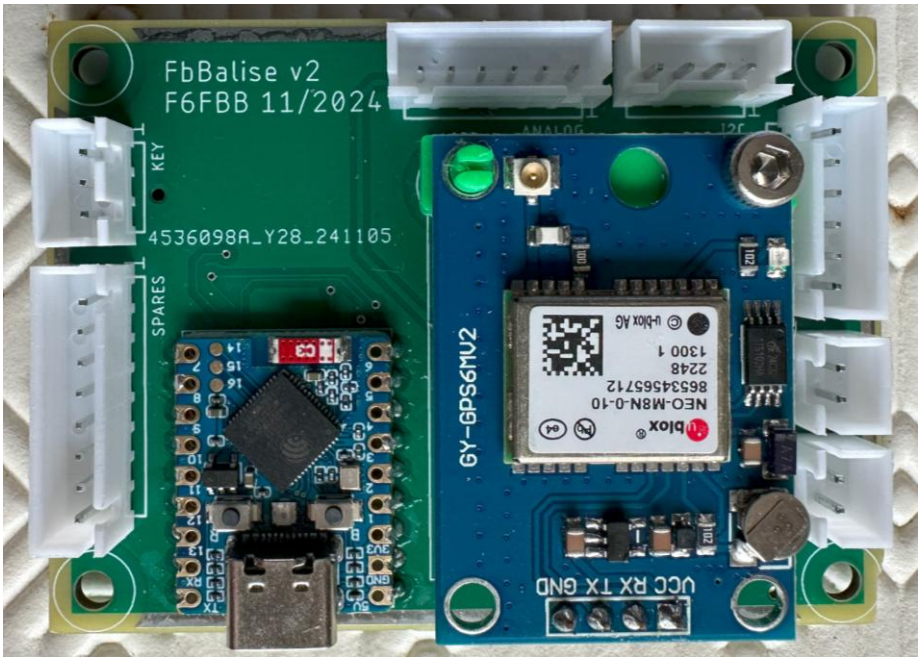
FbBalise – Gestion de Balise

FbBalise est un gestionnaire de balise basé sur un module microcontrôleur ESP32s3.



Il permet de générer des séquences programmées par script et enregistrées en mémoire flash. Une interface WEB permet créer ou modifier les scripts et configurer le système. La taille maximale d'un script est 4K caractères.

L'accès se fait par un ordinateur ou smartphone en se connectant au réseau Wifi « FbBalise » avec le mot de passe « 12345678 » par défaut. Le nom du point d'accès et son mot de passe sont modifiables par configuration.

FbBalise peut également être connecté à un réseau WiFi local.

Caractéristiques principales :

- Interface de configuration par navigateur (chrome, opéra, firefox) via Wifi
- Alimentation 7-18V (moins de 40mA à 14v sans Wifi)
- Modes CW, OPERA, PI4, JT4, JT9, JT65 et WSPR
- Synchronisation du temps et calcul locator par GPS
- Dimensions : (cm) 5.25 x 7.15

Table des matières

1. Démarrage 4
2. Page accueil 5
3. Page Paramètres 6
   3.1. Configuration WiFi 7
      Paramètes WiFi 7
      Option « Délai WiFi » 7
      Activation 7
      Accès par WiFi internet 7
   3.2. Paramètres Généraux 8
      Sortie du signal KEY télégraphie 8
      Inversion signal KEY télégraphie 8
      Script 1 8
      Géné externe 8
   3.3. Configuration SPI 9
   3.4. Mise à jour 10
4. Mots clé du script 11
   4.1. DEBUG 11
   4.2. DELAY 11
   4.3. IF / ELