :



Cette opération nécessite de créer un lien logiciel vers l'interface de disque USB donnant accès à une partie de la mémoire de cette carte, afin d'y déposer un des fichiers qui viennent d'être générés. Le logiciel déjà présent dans cette carte se chargera de transférer ce fichier au bon endroit dans le microcontrôleur. Ce n'est pas plus compliqué que ça !

Vérifier tout d'abord que le lien USB est fonctionnel, en utilisant cette commande : sudo fdisk -l

Parmi les informations affichées, vous devez voir celles-ci :



Notez bien le chemin d'accès au disque, car il va nous servir juste après à créer le point de montage. Ici, il est écrit **/dev/sdb**. Le chemin n'est pas forcément celui visible ci-dessus, donc soyez bien attentifs.

Créer le point de montage avec la commande suivante : *sudo mkdir /mnt/usb* 

Monter le disque USB : sudo mount /dev/sdb /mnt/usb

Copier l'un des fichiers générés à l'issue de la compilation dans ce disque USB :

Juste après cette dernière commande, la LED la plus proche du connecteur USB de la carte **Nucleo STM32F446** se met à clignoter en rouge et vert, puis la carte redémarre automatiquement une fois le chargement du fichier terminé.

On peut d'ailleurs constater que le fichier précédemment chargé sur le disque USB de la carte **Nucleo STM32F446** a été supprimé automatiquement après avoir été chargé dans le microcontrôleur, en vérifiant le contenu du répertoire où il avait été déposé : *ls /mnt/usb/* 



C'est fini pour cette partie du MMDVM. Vous pouvez enficher la carte d'interface radio sur la carte **Nucleo STM32F446**:



# Chargement et installation des logiciels sur la carte Raspberry Pi

Je laisse la description ci-dessous à titre d'information, car au moment où j'édite ce document, il n'est plus possible d'utiliser python2 et python-serial sur le système d'exploitation Raspbian bullseye, et le programme nextion.py fourni par G4KLX n'est pas encore compatible avec python3. Les étapes restent applicables sur un ordinateur qui autorise encore l'utilisation de python2.

## **Pré-requis**

## Installation du module de communication par le port série pour Python

sudo apt-get install python-serial

## Installation de MMDVMHost

cd /opt sudo git clone https://github.com/G4KLX/MMDVMHost cd /opt/MMDVMHost/ sudo make

## Chargement des fichiers d'affichage dans l'écran Nextion

Même si vous n'effectuez qu'une mise à jour d'un MMDVM déjà en service, cette action peut rester nécessaire car le dialogue avec l'écran est susceptible d'évoluer lui aussi.

Les procédures décrites habituellement utilisent le port RS232 déjà présent sur la carte Raspberry Pi, mais cette action est devenue compliquée depuis que cette voie de communication est utilisée par défaut par l'interface WiFi / Bluetooth. Je préfère utiliser une carte interface USB vers RS232 qui est reconnue automatiquement par le système d'exploitation de la carte Raspberry Pi, et qui évite d'éditer un fichier de configuration sensible sur la carte **Raspberry Pi**.

On trouve très facilement ce type de carte sur les sites de vente en ligne, avec les mots clé **USB FT232 TTL**, pour moins de 2€. Cette carte est donc connectée par son port USB sur la carte Raspberry Pi, et on relie les contacts de l'afficheur sur les broches mâles.

Le brochage est le suivant : Fil noir = GND (masse) Fil rouge = +5V (alimentation électrique de l'écran) Fil jaune = RX de l'écran, à relier sur le TX du module USB RS232 Fil bleu = TX de l'écran, à relier sur le RX du module USB RS232

On prendra soin de sélectionner au préalable une tension de fonctionnement de 5V sur la carte interface USB RS232, si celle-ci n'est pas câblée nativement pour être connectée à des signaux UART à la norme TTL.






Côté Raspberry Pi, on doit voir le nom **ttyUSB0** dans la liste des périphériques :

cd /dev ls tty\*

🛃 pi@r	aspberrypi	: /dev												
niĝras	nberryr	i:/dev	\$ pwd											^
/dev	-pactage	21, 40,	+ pau											
pi@ras	pberryp	i:/dev	\$ 1s tt	¥¥										
tty tty0 tty1 tty10 tty11	tty12 tty13 tty14 tty15 tty16	tty17 tty18 tty19 tty2 tty20	tty21 tty22 tty23 tty24 tty25	tty26 tty27 tty28 tty29 tty3	tty30 tty31 tty32 tty33 tty34	tty35 tty36 tty37 tty38 tty39	tty4 tty40 tty41 tty42 tty43	tty44 tty45 tty46 tty47 tty48	tty49 tty5 tty50 tty51 tty52	tty53 tty54 tty55 tty56 tty57	tty58 tty59 tty6 tty60 tty61	tty62 tty63 tty7 tty8 tty9	ttyAMA0 ttyprintk <mark>ttyUSB0</mark>	
pi@ras	pberryp	pi:/dev	Ş											

Avant de charger le fichier prêt à l'emploi dans l'écran, il faut noter la référence de celui-ci, indiquée sur sa carte électronique :



NX3224T024



On peut lister toutes les références compatibles en se rendant dans le répertoire contenant les fichiers :

cd /opt/MMDVMHost/Nextion\_G4KLX/

🗗 pi@raspberrypi: /opt/MMDVMHost/Nextion\_G4KLX



On choisit alors le fichier portant la référence de l'écran.

En prenant comme exemple l'écran portant la référence NX4832T035, le chargement s'effectue avec la commande suivante : sudo python /opt/MMDVMHost/Nextion\_G4KLX/nextion.py /opt/MMDVMHost/Nextion\_G4KLX/NX4832T035.tft /dev/ttyUSB0

On peut observer la progression du chargement sur la console Linux et sur l'écran Nextion :



L'écran redémarre sur le logo MMDVM à la fin du chargement.

On peut alors déconnecter l'écran de sur le convertisseur USB RS232, et le relier à la carte interface radio sur le connecteur dédié:

Sur la carte interface de F5UII, le brochage est le suivant :

Fil noir = GND Fil rouge = 5V Fil jaune = RX Fil bleu = TX



# Installation de MMDVMCal

cd /opt sudo git clone https://github.com/G4KLX/MMDVMCal cd /opt/MMDVMCal/ sudo make

# Installation de DMRGateway (seulement si l'accès à un serveur d'interconnexion de relais phonie est souhaité) :

cd /opt sudo git clone https://github.com/G4KLX/DMRGateway cd /opt/DMRGateway/ sudo make

# **Installation de DAPNETGateway** (seulement si l'accès à un serveur d'interconnexion de relais POCSAG est souhaité) :

cd /opt sudo git clone https://github.com/G4KLX/DAPNETGateway cd /opt/DAPNETGateway/ sudo make

# Installation de MMDVM-Dashboard

sudo apt-get install lighttpd -y sudo groupadd www-data sudo usermod -G www-data -a pi

sudo chown -R www-data:www-data /var/www/html sudo chmod -R 775 /var/www/html sudo apt-get install php7.4-common php7.4-cgi php -y sudo lighty-enable-mod fastcgi sudo lighty-enable-mod fastcgi-php sudo service lighttpd start sudo service lighttpd force-reload cd ~ sudo git clone https://github.com/dg9vh/MMDVMHost-Dashboard.git sudo cp -R /home/pi/MMDVMHost-Dashboard/\* /var/www/html/

À présent vous devez pouvoir vous connecter à l'interface web du tableau de bord MMDVM sur un explorateur internet, en tapant son IP dans la barre d'adresse :

### MMDVM-Dashboard by DG9VH for Hotspot:

### DMR-Id:

MMDVMHost by G4KLX Version: Firmware:							
You forgot to remove setup.php in root-directo	ory of your dashboard or you forgot to configure it! Plea	e delete the file or configure your Dashboard	by calling setup.php!				
Currently TXing							
Time (UTC)	Mode	Callsign		Target	Source	TX-Time	
System Info							
CPU-Temperature	CPU-Frequency	System-Load		CPU-Usage	Uptime	Idle	
34.324 °C	1400 MHz	0.77 / 0.29 / 0.18		46.1 %	12 mins, 33 s	11 mins, 37 s	s
Disk Use							
File System	Mount Point		Use	Free	Used	Total	
/dev/root	/		213	5.38 GB	1.41 GB	7.11 GB	
/dev/mmcblk0p1	/boot		198	204.41 MB	47.64 MB	252.05 MB	
Repeater Info							
Current Mode							
idle							
Location		TX-Freq.			RX-Freq.		
		. MHz			. MHz		
Enabled Modes							
DMR DMR Network	D-Star D-Star Network	System Fusion System F	usion Network	P25 P25 Network	NXDN NXDN Network	POCSAG	POCSAG Network
Last Heard List of today's 0 callsigns. Show 10 v entries							Search:
Time (UTC)	▼ Mode	¢ ¢	Target	\$ Source	Dur (s)	4 Loss	\$ BER \$
			No data ava	ilable in table			
Showing 0 to 0 of 0 entries							Previous Next
Today's local transmissions Show 10 V entries							Search:
Time (UTC)	🚽 Mode	Target	Source	e (s)	Loss	ER   RSSI (av	zg) ¢
			No data ave	ilable in table			

MMDVMHost-Dashboard V 20201031-1 (GitlD unknown) | Last Reload 2021-04-16, 19:29:02 (UTC) | stop refreshing | get your own at: https://github.com/dg9vh/MMDVMHost-Dashboard | Follow me @DG9VH | Credits

### Rev. 04/06/2022 @ 22:29

# Paramétrage du tableau de bord MMDVM-Dashboard

AJOUTER ICI LA PROCÉDURE

# Test rapide de dialogue entre la Raspberry Pi, l'interface MMDVM et l'écran Nextion

Tous les logiciels sont installés, et les cartes sont reliées entre elles. Nous allons donc pouvoir faire un premier test de bon fonctionnement de l'ensemble des cartes constituant le MMDVM, c'est toujours rassurant de voir que le temps passé à exécuter toutes les actions précédentes n'a pas été perdu.

Dans un premier temps, seuls les paramètres définissant les interfaces de communication entre chaque carte sont nécessaires.

Éditer le fichier MMDVM.ini avec la commande suivante : sudo nano /opt/MMDVMHost/MMDVM.ini

Chaque fonction a ses paramètres regroupés dans un même bloc, identifié par le texte compris entre les crochets [ et ]. Afin de vous aider à trouver facilement les paramètres à modifier, vous trouverez ci-dessous dans la colonne de gauche le nom du bloc concerné par la modification, et à sa droite le paramètre à modifier, avec les modifications apportées surlignées en vert :

[General]	Display=N <mark>extion</mark>
[Modem]	UARTPort=/dev/ttyA <mark>CM</mark> 0
[Modem]	UARTSpeed=115200
[Nextion] Ici on va enlever le signe # au début de la ligne suivante :	Port=modem
[Nextion] Ici on va ajouter le signe # au début de la ligne suivante :	<mark>#</mark> Port=/dev/ttyAMA0
[Nextion]	ScreenLayout= <mark>0</mark>

Quitter enfin l'édition en sauvegardant les modifications apportées : **ctrl+x** puis **Entrée**.

Rev. 04/06/2022 @ 22:29

#### http://blog.shibby.fr

Une fois les paramètres de fonctionnement définis, on peut tester rapidement le bon fonctionnement du MMDVM en exécutant le programme principal avec la commande suivante :

sudo /opt/MMDVMHost/MMDVMHost /opt/MMDVMHost/MMDVM.ini

Si l'exécution se termine par ce message, vérifier que le MMDVM est bien connecté à la Raspberry Pi : E: 2022-06-04 19:25:03.626 Cannot open device - /dev/ttyACM0

Si la liaison entre la Raspberry Pi et l'interface MMDVM fonctionne, on doit voir ce type de message :

I: 2022-06-04 19:28:29.291 This software is for use on amateur radio networks only,

I: 2022-06-04 19:28:29.292 it is to be used for educational purposes only. Its use on

I: 2022-06-04 19:28:29.292 commercial networks is strictly prohibited.

I: 2022-06-04 19:28:29.292 Copyright(C) 2015-2021 by Jonathan Naylor, G4KLX and others

M: 2022-06-04 19:28:29.292 MMDVMHost-20220523 is starting

M: 2022-06-04 19:28:29.292 Built 16:40:46 Jun 4 2022 (GitID #fe195c4)

Les informations indiquées ensuite n'ont aucun intérêt pour ce premier test, je les ai donc supprimées pour une meilleure lisibilité
 M: 2022-06-04 19:28:32.344 MMDVMHost-20220523 is running

Cette dernière ligne nous confirme que le MMDVM est en cours d'exécution.

La communication avec l'écran Nextion est elle aussi validée car celui-ci affiche des informations liées au MMDVM à présent :

Rev. 04/06/2022 @ 22:29



On peut quitter l'application avec **Ctrl+Z**.

Le message suivant apparaît alors à l'écran:

^Z
[2]+ Stopped sudo /opt/MMDVMHost/MMDVMHost /opt/MMDVMHost/MMDVM.ini
pi@raspberrypi:/opt \$

On voit la commande qui nous a permis de quitter le logiciel « <sup>A</sup>Z », le nom du logiciel qui était en cours d'exécution, et le retour à l'invite de commande sur la dernière ligne.

# Configuration de test du relais MMDVM



Chaque poste est connecté à l'interface MMDVM par l'intermédiaire de sa prise « accessoires » en face arrière, jusqu'à la prise mini-DIN 6 broches de l'interface avec la carte **Nucleo STM32F446**. Les entrées/sorties audio des postes sont récupérées **avant filtrage en interne**.

Rev. 04/06/2022 @ 22:29



Il faut noter ici que la connexion sur l'interface MMDVM des équipements d'émission/réception constituant le relais est **obligatoire** pour les étapes de calibration.

### Brochage des connecteurs Mini-Din 6 points et liaisons externes



\* Seulement valable les broches 1 et 2 de JP1 sont reliées ensemble, et si les broches 3 et 4 sont reliées ensemble.

# Calibration du MMDVM avec MMDVMCal

L'étape de calibration permet de régler les niveaux de réception et d'émission sur la carte interface analogique, qui est chargée d'adapter les signaux issus des équipements radio avant de les appliquer à l'entrée et à la sortie de la carte processeur STM32.

Cette étape est cruciale et doit être réalisée avec le plus grand soin, car elle assure le bon fonctionnement du MMDVM en réception et en émission. Des signaux audio trop forts ou trop faibles impliqueraient un taux d'erreur important, dégradant significativement la qualité des transmissions, allant jusqu'à rendre impossible le décodage des émissions, aussi bien du côté du relais que de celui des équipements radio des opérateurs.

Cette opération est grandement simplifiée grâce aux informations fournies par MMDVMCal d'une part, et à celles obtenues avec une simple clé SDR.

Pour que MMDVMCal puisse être exécuté, il faut que le MMDVM ne soit pas déjà sollicité par un autre logiciel, comme MMDVMHost par exemple. Il faut donc quitter si besoin MMDVMHost avant d'exécuter MMDVMCal.

On ajustera le niveau audio du signal reçu à l'aide de la résistance variable multi-tours R99 (à gauche), et celui du signal émis par l'intermédiaire de la résistance variable multi-tours R98 (à droite) :



Commande permettant d'exécuter MMDVMCal : sudo /opt/MMDVMCal/MMDVMCal /dev/ttyACM0

### Exemple de calibration pour un émetteur DAPNET

Pi@raspberrypi: /opt/MMDVMCal				
DiGraspberrypi:/opt/MMDVMCal \$ sudo /opt/MMDVMCal/MMDVMCal /dev/ttyACM0				
Version: 1,	description: MMDVM 20210101 (D-Star/DMR/System Fusion/F25/NXDN/FOCSAG/FM) 12.0000 MHz GitID #0000000			
The commands	are:			
H/h	Display help			
Q/q	Quit			
W/w	Enable/disable modem debug messages			
I	Toggle transmit inversion			
i	Toggle receive inversion			
0	Increase TX DC offset level			
0	Decrease TX DC offset level			
С	Increase RX DC offset level			
с	Decrease RX DC offset level			
P/p	Toggle PTT inversion			
R	Increase receive level			
r	Decrease receive level			
Т	Increase transmit level			
t	Decrease transmit level			
d	D-Star Mode			
F	FM Deviation Mode (Adjust for correct Deviation)			
D	DMR Deviation Mode (Adjust for 2.75Khz Deviation)			
L/1	DMR Low Frequency Mode (80 Hz square wave)			
A	DMR Duplex 1031 Hz Test Pattern (TS2 CC1 ID1 TG9)			
M/m	DMR Simplex 1031 Hz Test Pattern (CC1 ID1 TG9)			
a	P25 1011 Hz Test Pattern (NAC293 ID1 TG1)			
N	NXDN 1031 Hz Test Pattern (RAN1 ID1 TG1)			
K/k	BER Test Mode (FEC) for D-Star			
b	BER Test Mode (FEC) for DMR Simplex (CC1)			
В	BER Test Mode (1031 Hz Test Pattern) for DMR Simplex (CC1 ID1 TG9)			
J	BER Test Mode (FEC) for YSF			
j	BER Test Mode (FEC) for P25			
n	BER Test Mode (FEC) for NXDN			
g	POCSAG 600Hz Test Pattern			
S/s	RSSI Mode			
V/v	Display version of MMDVMCal			
<space></space>	Toggle transmit			
		-		

Notez bien que la casse (majuscule/minuscule) est à respecter car certaines lettres ont deux fonctions différentes !

Le logiciel démarre en indiquant toutes les commandes disponibles. On pourra par la suite lui demander de les afficher à nouveau en appuyant simplement sur la touche **h** du clavier.

Dans cet exemple, nous souhaitons ajuster l'excursion de fréquence d'un émetteur DAPNET, utilisant le protocole radio **POCSAG** Il faut donc appuyer sur la touche **g** du clavier pour entrer dans le mode de test **POCSAG**, puis ensuite piloter le signal **PTT** de l'émetteur en appuyant sur la **barre d'espace** du clavier. Chaque appui sur celle-ci alternera le mode de fonctionnement de l'émetteur en émission ou en réception :

🧬 pi@raspberryp	i: /opt/MMDVMCal				
n	BER Test Mode (FEC) for NXDN	A			
g	POCSAG 600Hz Test Pattern				
S/s	RSSI Mode				
V/v	Display version of MMDVMCal				
<space></space>	Toggle transmit				
POCSAG 600 H	POCSAG 600 Hz Test Pattern				
Set transmitter ON					
Set transmit	Set transmitter OFF				
Set transmitter ON					
Set transmitter OFF					
		-			

On doit entendre la tonalité de 600Hz sur un récepteur calé sur la même fréquence de l'émetteur. Si on n'entend qu'une porteuse, il faut vérifier que le signal nommé « AUDIOTX » sur la carte interface est bien relié sur l'entrée audio du poste émetteur, et le cas échéant ajuster R98 pour entendre le signal audio émis par le MMDVM.

Une fois cette vérification effectuée, on peut procéder au réglage de l'excursion de fréquence en s'aidant d'un récepteur SDR (même un modèle premier prix) et d'un logiciel de décodage quelconque.

Dans l'exemple suivant, j'utilise une de ces clés « DVB-T+FM+DAB » montées dans un boîtier plastique bleu, et le logiciel **SDR#**.

### Réglage audio au plus bas, on ne voit qu'une porteuse. On notera le réglage « Bandwidth » à 12500 Hz :



J'augmente petit à petit le niveau audio de la carte interface. Les raies du signal modulé apparaissent et celle de la porteuse est toujours bien visible au centre. Elle est placée pile sur la ligne verticale rouge en haut de l'écran, et sur la ligne verticale bleu clair en bas :



L'objectif est maintenant de continuer à augmenter le niveau audio jusqu'à ce que la raie centrale disparaisse. Ce réglage est très précis, et on s'aidera du spectre audio en bas de l'écran pour s'assurer qu'on n'a pas dépassé le point de réglage :



Si on dépasse le point de réglage, on constate que l'émission dépasse de la bande passante allouée, ce qu'il faut à tout prix éviter car cela rend l'émission inexploitable par les opérateurs qui utiliseront votre relais MMDVM :



# Cas d'un relais Multi Mode

Lorsqu'on utilise le MMDVM en tant qu'émetteur DAPNET, il est très fortement déconseillé de l'utiliser en même temps dans d'autres modes. La raison est simple : les messages texte émis sur le DAPNET ne sont pas mémorisés en vue d'être émis lorsque c'est possible. Une émission DAPNET a donc de très fortes chances d'être perdue si le relais MMDVM est en cours d'utilisation dans un autre mode au même moment.

Une autre raison imposant l'incompatibilité du DAPNET avec d'autres modes sur le même relais est l'utilisation d'une fréquence d'émission unique pour les réseau DAPNET, dont les différents relais se partagent l'utilisation selon une fenêtre temporelle définie. Un même message est donc émis avec un léger décalage dans le temps entre des relais proches, afin d'assurer l'émission d'un seul relais à la fois et ainsi la bonne réception du signal. Ce décalage de transmission du message ne doit pas être confondu avec une mise en mémoire des messages. Il s'agit juste d'une distribution de ceux-ci étalée dans le temps par le serveur d'interconnexion au réseau.

Parlons à présent des autres modes gérés par le MMDVM. Même s'ils ne peuvent pas être utilisés simultanément lors d'une discussion, puisque le MMDVM n'effectue pas de conversion d'un mode vers un autre, il est tout à fait possible d'effectuer une conversation dans un mode, puis une autre dans un autre mode. Il faudra donc s'assurer que chacun des modes de transmission sera correctement retransmis par le MMDVM.

Ceci est défini d'une façon semblable au réglage d'excursion décrit précédemment pour le DAPNET, mais il faut aussi s'assurer que le réglage est valable pour tous les modes de transmission qui seront garés par le MMDVM. On utilisera alors une approche légèrement différente, qui consiste à définir un niveau d'émission fixe avec R98, et ensuite définir un niveau audio spécifique pour chaque mode et permettant d'aboutir au réglage d'excursion idéal pour chacun d'entre eux.

### Réglage de l'excursion en émission

Pour cet exemple, je commence par le réglage de l'excursion en mode DMR, en laissant le niveau audio au réglage logiciel médian (50%). Le mode de test DMR est sélectionné en appuyant sur la touche **D** du clavier. Un appui rapide sur **T** puis **t** permet de vérifier que le niveau audio défini de façon logicielle est bien à 50 %. On active ensuite l'émission avec la **barre d'espace** et on règle le niveau de signal audio transmis avec R98, jusqu'à la disparition de la raie centrale sur les signaux affichés par le logiciel SDR.

🧬 pi@raspberryp	i: /opt/MMDVMCal			
pi@raspberrv	pi:/opt/MMDVMCal \$			
pi@raspberry	pi:/opt/MMDVMCal \$ sudo /opt/MMDVMCal/MMDVMCal /dev/ttyACM0			
Version: 1, (	description: MMDVM 20210101 (D-Star/DMR/System Fusion/P25/NXDN/POCSAG/FM) 12.0000 MHz GitID #0000000			
The commands	are:			
H/h	Display help			
Q/q	Quit			
W/w	Enable/disable modem debug messages			
I	Toggle transmit inversion			
i	Toggle receive inversion			
0	Increase TX DC offset level			
0	Decrease TX DC offset level			
С	Increase RX DC offset level			
С	Decrease RX DC offset level			
P/p	Toggle PTT inversion			
R	Increase receive level			
r	Decrease receive level			
Т	Increase transmit level			
t	Decrease transmit level			
d	D-Star Mode			
F	FM Deviation Mode (Adjust for correct Deviation)			
D	DMR Deviation Mode (Adjust for 2.75Khz Deviation)			
L/1	DMR Low Frequency Mode (80 Hz square wave)			
A	DMR Duplex 1031 Hz Test Pattern (TS2 CC1 ID1 TG9)			
M/m	DMR Simplex 1031 Hz Test Pattern (CC1 ID1 TG9)			
a	P25 1011 Hz Test Pattern (NAC293 ID1 TG1)			
N	NXDN 1031 Hz Test Pattern (RAN1 ID1 TG1)			
K/k	BER Test Mode (FEC) for D-Star			
b	BER Test Mode (FEC) for DMR Simplex (CC1)			
В	BER Test Mode (1031 Hz Test Pattern) for DMR Simplex (CC1 ID1 TG9)			
J	BER Test Mode (FEC) for YSF			
j	BER Test Mode (FEC) for P25			
n	BER Test Mode (FEC) for NXDN			
g	POCSAG 600Hz Test Pattern			
S/s	RSSI Mode			
V/V	Display version of MMDVMCal			
<space></space>	Toggle transmit			
TX Level: 50				
TX Level: 50				
DMR Deviation Mode (Set to 2.75Khz Deviation)				
Set transmit		-		
Set transmit				
		1		

### J'obtiens le signal suivant en réglant R89 :



Rev. 04/06/2022 @ 22:29

Maintenant, je passe dans le mode FM par exemple en appuyant sur **F**, je force l'émission de la tonalité de test avec la **barre d'espace** et je règle à présent le niveau audio d'émission de façon logicielle, avec les touches **T** pour augmenter ou **t** pour diminuer :

pi@raspberrypi: /opt/MMDVMCal	
V/v Display version of MMDVMCal	<u>^</u>
<space> Toggle transmit</space>	
TX Level: 50.5%	
TX Level: 50.0%	
DMR Deviation Mode (Set to 2.75Khz Deviation)	
Set transmitter ON	
Set transmitter OFF	
FM 10Khz channel spacing with 2.30Khz Deviation (956hz)	
Set transmitter ON	
TX Level: 49.5%	
TX Level: 49.0%	
TX Level: 48.5%	
TX Level: 48.0%	
TX Level: 47.5%	
TX Level: 47.0%	
TX Level: 46.5%	
TX Level: 46.0%	
TX Level: 45.5%	
TX Level: 45.0%	
TX Level: 44.5%	
TX Level: 44.0%	
TX Level: 43.5%	
TX Level: 43.0%	
TX Level: 42.5%	
TX Level: 43.0%	
Set transmitter OFF	
	· ·

Une fois le réglage effectué, j'appuie à nouveau sur la **barre d'espace** pour cesser l'émission.

### Le résultat obtenu est le suivant :



Il n'y a pas d'autre mode de calibration en émission, donc on note les niveaux audio logiciels, et on ne touche plus jamais à R98, sauf pendant les campagnes d'entretien et de maintenance du relais :

DMR = 50 %

FM = 43 %

On reportera ces valeurs dans le fichier MMDVM.ini plus tard dans ce document.

### Réglage du taux d'erreur binaire en réception

Comme son nom l'indique, il s'agit d'une valeur indiquant un pourcentage d'erreur de décodage des données reçues. L'acronyme couramment utilisé est **BER** : **B**it **E**rror **R**ate. L'objectif est donc d'obtenir ici une valeur stable et nulle.

En plus du niveau audio du signal reçu s'ajoutent des contraintes de stabilité de la base de temps fournie par l'oscillateur câblé sur la carte interface analogique. Comme indiqué dans le fichier **Config.h** modifié lors de la compilation du logiciel du MMDVM STM32F466 effectuée précédemment dans ce document, j'utilise un oscillateur fournissant au STM32 une horloge de 12MHz. Ce signal lui sert à gérer les fenêtres temporelles des signaux numériques, dont celle des « Time Slots » en DMR. Pour rappel, un Time Slot est une portion de temps pendant laquelle une émission va avoir lieu, ce qui permet de diffuser plusieurs émissions simultanément sur une même fréquence. La moindre dérive de la base de temps créée par l'horloge 12MHz rendrait le MMDVM incapable de savoir quand traiter le Time Slot 1, et le Time Slot 2 en DMR.

C'est pour cette raison qu'il faut impérativement utiliser une source d'horloge la plus stable possible, ce qui exclut l'utilisation d'un simple oscillateur. L'utilisation d'un TCXO (**T**emperature **C**ontrolled **X**tal **O**scillator / Oscilletaur à quartz compensé en température) assurera ainsi l'absence d'erreur causée par une dérive de fréquence.

Cette contrainte technique ne concerne que les relais utilisés en half-duplex. En DMR simplex (emission et réception sur la même fréquence), il n'y a pas de notion de Time Slot. L'utilisation d'un TCXO ne sera pas nécessaire dans ce cas précis.

Si votre interface est utilisée en half-duplex et n'est pas équipée d'un TCXO, attendez-vous donc à rencontrer des problèmes de stabilité de transmission en DMR.

Pour effectuer le réglage en réception, il faut avant tout paramétrer le poste DMR en simplex. Pour ce faire, il suffit de paramétrer un canal avec la même fréquence de réception et d'émission, identique à celle paramétrée sur le récepteur de votre relais MMDVM. Ainsi, le poste DMR utilisé pour les tests n'utilisera pas de Time Slots, et MMDVMCal pourra décoder correctement le signal émis.

On entre dans ce mode de test en appuyant sur la touche **b** :

A DMR Duplex 1031 Hz Test Pattern (TS2 CC1 ID1 TG9)	^
M/m DMR Simplex 1031 Hz Test Pattern (CC1 ID1 TG9)	
a P25 1011 Hz Test Pattern (NAC293 ID1 TG1)	
N NXDN 1031 Hz Test Pattern (RAN1 ID1 TG1)	
K/k BER Test Mode (FEC) for D-Star	
b BER Test Mode (FEC) for DMR Simplex (CC1)	
B BER Test Mode (1031 Hz Test Pattern) for DMR Simplex (CC1 ID1 TG9)	
J BER Test Mode (FEC) for YSF	
j BER Test Mode (FEC) for P25	
n BER Test Mode (FEC) for NXDN	
g POCSAG 600Hz Test Pattern	
S/s RSSI Mode	
V/v Display version of MMDVMCal	
<space> Toggle transmit</space>	
BER Test Mode (FEC) for DMR Simplex	
	-

Il faut ensuite appuyer sur le bouton PTT du poste DMR et vérifier la valeur de BER indiquée à la fin de chaque ligne qui s'affiche à l'écran. Durant mes essais, j'ai constaté que l'émission de mon poste DMR (TYT MD380) avec son antenne d'origine et à faible puissance a tout de même tendance à perturber les signaux analogiques et/ou les amplis de l'interface du MMDVM. Pensez à éloigner autant que possible votre poste du MMDVM lors des essais.

Si vous ne voyez les indications de BER s'afficher uniquement lorsque vous relâchez le PTT, cela signifie que vous devez inverser le mode de réception du MMDVM :

🛃 pi	berrypi: ~	I X
DMR	o seq. 2, FEC BER % (errs): 6.383% (9/141)	×
DMR	o seq. 3, FEC BER % (errs): 6.383% (9/141)	
DMR	o seq. 4, FEC BER % (errs): 6.383% (9/141)	
DMR	o seq. 5, FEC BER % (errs): 6.383% (9/141)	
DMR	o seq. 0, FEC BER % (errs): 6.383% (9/141)	
DMR	o seq. 1, FEC BER % (errs): 6.383% (9/141)	
DMR	o seq. 2, FEC BER % (errs): 6.383% (9/141)	
DMR	o seq. 3, FEC BER % (errs): 6.383% (9/141)	
DMR	o seq. 4, FEC BER % (errs): 6.383% (9/141)	
DMR	o seq. 5, FEC BER % (errs): 6.383% (9/141)	
DMR	o seq. 0, FEC BER % (errs): 6.383% (9/141)	
Tran	sion lost, total frames: 28, bits: 3948, errors: 436, BER: 11.04357%	
		+
DMR DMR DMR DMR DMR DMR DMR Tran	<pre>0 Seq. 4, FEC BER % (errs): 6.383% (9/141) 0 seq. 0, FEC BER % (errs): 6.383% (9/141) 0 seq. 1, FEC BER % (errs): 6.383% (9/141) 0 seq. 2, FEC BER % (errs): 6.383% (9/141) 0 seq. 3, FEC BER % (errs): 6.383% (9/141) 0 seq. 4, FEC BER % (errs): 6.383% (9/141) 0 seq. 5, FEC BER % (errs): 6.383% (9/141) 0 seq. 0, FEC BER % (errs): 6.383% (9/141) 10 seq. 0, FEC BER % (errs): 6.383% (9/141) 11 sion lost, total frames: 28, bits: 3948, errors: 436, BER: 11.04357%</pre>	

Cas typique de test dans le mauvais mode de réception ci-dessus. Et même test après avoir appuyé sur la touche i ci-dessous :



L'indication « **DMR voice header received** » prouve qu'on intercepte bien les données dès l'appui sur le PTT du poste DMR. On reportera ce paramétrage de réception inversés dans le fichier de configuration du MMDVM un peu plus loin dans ce document.

Ensuite, tout en restant en émission, on tourne **R99** jusqu'à obtenir un BER de 0.000% :

Pi@raspberrypi: ~	- 🗆 🗙
DMR voice header received	
DMR voice header received	
DMR voice header received	
DMR audio seq. 0, FEC BER % (errs): 0.000% (0/141)	
DMR audio seq. 1, FEC BER % (errs): 0.000% (0/141)	
DMR audio seq. 2, FEC BER % (errs): 0.000% (0/141)	
DMR audio seq. 3, FEC BER % (errs): 0.000% (0/141)	
DMR audio seq. 4, FEC BER % (errs): 0.000% (0/141)	
DMR audio seq. 5, FEC BER % (errs): 0.000% (0/141)	
DMR audio seq. 0, FEC BER % (errs): 0.000% (0/141)	
DMR audio seg. 1, FEC BER % (errs): 0.000% (0/141)	
DMR audio seg. 2, FEC BER % (errs): 0.000% (0/141)	
DMR audio seq. 3, FEC BER $\hat{s}$ (errs): 0.000 $\hat{s}$ (0/141)	
DMR audio seq. 4, FEC BER 8 (errs): 0.0008 (0/141)	
DMR audio seq. 5, FEC DER  (errs): 0.000 (0/141)	
DMR audio seq. 1. FEC BER  (errs): 0.000 (0/141)	
DMR audio seq. 2. FEC BER  (errs): 0.000 (0/141)	
DMR audio seg. 3, FEC BER % (errs): 0.000% (0/141)	
DMR audio seg. 4, FEC BER % (errs): 0.000% (0/141)	
DMR audio seg. 5, FEC BER % (errs): 0.000% (0/141)	
DMR audio seq. 0, FEC BER % (errs): 0.000% (0/141)	
DMR audio seq. 1, FEC BER % (errs): 0.000% (0/141)	
DMR audio seq. 2, FEC BER % (errs): 0.000% (0/141)	
DMR audio seq. 3, FEC BER % (errs): 0.000% (0/141)	
DMR audio seq. 4, FEC BER % (errs): 0.000% (0/141)	
DMR audio seq. 5, FEC BER % (errs): 0.000% (0/141)	
DMR audio seq. 0, FEC BER % (errs): 0.000% (0/141)	
DMR audio seq. 1, FEC BER % (errs): 0.000% (0/141)	
DMR audio seq. 2, FEC BER % (errs): 0.000% (0/141)	
DMR audio seq. 3, FEC BER % (errs): 0.000% (0/141)	
DMR audio seg. 4, FEC BER % (errs): 0.000% (0/141)	
DMR audio seg. 5, FEC BER & (errs): 0.000% (0/141)	
DAR Voice end received, total frames: 30, bits: 4230, erfors: 0, BER: 0.0000%	

# Paramétrage de MMDVMHost

Le principe du MMDVM, c'est de pouvoir servir de relais radio pour différents modes de transmission, numériques ou analogique. Cela n'implique pas pour autant que tous ces modes seront activés par défaut. Il faut donc définir entre autre, dans le fichier de configuration du logiciel principal, les modes qui seront gérés par le relais MMDVM. À cela s'ajoutent les paramètres relatifs à la balise d'identification du relais en code morse, les niveaux audio en émission pour chaque mode, le paramétrage des différents périphériques de la carte MMDVM et la connexion éventuelle aux passerelles d'interconnexion des relais par l'intermédiaire de serveurs Internet.

L'accès en édition au fichier de configuration est effectué avec la commande suivante : *sudo nano /opt/MMDVMHost/MMDVM.ini* 

Ci-dessous une liste non exhaustive des paramètres, avec quelques explications.

### Paramètres généraux

[General]	
Callsign=Exvvv	Indicatif du relais, affiché sur l'écran relié au MMDVM
Td=208aabbbb	Identifiant (CS7 du relais affiché sur l'écran relié au MMDVM
Timeout=180	Identifiant CC3/ du l'Etais, all'iche sul i etian l'ette au minovi
Duplex=1	Fonctionnement en half-duplex (relais), $0 = simplex$ (point d'accès)
# ModeHang=10	· ····································
RFModeHang=10	
NetModeHang=3	
Display=Nextion	Type d'afficheur
Daemon=0	Exécution de MMDVMHost en tant que démon (pas utilisé, exécuté en tant que service grâce au paramétrage détaillé plus loin dans ce document)

### Informations sur le MMDVM

	Données complémentaires communiquées aux serveurs
[Info]	d'interconnexion
RXFrequency=439987500	Fréquence de réception
TXFrequency=430587500	Fréquence d'émission
Power=5	Puissance d'émission (P.A.R.)
# The following lines are only needed if a direct	
connection to a DMR master is being used	
Latitude=0.00	Coordonnée GPS : Latitude
Longitude=0.00	Coordonnée GPS : Longitude
Height=10	Hauteur de l'antenne par rapport au niveau de la mer
Location=JN12BL	Locator
Description=Relais MMDVM	Description de l'installation radio
URL=www.shibby.fr	Site Internet en lien avec le relais

### Journaux d'événements

[Log]	
<pre># Logging levels, 0=No logging</pre>	
DisplayLevel=1	
FileLevel=1	
FilePath=.	
FileRoot=MMDVM	
FileRotate=1	

### Balise d'identification en code morse

[CW Id]	
Enable=1	1 = activé, 0 = désactivé
Time=10	Délai en minutes entre chaque émission de la balise
Callsign=fxyyy locator 73	Message à émettre

### Fichiers de correspondance entre les identifiants (ex. : CCS7) et les indicatifs

[DMR Id Lookup] File=DMRIds.dat Time=24	
[NXDN Id Lookup] File=NXDN.cs∨ Time=24	

### Paramétrage du modem MMDVM

[Modem] Port=/dev/ttyACM0	Interface de communication Raspberry Pi ↔ MMDVM
<pre># Port=/dev/ttyAMA0</pre>	
<pre># Port=\\.\COM4</pre>	
Protocol=uart	Protocole de communication
# Address=0x22	
TXInvert=1	1 = Inversion de l'émission
RXInvert=1	1 = Inversion de la réception
PTTInvert=0	1 = Inversion de la commande PTT, $0 = $ pas d'inversion
TXDelay=100	

RXOffset=0	
TXOffset=0	
DMRDelay=83	
RXLevel=50	
TXLevel=50	
RXDCOffset=0	
TXDCOffset=0	
RFLevel=100	
CWIdTXLevel=50	Niveau audio spécifique à l'émission de la balise CW
# D-StarTXLevel=50	
DMRTXLevel=50	Niveau audio spécifique à l'émission DMR
# YSFTXLevel=50	
# P25TXLevel=50	
# NXDNTXLevel=50	
POCSAGTXLevel=18.5	Niveau audio spécifique à l'émission DAPNET
FMTXLevel=43	Niveau audio spécifique à l'émission analogique
RSSIMappingFile=RSSI.dat	
UseCOSAsLockout=0	
Trace=0	
Debug=0	

## Paramétrage DMR « local »

[DMR]	
Enable=1	1 = Relayage des émissions DMR activé, 0 = désactivé
Beacons=0	
BeaconInterval=60	
BeaconDuration=3	
ColorCode=1	
SelfOnly=0	

EmbeddedLCOnly=0	
DumpTAData=1	
# Prefixes=234,235	
<pre># Slot1TGWhiteList=</pre>	
<pre># Slot2TGWhiteList=</pre>	
CallHang=3	
TXHang=4	
# ModeHang=10	
<pre># OVCM Values, 0=off, 1=rx_on, 2=tx_on, 3=both_on,</pre>	
4=force off	
# OVCM=0	

# Paramétrage général DAPNET

[POCSAG]	
Enable=1	1 = Activé
Frequency=439987500	Fréquence d'émission

## Paramétrage général FM (relais analogique)

[FM]	
Enable=0	1 = activé, 0 = désactivé
Callsign=fxyyy locator 73	Indicatif émis par la balise CW
CallsignSpeed=15	Nombre de mots par minute
CallsignFrequency=820	Fréquence de la tonalité CW
CallsignTime=10	Délai en minutes entre chaque émission de la balise
CallsignHoldoff=0	
CallsignHighLevel=50	Niveau audio de la balise en l'absence d'émission FM
CallsignLowLevel=20	Niveau audio si une émission phonie est en cours

CallsignAtStart=1	Émission de la balise au déclenchement du relais
CallsignAtEnd=1	Émission de la balise à la fermeture du relais
CallsignAtLatch=0	
RFAck=K	Indication de fin de prise de parole « K »
ExtAck=N	
AckSpeed=20	Nombre de mots par minutes du « K »
AckFrequency=1750	Fréquence de la tonalité du « K »
AckMinTime=4	Durée min d'une émission en secondes suivie d'un « K »
AckDelay=1000	Délai en millisecondes entre la fin d'émission et le K
AckLevel=50	Niveau audio du « K »
# Timeout=180	
TimeoutLevel=80	
CTCSSFrequency=88.4	Fréquence de la tonalité subaudible « CTCSS »
CTCSSThreshold=30	Seuil de détection du CTCSS
<pre># CTCSSHighThreshold=30</pre>	Seuil haut d'hystérésis de détection du CTCSS
<pre># CTCSSLowThreshold=20</pre>	Seuil bas d'hystérésis de détection du CTCSS
CTCSSLevel=20	
KerchunkTime=0	Délai avant ouverture du relais
HangTime=7	
AccessMode=1	
COSInvert=0	
RFAudioBoost=1	
MaxDevLevel=90	
ExtAudioBoost=1	

# Paramétrage d'accès à la passerelle DMR (DMRGateway)

[DMR Network] Enable=0	1 = activé, 0 = désactivé
# Type may be either 'Direct' or 'Gateway'. When	
Direct you must provide the Master's # address as well as the Password and for DMR+	
--	---
Options also.	
Type=Gateway	Laisser tel quel
Address=127.0.0.1	Adresse du DMRGateway : 127.0.0.1 = local
Port=62031	Port distant, laisser tel quel
Local=62032	Port local, laisser tel quel
Password=P@ssw0rd	Mot de passe d'accès à la passerelle locale
Jitter=360	
Slot1=1	
Slot2=1	
# Options=	
# ModeHang=3	
Debug=0	

# Paramétrage d'accès à la passerelle DAPNET (DAPNETGateway)

[POCSAG Network] Enable=1	
LocalAddress=127.0.0.1	
LocalPort=3800	
GatewayAddress=127.0.0.1	
GatewayPort=4800	
# ModeHang=3	
Debug=1	

# Paramétrage de l'afficheur Nextion

[Nextion]

Port=modem	Port sur lequel est relié l'afficheur, modem =
#Port=/dev/ttyAMA0	connecteur dédié sur l'interface MMDVM
Brightness=30	Luminosité quand le MMDVM est actif (TX en cours)
DisplayClock=1	Affichage de l'horloge : 1 = activé, 0 = désactivé
UTC=0	Affichage de l'heure UTC : 1 = oui, 0 = non
#Screen Layout: 0=G4KLX 2=ON7LDS	Mode d'affichage, selon la source du fichier chargé
ScreenLayout=0	dans l'écran Nextion avec Nextion.py
IdleBrightness=20	Luminosité de l'écran lorsque le relais est inactif

Il faut noter que la journalisation des événements servant au « débug » créé des fichiers pouvant devenir volumineux, selon le niveau de détail des événements enregistrés. La carte mémoire peut être saturée par ces fichiers, ce qui causera de nombreux dysfonctionnements du système Linux lui-même, et rendra le MMDVM inopérant. Lorsque la présence de ces fichiers n'est plus nécessaire, soit une fois que le MMDVM est stable, pensez à supprimer les fichiers se terminant par l'extension .log présents dans les répertoires d'installation des différents logiciels constituant le MMDMV, et à désactiver la création de ces fichiers en remplaçant le paramétrage Debug=1 par Debug=0 partout où il apparaît dans les fichiers de configuration mentionnés dans ce document.

# Exécution automatique des logiciels au démarrage de la Raspberry Pi

Ici nous allons nous intéresser au démarrage automatisé des services liés au MMDVM, afin que cet équipement puisse fonctionner de façon autonome tant que son alimentation électrique est présente, ainsi qu'après une perte temporaire de sa source d'alimentation électrique.

Certains logiciels ne vous seront pas forcément nécessaires, selon le type de relais que vous souhaitez mettre en place.

**MMDVMHost** constitue la base du MMDVM, ce qui rend son exécution obligatoire.

**DMRGateway** permet le lien entre votre relais MMDVM DMR et un serveur spécifique permettant l'interconnexion avec d'autres relais par un lien Internet.

**DAPNETGateway** réalise aussi un lien entre votre relais MMDVM DAPNET et un serveur spécifique permettant la diffusion des messages transitant par celui-ci. Ceci permet par exemple de diffuser une info vers des destinataires quel que soit le relais dont ils dépendent.

# **Pré-requis**

## Installation du multiplexeur de terminaux « screen »

sudo apt-get install screen -y

# **MMDVMHost**

sudo nano /lib/systemd/system/mmdvmhost.service

Copier/coller le texte affiché en gras ci-dessous dans le fichier mmdvmhost.service en cours de création : [Unit] Description=MMDVM Host Service After=syslog.target network.target

[Service] User=root WorkingDirectory=/opt/MMDVMHost ExecStart=/usr/bin/screen -S MMDVMHost -D -m /opt/MMDVMHost/MMDVMHost /opt/MMDVMHost/MMDVM.ini ExecStop=/usr/bin/screen -S MMDVMHost -X quit Restart=always RestartSec=10

## [Install] WantedBy=multi-user.target

Quitter l'éditeur de texte en sauvegardant le fichier avec **Ctrl+X**, puis valider l'enregistrement avec la touche **Y** et **Entrée**.

sudo nano /lib/systemd/system/mmdvmhost.timer

Copier/coller le texte affiché en gras ci-dessous dans le fichier mmdvmhost.timer en cours de création : [Timer] OnStartupSec=60

## [Install] WantedBy=multi-user.target

Quitter l'éditeur de texte en sauvegardant le fichier avec Ctrl+X, puis valider l'enregistrement avec la touche Y et Entrée.

sudo chmod 755 /lib/systemd/system/mmdvmhost.service sudo chmod 755 /lib/systemd/system/mmdvmhost.timer sudo ln -s /lib/systemd/system/mmdvmhost.service /etc/systemd/system/mmdvmhost.service sudo ln -s /lib/systemd/system/mmdvmhost.timer /etc/systemd/system/mmdvmhost.timer

Activer l'exécution automatique du timer à chaque mise sous tension de la Raspberry Pi : *sudo systemctl enable mmdvmhost.timer* 

Redémarrer le démon pour prendre en compte les nouveaux paramètres : *sudo systemctl daemon-reload* 

Fonctions de gestion du service (à mémoriser, il n'est pas utile de les exécuter maintenant) : sudo systemctl start mmdvmhost.service sudo systemctl restart mmdvmhost.service sudo systemctl stop mmdvmhost.service

Information sur le statut du service (à mémoriser, il n'est pas utile de l'exécuter maintenant) : *sudo systemctl status mmdvmhost.service* 

# **DMRGateway**

sudo nano /lib/systemd/system/dmrgateway.service

Copier/coller le texte affiché en gras ci-dessous dans le fichier dmrgateway.service en cours de création : [Unit] Description=DAPNET Gateway Service After=syslog.target network.target

```
[Service]
User=root
WorkingDirectory=/opt/DMRGateway
ExecStart=/usr/bin/screen -S DMRGateway -D -m /opt/DMRGateway/DMRGateway /opt/DMRGateway/DMRGateway.ini
ExecStop=/usr/bin/screen -S DMRGateway -X quit
Restart=always
RestartSec=10
```

[Install] WantedBy=multi-user.target

Quitter l'éditeur de texte en sauvegardant le fichier avec **Ctrl+X**, puis valider l'enregistrement avec la touche **Y** et **Entrée**.

sudo nano /lib/systemd/system/dmrgateway.timer

Copier/coller le texte affiché en gras ci-dessous dans le fichier dmrgateway.timer en cours de création : [Timer] OnStartupSec=60

## [Install] WantedBy=multi-user.target

Quitter l'éditeur de texte en sauvegardant le fichier avec Ctrl+X, puis valider l'enregistrement avec la touche Y et Entrée.

sudo chmod 755 /lib/systemd/system/dmrgateway.service sudo chmod 755 /lib/systemd/system/dmrgateway.timer sudo ln -s /lib/systemd/system/dmrgateway.service /etc/systemd/system/dmrgateway.service sudo ln -s /lib/systemd/system/dmrgateway.timer /etc/systemd/system/dmrgateway.timer

Activer l'exécution automatique du timer à chaque mise sous tension de la Raspberry Pi : *sudo systemctl enable dmrgateway.timer* 

Redémarrer le démon pour prendre en compte les nouveaux paramètres : *sudo systemctl daemon-reload* 

Fonctions de gestion du service (à mémoriser, il n'est pas utile de les exécuter maintenant) : sudo systemctl start dmrgateway.service sudo systemctl restart dmrgateway.service sudo systemctl stop dmrgateway.service

Information sur le statut du service (à mémoriser, il n'est pas utile de l'exécuter maintenant) : *sudo systemctl status dmrgateway.service* 

# **DAPNETGateway**

sudo nano /lib/systemd/system/dapnetgateway.service

Copier/coller le texte affiché en gras ci-dessous dans le fichier dapnetgateway.service en cours de création : [Unit] Description=DAPNET Gateway Service After=syslog.target network.target

[Service] User=root WorkingDirectory=/opt/DAPNETGateway ExecStart=/usr/bin/screen -S DAPNETGateway -D -m /opt/DAPNETGateway/DAPNETGateway /opt/DAPNETGateway/DAPNETGateway.ini ExecStop=/usr/bin/screen -S DAPNETGateway -X quit Restart=always RestartSec=10

[Install] WantedBy=multi-user.target

Quitter l'éditeur de texte en sauvegardant le fichier avec **Ctrl+X**, puis valider l'enregistrement avec la touche **Y** et **Entrée**.

sudo nano /lib/systemd/system/dapnetgateway.timer

Copier/coller le texte affiché en gras ci-dessous dans le fichier dapnetgateway.timer en cours de création : **[Timer]** 

### OnStartupSec=60

### [Install] WantedBy=multi-user.target

Quitter l'éditeur de texte en sauvegardant le fichier avec **Ctrl+X**, puis valider l'enregistrement avec la touche **Y** et **Entrée**.

sudo chmod 755 /lib/systemd/system/dapnetgateway.service sudo chmod 755 /lib/systemd/system/dapnetgateway.timer sudo ln -s /lib/systemd/system/dapnetgateway.service /etc/systemd/system/dapnetgateway.service sudo ln -s /lib/systemd/system/dapnetgateway.timer /etc/systemd/system/dapnetgateway.timer

Activer l'exécution automatique du timer à chaque mise sous tension de la Raspberry Pi : *sudo systemctl enable dapnetgateway.timer* 

Redémarrer le démon pour prendre en compte les nouveaux paramètres : *sudo systemctl daemon-reload* 

Fonctions de gestion du service (à mémoriser, il n'est pas utile de les exécuter maintenant) : sudo systemctl start dapnetgateway.service sudo systemctl restart dapnetgateway.service sudo systemctl stop dapnetgateway.service

Information sur le statut du service (à mémoriser, il n'est pas utile de l'exécuter maintenant) : sudo systemctl status dapnetgateway.service

# Et maintenant...

Après avoir réalisé toutes les étapes décrites ci-dessus, votre MMDVM devrait être en mesure de retransmettre vos premiers essais radio. Quelques modifications et ajustements de paramétrage de MMDVMHost peuvent être encore nécessaires, mais vous êtes proche du but. Ce document est encore en cours d'édition. Vous pouvez tout de même poster vos questions éventuelles sur mon blog en cliquant sur le lien fourni au début de la première page. Ceci m'aidera à rendre ce document le plus complet possible.

# Annexe 1 – Ancienne méthode de compilation, sous Windows

Les informations fournies ci-dessous sont conservées à titre purement informatif. Une nouvelle méthode, beaucoup plus simple, est fournie précédemment dans ce document.

# Chargement du projet MMDVM et modification des paramètres de compilation

Aller sur la page Github suivante : <u>https://github.com/g4klx/MMDVM</u> Cliquer sur le bouton vert en haut à droite :

Go to file	🛓 Code 🗸			
Clone     HTTPS GitHub CLI	0			
https://github.com/g4klx/MMDVM.git       Use Git or checkout with SVN using the web URL.				
① Open with GitHub Desktop				
a Download ZIP				

x

#### http://blog.shibby.fr

Cliquer sur le sous-menu de téléchargement des fichiers sous format d'archive ZIP : **Download ZIP**. Enregistrer le fichier dans le répertoire souhaité :

Vous avez choisi d'ouvrir : MMDVM-master.zip qui est un fichier de type : Compressed (zipped) Folder à partir de : https://codeload.github.com Que doit faire Firefox avec ce fichier ? Quvrir avec Explorateur Windows (par défaut) Enregistrer le fichier Toujours effectuer cette action pour ce type de fichier. OK Annuler

Ouverture de MMDVM-master.zip

Ici je choisis de l'enregistrer dans c:\temp\MMDVM\ :

🐞 Saisissez le nom du	fichier pour l'enregistrement		x
C→ ↓ Ord	dinateur ► Disque local (C:) ► temp ► MMDVM -	← Rechercher dans : MMDVM	٩
<u>N</u> om du fichier :	MMDVM-master.zip		•
<u>T</u> ype :	Compressed (zipped) Folder (*.zip)		•
💿 <u>P</u> arcourir les dossi	iers	<u>Enregistrer</u> Annuler	

Depuis le répertoire où est enregistré le fichier, double-cliquer sur le fichier et cliquer sur le bouton Extraire les fichiers :



Définir le répertoire dans lequel les fichiers vont être extraits. Je conserve le chemin défini par défaut dans cet exemple :

🕞 🚯 Extraire les dossiers compressés	×
Sélectionner une destination et extraire les fichiers	
Les fichiers seront extraits dans ce dossier :	
C:\temp\MMDVM\MMDVM-master	P <u>a</u> rcourir
Affic <u>h</u> er les dossiers extraits une fois l'opération terminée	
	<u>Extraire</u> Annuler

Cliquer sur Extraire. L'extraction est exécutée :



Ce nouveau répertoire apparaît à l'endroit défini à la fin de l'opération :

Organiser ▼ 📜 Ouvrir Inclure d	cal (C lans la	:) ▶ temp ▶ MMDVM ▶ i bibliothèque ▼ Partager avec ▼ Grave	✓ ← Rechercher Nouveau dossier	dans : MMDVM	<u>م</u> م
🤣 Groupe résidentiel	^	Nom  MMDVM-master	Modifié le 24/04/2021 12:17	Type Dossier de fichiers	Taille
Ordinateur     Ordinateur     Second (C:)     USE Storage (\)102168.01\//Hi)	Ţ	MMDVM- Date de création : 24/04/2021 12 Taille : 634 Ko Dossiers : MMDVM-master	17 1/04/2021 12:14	Dossier compressé	217 Ko
1 élément sélectionné				p Ordinateur	

Accéder au contenu de ce nouveau répertoire pour atteindre le fichier **config.h**, effectuer un clic droit sur le nom du fichier et cliquer sur **Modifier** :



Si vous utilisez l'interface de F5UII sur une carte ST NUCLEO-64 STM32F446, vous devez définir les paramètres suivants :

### /\*

\* Copyright (C) 2015,2016,2017,2018,2020 by Jonathan Naylor G4KLX

\*

\* This program is free software; you can redistribute it and/or modify

\* it under the terms of the GNU General Public License as published by

\* the Free Software Foundation; either version 2 of the License, or

\* (at your option) any later version.

\*

\* This program is distributed in the hope that it will be useful,

\* but WITHOUT ANY WARRANTY; without even the implied warranty of

\* MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the

\* GNU General Public License for more details.

\*

\* You should have received a copy of the GNU General Public License

\* along with this program; if not, write to the Free Software

\* Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

```
*/
```

#if !defined(CONFIG\_H)
#define CONFIG H

// Allow for the use of high quality external clock oscillators

// The number is the frequency of the oscillator in Hertz.

//

// The frequency of the TCXO must be an integer multiple of 48000.

// Frequencies such as 12.0 Mhz (48000 \* 250) and 14.4 Mhz (48000 \* 300) are suitable.

// Frequencies such as 10.0 Mhz (48000 \* 208.333) or 20 Mhz (48000 \* 416.666) are not suitable.

// // For 12 MHz #define EXTERNAL\_OSC 12000000 // For 12.288 MHz // #define EXTERNAL\_OSC 12288000 // For 14.4 MHz // #define EXTERNAL\_OSC 14400000 // For 19.2 MHz // #define EXTERNAL\_OSC 19200000

// Use pins to output the current mode via LEDs
#define MODE\_LEDS

// For the original Arduino Due pin layout
// #define ARDUINO\_DUE\_PAPA

#if defined(STM32F1)
// For the SQ6POG board
#define STM32F1\_POG
#else
// For the ZUM V1.0 and V1.0.1 boards pin layout
#define ARDUINO\_DUE\_ZUM\_V10
#endif

// For the SP8NTH board
// #define ARDUINO\_DUE\_NTH

// For ST Nucleo-64 STM32F446RE board
#define STM32F4\_NUCLEO\_MORPHO\_HEADER
// #define STM32F4\_NUCLEO\_ARDUINO\_HEADER

// Use separate mode pins to switch external channel/filters/bandwidth for example #define MODE\_PINS

// For the VK6MST Pi3 Shield communicating over i2c. i2c address & speed defined in i2cTeensy.cpp // #define VK6MST\_TEENSY\_PI3\_SHIELD\_I2C

// Pass RSSI information to the host
#define SEND\_RSSI\_DATA

// Use the modem as a serial repeater for Nextion displays
#define SERIAL\_REPEATER

// To reduce CPU load, you can remove the DC blocker by commenting out the next line #define USE\_DCBLOCKER

// Constant Service LED once repeater is running
// Do not use if employing an external hardware watchdog
// #define CONSTANT\_SRV\_LED

// Use the YSF and P25 LEDs for NXDN
#define USE\_ALTERNATE\_NXDN\_LEDS

// Use the D-Star and DMR LEDs for POCSAG

### #define USE\_ALTERNATE\_POCSAG\_LEDS

# // Use the D-Star and YSF LEDs for FM #define USE\_ALTERNATE\_FM\_LEDS

### #endif

Enregistrez les modifications dans le fichier **Config.h** et quittez l'éditeur.**Librairie STM32F4XX** 

Télécharger l'archive ZIP disponible à cette adresse : <u>http://dl.shibby.fr/Radio/MMDVM/STM32F4XX\_Lib.zip</u> Ici je choisis de l'enregistrer dans **c:\temp\MMDVM\** :

👏 Saisissez le nom du	fichier pour l'enregistrement			×
😋 🗢 🗣 🚺 « Dis	ique local (C:) → temp → MMDVM →	<b>- 4</b> 7	Rechercher dans : MMDVM	م
<u>N</u> om du fichier :	STM32F4XX_Lib.zip			-
<u>T</u> ype :	Compressed (zipped) Folder (*.zip)			•
<u>P</u> arcourir les doss	iers	(	<u>Enregistrer</u> Annu	ler

Depuis le répertoire où est enregistré le fichier, double-cliquer sur le fichier et cliquer sur le bouton Extraire les fichiers :

G 🖉 🚽 🚺 🕨 Ordinateur	Disque local (C:) + temp + MMDVM + STM	B2F4XX_Lib.zip ►		✓  Freehercher	r dans : STM32F4XX_Lib.zip 👂
Organiser 👻 Extraire tous	s les fichiers				III • 🔟 🔞
Téléchargements	Extraire tous les fichiers	Туре	Taille compressée	Protégé pa Taille	Ratio Modifié le
Bibliothèques	STM32F4XX_Lib	Dossier de fichiers			02/12/2016 19:18
🥽 Bibliothèques	II				
🍓 Groupe résidentiel					
🖳 Ordinateur					
🏭 Disque local (C:)	<b>▼</b>		III		•
STM32F4XX_Lib	Modifié le : 02/12/2016 19:18 Type : Dossier de fichiers				
1 élément sélectionné					

Définir le répertoire dans lequel les fichiers vont être extraits. Il faut que le répertoire de destination soit celui qui contient les fichiers MMDVM. Si vous suivez à la lettre cet exemple, il faut définir le chemin suivant : **C:\temp\MMDVM\MMDVM-master\MMDVM-master** 

	×
🕞 🚹 Extraire les dossiers compressés	
Sélectionner une destination et extraire les fichiers	
Les fichiers seront extraits dans ce dossier :	
C:\temp\MMDVM\MMDVM-master\MMDVM-master	P <u>a</u> rcourir
	<u>E</u> xtraire Annuler

Cliquer sur Extraire. L'extraction est exécutée :



De retour dans le répertoire contenant les fichiers MMDVM, on constate que le répertoire a été créé et contient les fichiers contenus dans l'archive ZIP :

#### Rev. 04/06/2022 @ 22:29

							x
C Vidinateur > Disque loc	al (C:) + temp + MMDVM + MMDVM-master	► MMDVM-master ►		•	Rechercher dans : MMDV	M-master	Q
Organiser 🔻 😭 Ouvrir 🛛 Inclure da	ns la bibliothèque 🔻 🛛 Partager avec 💌 Gr	raver Nouveau dossier				•	0
🐌 Téléchargements	^ Nom	Modifié le	Туре	Taille			-
詞 Bibliothèques	📄 🔑 pins	24/04/2021 12:17	Dossier de fichiers				
Di Li atta à sura s	STM32F4XX_Lib	25/04/2021 17:16	Dossier de fichiers				
Groupe résidentiel	STM32F7XX_Lib STM32F10X_Lit Taille : 5,20 Mo	20/04/2021 04:51 4/2021 12:17	<u>Dossier</u> de fichiers de fichiers				
	system_stm32f: Dossiers : CMSIS, Device Tools	e, STM32F4xx_StdPeriph_Dr rty SW License Agreement \	iver de fichiers /2.pdf, de fichiers				
Lisque local (C:)	gitignore	24/04/2021 12:17	Fichier GITIGNORE	1 Ko			
STM32F4XX_Lib Modifié le : Dossier de fichiers	25/04/2021 17:16	24/04/2021 12:17	Fichier GITMODU	1 Ko			T
1 élément sélectionné					j🌉 O	rdinateur	

# **Compilateur gnuwin32**

Télécharger l'archive ZIP disponible à cette adresse : <u>http://dl.shibby.fr/Radio/MMDVM/make.zip</u> Ici je choisis de l'enregistrer dans **c:\temp\MMDVM\** :

🐞 Saisissez le nom du	fichier pour l'enregistrement	×
😋 🖉 🛡 📕 « Dis	que local (C:) → temp → MMDVM →	<ul> <li>✓</li> <li>✓</li></ul>
<u>N</u> om du fichier :	make.zip	•
<u>T</u> ype :	Compressed (zipped) Folder (*.zip)	-
💿 <u>P</u> arcourir les doss	iers	<u>Enregistrer</u> Annuler

Depuis le répertoire où est enregistré le fichier, double-cliquer sur le fichier et cliquer sur le bouton Extraire les fichiers :

Ordinateur ►	Disque local (C:	:) ▶ temp ▶ MMDVM ▶ make	.zip		<b>→</b> 4	Rechercher dans : m	ake.zip
Organiser 🔻 🛛 Extraire tous le	es fichiers						:= • 🔟 🔞
🐌 Téléchargements	Extraire tous	les fichiers	Туре	Taille compressée	Protégé pa Ta	lle Rat	io Modifié le
詞 Bibliothèques		🚳 libiconv2.dll	Extension de l'application	624 Ko	Non	877 Ko 29	% 07/04/2018 11:44
🚍 Bibliothèques	E	libintl3.dll	Extension de l'application	50 Ko	Non	101 Ko 51 °	% 07/04/2018 11:44
(a) bibliotricques		🗾 make.exe	Application	89 Ko	Non	171 Ko 49	% 07/04/2018 11:43
🤣 Groupe résidentiel							
🖳 Ordinateur							
🏭 Disque local (C:)		•		III			
3 élément(s)							
3 éléments							

Définir le répertoire dans lequel les fichiers vont être extraits. Il faut que le répertoire de destination soit celui qui contient les fichiers MMDVM. Si vous suivez à la lettre cet exemple, il faut définir le chemin suivant : **C:\temp\MMDVM\MMDVM-master\MMDVM-master** 

Evtraire les dossiers compressés	
Sélectionner une destination et extraire les fichiers	
Les fichiers seront extraits dans ce dossier :	
C:\temp\MMDVM\MMDVM-master\MMDVM-master	P <u>a</u> rcourir
	<u>Extraire</u> Annuler

Cliquer sur Extraire. L'extraction est exécutée :

Scopie de 3 éléments (1,12 Mo)	
Copie de 3 éléments (1,12 Mo)	
de <b>make.zip</b> (C:\make.zip) vers <b>MMDVM-ma</b> Détection de 3 éléments (1,12 Mo)	\MMDVM-ma
Plus de détails	Annuler

De retour dans le répertoire contenant les fichiers MMDVM, on constate que les fichiers contenus dans l'archive ZIP ont bien été copiés :

								x	
🚱 🗢 🔋 🕨 Ordinateur 🕨 Disque local (	(C:) ▶ temp ▶ MMDVM ▶ MMDVM-master	► MMDVM-master ►			🔹 🍫 Recherch	er dans : MMDVM-maste	r	٩	
Organiser 🔻 🔳 Ouvrir avec Grave	er Nouveau dossier					:==	•	0	
📜 Téléchargements 🔺	Nom	Modifié le	Туре	Taille				-	
Bibliothèques	S libiconv2.dll	25/04/2021 17:26	Extension de l'app	877 Ko					
E Riklinsk konse	💿 libintl3.dll	25/04/2021 17:26	Extension de l'app	101 Ko					
	💜 make.exe	25/04/2021 17:26	Application	171 Ko					
a Groupo récidential	FMDirectForm1.h	24/04/2021 12:17	Fichier H	4 Ko					
Groupe residentier	FMDownsampler.cpp	24/04/2021 12:17	Fichier CPP	2 Ko					
Ordinateur	FMDownsampler.h	24/04/2021 12:17	Fichier H	2 Ko					
	FMDownsampleRB.cpp	24/04/2021 12:17	Fichier CPP	2 Ko					
	FMDownsampleRR h	24/04/2021 12:17	Fichier H	2 Ko				Ŧ	
3 éléments sélectionnés Modif T	3 éléments sélectionnés       Modifié le : 25/04/2021 17:26       Date de création : 07/04/2018 11:44         Taille : 1,12 Mo       Date de création : 07/04/2018 11:44								
3 éléments sélectionnés						j 💺 Ordinateu	r	.#	

### Télécharger l'outil de compilation pour processeur ARM

Cet outil permet de créer un fichier qui pourra être exécuté par le microcontrôleur présent sur la carte STM32F466.

Télécharger l'archive ZIP disponible à cette adresse :<u>http://dl.shibby.fr/Radio/MMDVM/gcc-arm-none-eabi-5\_4-2016q3-20160926-win32.exe</u>

Ici je choisis de l'enregistrer dans c:\temp\MMDVM\ :

📦 Saisissez le nom du	fichier pour l'enregistrement	
OO - 📔 « Die	sque local (C:) → temp → MMDVM →	✓ 4y Rechercher dans : MMDVM
<u>N</u> om du fichier :	gcc-arm-none-eabi-5_4-2016q3-20160926-win32.exe	•
<u>T</u> ype :	exe File (*.exe)	▼
ercourir les doss	iers	Enregistrer Annuler

Depuis le répertoire où est enregistré le fichier, double-cliquer sur le fichier et suivre les étapes ci-dessous :

• Valider le langage utilisé par le programme d'installation en cliquant sur **OK** :

Installer La	nguage						
Please select a language.							
	French						
	OK Cancel						

• Cliquer sur **Suivant** :



• Accepter la licence utilisateur en cliquant sur J'accepte :



• Laisser le dossier d'installation par défaut et valider l'installation en cliquant sur Installer :

Choisissez le dossier d'installation	1	NUMB
Choisissez le dossier dans lequel insta 5.4 2016.	ller GNU Tools for ARM Embedded Pro	cessors 🥡
Ceci installera GNU Tools for ARM Eml Pour installer dans un autre dossier, Cliquez sur Installer pour démarrer l'in	oedded Processors 5.4 2016 dans le c liquez sur Parcourir et choisissez un a stallation.	lossier suivant. utre dossier.
Dossier d'installation C:\Program Files (x86)\GNU Tool	ARM Embedded\5.4 2016q3	P <u>a</u> rcourir
Espace requis : 419.4Mo Espace disponible : 336.5Go		
Espace requis : 419.4Mo Espace disponible : 336.5Go		

L'installation est exécutée. Attendre qu'elle se termine :

G GNU Tools for ARM Embedded Processors
Installation en cours Veuillez patienter pendant que GNU Tools for ARM Embedded Processors 5.4 2016 est en train d'être installé.
Extraction : libstdc++.a
Plus d'infos
Nullsoft Install System v07-Apr-2015.cvs

À la fin de l'installation, cocher la case indiquée ci-dessous et cliquer sur Fermer :



# Compilation

On va enfin pouvoir compiler le logiciel embarqué destiné à la carte STM32F446 !

Dans la barre des tâches de Windows, cliquer sur l'icône à gauche (menu « démarrer »), et taper GCC :

Programmes (1)	
GCC Command Prompt	
₽ Voir plus de résultats	
gcc ×	Arrêter 🕨

Cliquer sur GCC Command Prompt. La fenêtre suivante apparaît à l'écran :



Aller dans le répertoire où se trouvent tous les fichiers MMDVM avec la commande suivante :

### cd c:\temp\MMDVM\MMDVM-master\MMDVM-master

Valider avec la touche Entrée.



Taper make nucleo et valider avec la touche Entrée. La compilation des fichiers commence alors :



À la fin de cette opération, vous devez voir le message suivant s'afficher :

ſ	Administrateur : GCC Command Prompt	
	j_f4/CWIdTX.o obj_f4/DMRDMORX.o obj_f4/DMRDMOTX.o obj_f4/DMRIdleRX.o obj_f4/DMRR X.o obj_f4/DMRSlotRX.o obj_f4/DMRSlotType.o obj_f4/DMRTX.o obj_f4/DStarRX.o obj_	*
	f4/DStarTX.o obj_f4/FM.o obj_f4/FMBlanking.o obj_f4/FMCTCSSRX.o obj_f4/FMCTCSSTX .o obj_f4/FMDownsampler.o obj_f4/FMDownsampleRB.o obj_f4/FMKeyer.o obj_f4/FMRB.o	
	obj_f4/FMTimeout.o`obj_f4/FMTimer.o obj_f4/I2CTeensy.o obj_f4/I0.o obj_f4/I0Due .o obj_f4/I0STM.o obj_f4/I0STM_CMSIS.o obj_f4/I0Teensy.o obj_f4/MMDUM.o obj_f4/N	
	XDNRX.o obj_f4/NXDNTX.o obj_f4/P25RX.o obj_f4/P25TX.o obj_f4/POCSAGTX.o obj_f4/R SSIRB.o obj_f4/SampleRB.o obj_f4/SerialArduino.o obj_f4/SerialPort.o obj_f4/Seri	
	alRB.o obj_f4/SerialSTM.o obj_f4/SerialSTM_CMSIS.o obj_f4/Utils.o obj_f4/YSFRX.o obj_f4/YSFTX.o obj_f4/misc.o obj_f4/stm32f4xx_adc.o obj_f4/stm32f4xx_dac.o obj_	
	f4/stm32f4xx_gpio.o obj_f4/stm32f4xx_rcc.o obj_f4/stm32f4xx_tim.o obj_f4/stm32f4 xx_usart.o obj_f4/system_stm32f4xx.o obj_f4/startup_stm32f4xx.o -Osspecs=nano	
	.specs -W1,-Map=bin/mmdvm.map -T stm32f4xx_link.ld -mcpu=cortex-m4 -mthumb -mlit tle-endian -mfpu=fpv4-sp-d16 -mfloat-abi=hard -mthumb-interworkspecs=nosys.sp	
	ecs ./SIM32F4XX_Lib/CMSIS/Lib/GCC/libarm_cortexM41f_math.a -o bin/mmdym_f4.elf "Linking complete!\n"	
	arm-none-eadi-size bin/mmdvm_f4.elf text data bss dec hex filename 200002 d(co 5004 242064 52-24 bis (sedue 64 alf	
	arm-none-eabi-objcopy -0 ihex bin/mmdvm_f4.elf bin/mmdvm_f4.hex	
	arm-none-eabi-objcopy -0 binary bin/mmdvm_f4.elf bin/mmdvm_f4.bin	
	c:\temp\MMDHM\MMDHM=mastex\MMDHM=mastex\	-
L		

On voit que les fichiers compilés **mmdvm\_f4.hex** et **mmdvm\_f4.bin** ont été créés. Il s'agit du même fichier compilé, mais dans deux formats différents.

# **Programmation de la carte STM32F466**

# Installation du logiciel de programmation

Télécharger l'archive ZIP disponible à cette : <u>http://dl.shibby.fr/Radio/MMDVM/en.stsw-link004.zip</u> Ici je choisis de l'enregistrer dans **c:\temp\MMDVM\** :

🐞 Saisissez le nom du	i fichier pour l'enregistrement	3
🖉 🖉 🖉 🖉 🖉	sque local (C:) > temp > MMDVM > - 4	٩
<u>N</u> om du fichier :	en.stsw-link004.zip	-
<u>T</u> ype :	Compressed (zipped) Folder (*.zip)	-
ercourir les doss	iers <u>Enregistrer</u> Annuler	]

Depuis le répertoire où est enregistré le fichier, double-cliquer sur le fichier, puis sur le répertoire, puis sur le fichier setup.exe

											x
😋 🔍 🔻 📕 🕨 Ordinateur 🕨 Disque la	ocal (C	:) ▶ temp ▶ MMD	VM ▶ en.stsw-link004.zip ▶ STM3	2 ST-LINK Utility v4.6.0		•	· <b>*</b> *	Rechercher dan	is : STM32	ST-LINK Utility v4.6.	0 🔎
Organiser   Extraire tous les fichiers										•= •	?
🗼 Téléchargements	*	Nom	Туре	Taille co	mpressée	Protégé pa	Taille		Ratio	Modifié le	
Calification Bibliothèques		setup.exe	Application		25 988 Ko	Non		27 004 Ko	4 %	01/10/2020 0	9:51
🥽 Bibliothèques	-										
📢 Groupe résidentiel											
🖳 Ordinateur											
🚢 Disque local (C:)	•	•									۰.
setup.exe Taille compressée Taille	: 25,3 : 26,3	Mo Mo	Ratio : 4 % Modifié le : 01/10/2020 09:51	Type: Application							
1 élément sélectionné											ai

### Cliquer ensuite sur **Exécuter** :



L'installation du logiciel s'exécute :



Suivre les étapes décrites ci-dessous :

• Cliquer sur **Next** :

Rev. 04/06/2022 @ 22:29



• Accepter la licence utilisateur en cliquant sur Yes :



• Conserver le chemin d'installation par défaut et valider en cliquant sur Next :
Choose De	stination Location
Select fol	der where Setup will install files.
Setup will	install STM32 ST-LINK Utility in the following folder.
To install another fo	to this folder, click Next. To install to a different folder, click Browse and select Ider.
Destina C:\\S	tion Folder TMicroelectronics\STM32 ST-LINK Utility\Browse

L'installation débute :

Setup Status	X
STM32 ST-LINK Utility Setup is performing the requested operations.	
Removing applications	
stallShield	
	Cancel

Valider l'installation des pilotes nécessaires pour communiquer avec la carte STM32F466 en cliquant sur **Suivant** :



Puis sur **Terminer** à la fin de l'installation :



Quitter enfin le programme d'installation en cliquant sur **Finish** :

# Rev. 04/06/2022 @ 22:29

# http://blog.shibby.fr



# Chargement du logiciel MMDVM dans la carte STM32F466

Connecter un cordon Mini USB entre le PC et la carte STM32F466. Si tout se passe bien, vous devez avoir 3 LED allumées sur la carte STM32 et vous devez voir un nouveau disque amovible nommé **NODE\_F466RE** dans l'explorateur Windows :

Juvrir Ouvrir	
G v rdinateur VODE_F446RE (E:)	Rechercher dans : NODE_F446 🔎
Organiser 🔻 Nouveau dossier	!≡ ▼ 🗍 🔞
▲ 🚖 Favoris ▲ Nom	Modifié le Type Taille
<ul> <li>Bureau</li> <li>Emplacements récents</li> <li>Téléchargements</li> <li>Bibliothèques</li> </ul>	Aucun élément ne correspond à votre recherche.
Dibliothèques	
> 🔣 Groupe résidentiel	
4 🖳 Ordinateur	
Disque local (C:)	
🛛 😨 NODE_F446RE (E:)	
<u>N</u> om du fichier : mmdvm_f4.hex	✓ Supported Files (*.bin *.hex *.srı ▼     Ou <u>v</u> rir ▼ Annuler

Afficher le bureau de Windows et double-cliquer sur l'icône **STM32 ST-LINK Utility** :



Dans la barre de menus affichée en haut du logiciel ST-LINK Utility, cliquer sur Target puis Connect :



🚟 STM32 ST-LINK Utility File Edit View Target ST-LINK External Loader Help SWV Memory display Device STM32F446xx 0x421 Device ID Address: 0x08000000 0x471F8 Data Width: 32 bits 👻 Size: Revision ID Rev A Flash size 512KBytes Device Memory @ 0x08000000 : File : mmdvm\_f4.hex LiveUpdate Target memory, Address range: [0x08000000 0x080471F8] Address 0 4 8 С ASCII . ... ©Ã..¥Ã..¥Ã.. 0x08000000 20020000 0800C3A5 0800C3A9 0800C3A5 0x08000010 0800C3A5 00000000 ¥Ã..¥Ã..¥Ã..... 0800C3A5 0800C3A5 0x08000020 00000000 00000000 00000000 0800C3A5 .....¥Ã... ¥Ã.....¥Ã...¥Ã.. 0x08000030 0800C3A5 00000000 0800C3A5 0800C3A5 0x08000040 0800C3A5 0800C3A5 0800C3A5 0800C3A5 ¥Ã..¥Ã..¥Ã..¥Ã.. 0x08000050 0800C3A5 0800C3A5 0800C3A5 ¥Ã..¥Ã..¥Ã..¥Ã.. 0800C3A5 ¥Ã..¥Ã..¥Ã..¥Ã.. 0x08000060 0800C3A5 0800C3A5 0800C3A5 0800C3A5 ¥Ã..¥Ã..¥Ã..¥Ã.. 0x08000070 0800C3A5 0800C3A5 0800C3A5 0800C3A5 ¥Ã..¥Ã..¥Ã..¥Ã.. 0800C3A5 0x08000080 0800C3A5 0800C3A5 0800C3A5 ш 18:21:20 : ST-LINK SN : 0668FF515250898367103336 18:21:20 : V2J30M19 18:21:20 : Connected via SWD. 18:21:20 : SWD Frequency = 4,0 MHz. 18:21:20 : Connection mode : Normal. 18:21:20 : Debug in Low Power mode enabled. 18:21:20 : Device ID:0x421 18:21:20 : Device flash Size : 512KBytes 18:21:20 : Device family :STM32F446xx Device ID:0x421 Debug in Low Power mode enabled.

Un affichage de ce type doit être visible à l'écran. Cela signifie que la connexion a été exécutée avec succès :

Cliquer à nouveau sur Target dans la barre de menu du haut, puis sur Program & Verify dans le menu déroulant :

STM32 ST-LINK Utility					
File Edit View	Target ST-LINK External Loader Help				
Memory display	Connect Disconnect CTRL+D				
Address: 0x080	Erase Chip CTRL+E Erase Bank1				
Device Memory @ 0	Erase Bank2				
Target memory, Add	Erase Sectors				
Address	Program				
0x08000000					
0x08000010	Program & Verity CTRL+P				
0x08000020	Blank Check				
0x08000030	Memory Checksum				
0x08000040	Target memory compare with file				
0x08000050	Ontion Bytes CTRL+B				
0x08000060	option bytes Criter b				
0x08000070	MCU Core				
0x08000080	Automatic Mode				
19-21-20 - ST I TNK	Settings				

Une fenêtre s'affiche et vous invite à définir le chemin d'accès au fichier. Cliquer sur **Browse** et naviguer dans l'explorateur de fichiers jusqu'au fichier créé lors de la compilation :

Start address File path	0×08000000	Browse
Extra options	🔲 Skip Flash Erase	Skip Flash Protection verification
Verification	Verify while programming	Verify after programming
Click "Start" to	program target.	

Suvrir						
🚱 🔵 🗢 🔰 « Disque local (C:) 🔸 temp 🔸 MMDVM 🔸 MMDVM-master 🔸 MMDVM-master 🔸 bin 🔹 🍫 Rechercher dans : bin						
Organiser ▼ Nouveau dossier 🗄 🕄 🚺 🔞						
🚖 Favoris	<u>^</u>	Nom	Modifié le	Туре	Taille	
🥅 Bureau		mmdvm_f4.bin	25/04/2021 17:58	Fichier BIN	284 Ko	
Emplacements récents	=	mmdvm_f4.hex	25/04/2021 17:58	Fichier HEX	799 Ko	
🗼 Téléchargements						
🥽 Bibliothèques						
🧊 Bibliothèques						
🤣 Groupe résidentiel						
1토 Ordinateur Misque local (C:) NODE_F446RE (E:)						
Nom du fichier :	mmdvm_f4.h	2X		-	✓ Supported Files (*.bin *.hex *.sri ▼	
Ou <u>v</u> rir ▼ Annuler						nuler

Start address	0x08000000	
File path	N:\Backup\DIFF\SHIBBY\PEF	ROVRADIO VMMDVM VKitC Browse
Extra options	Skip Flash Erase	Skip Flash Protection verification
Verification	Verify while programming	<ul> <li>Verify after programming</li> </ul>
<sup>-</sup> lash memory p	rogramming and verification	
After program	ming	

Valider en cliquant sur **Ouvrir**, puis lancer le chargement du fichier en cliquant sur **Start**.

À la fin de la programmation, le LED la plus proche du connecteur USB de la carte STM32 clignote rapidement en rouge et en vert. L'écran d'accueil du logiciel **ST-LINK Utility** doit afficher des lignes supplémentaires en vert et en mauve dans le journal des événements, en bas :

5 STM32 ST-LINK U	Jtility							
<u>F</u> ile <u>E</u> dit <u>V</u> iew <u>T</u> arget ST-LINK External Loader <u>H</u> elp								
🖴 🖥 🖖 🐳 🤗 🇭 📾								
Memory display Device STM32F446xx								
Address: 0x0800	Address: 0x08000000							
	Revision ID Rev A							
Device Memory @ 0	x08000000 : Fi	le : mmdvm_f4.h	ex			Flash size	512KBytes	🔲 Livel Indate
Target memory, Addr	ess range: [0x08	3000000 0x08047	71F8]					Encopulate
Address	0	4	8	С	ASCII			A
0x08000000	20020000	0800C3A9	0800C3A5	0800C3A5	©/	Ă¥Ã¥Ã.		
0x08000010	0800C3A5	0800C3A5	0800C3A5	00000000	¥Ã¥	Ã¥Ã		
0x08000020	00000000	00000000	00000000	0800C3A5		¥Ã		
0x08000030	0800C3A5	00000000	0800C3A5	0800C3A5	¥Ã	¥Ã¥Ã		
0x08000040	0800C3A5	0800C3A5	0800C3A5	0800C3A5	¥Ã¥	ååÃ		
0x08000050	0800C3A5	0800C3A5	0800C3A5	0800C3A5	¥Ã¥	ååÃ		
0x08000060	0800C3A5	0800C3A5	0800C3A5	0800C3A5	¥Ã¥	ååÃ		
0x08000070	0800C3A5	0800C3A5	0800C3A5	0800C3A5	¥Ã¥	ååÃ		
0x08000080	0800C3A5	0800C3A5	0800C3A5	0800C3A5	¥Ã¥	ååÃ		-
•			1	1	1			Þ
18:21:20 : Device flash Size : 512KBytes         18:24:06 : [mmdvm_f4.hex] opened successfully.         18:29:32 : [mmdvm_f4.hex] checksum : 0x01E27315         18:29:32 : [mmdvm_f4.hex] checksum : 0x01E27315         18:30:13 : Memory programmed in 7s and 784ms.         18:30:13 : VerificationOK								
Debug in Low Power mode enabled. Device ID:0x421 Core State : Live Update Disabled								

Cliquer sur **Target** dans la barre de menu du haut puis sur **Disconnect** dans le menu déroulant :

STM32 ST-LINK Utility		
<u>File Edit View Target</u> ST-LINK External Lo	ader <u>H</u> elp	
🖴 🖬 🖕 🤹 🥔 🐼 📷 -		
Memory display	Device	
Address: 0x08000000 - Size: 0x471F8	Data Width: 32 bits - Revision II	)
Device Memory File - mmdym f4 bey	Flash size	
Device Memory		
18:21:20 : Device family :51/2 : 51/20/yets 18:21:20 : Device family :51/32/E446ax 18:24:06 : [mmdvm_f4.hex] checksum : 0x01E27315 18:29:32 : [mmdvm_f4.hex] opened successfully.		<b>^</b>
18:29:32 : [mmdvm_f4.hex] checksum : 0x01E27315 18:30:13 : Memory programmed in 7s and 784ms		
18:30:13 : VerificationOK 18:30:13 : Programmed memory Checksum: 0x01E2731	15	=
18:33:24 : Disconnected from device.		-
Disconnected	Device ID :	Core State : No Memory Loaded

La carte STM32 peut être déconnectée définitivement du PC, enfin jusqu'à une future mise à jour de son logiciel, qui évolue périodiquement pour corriger des anomalies ou pour ajouter des fonctions. Il suffira de télécharger à nouveau l'archive ZIP mise à jour sur la page Github, d'extraire les fichiers en lieu et place de ceux utilisés précédemment, de compiler ces nouveaux fichiers et de les charger dans la carte STM32. Rien de compliqué maintenant que vous maîtrisez toutes ces étapes !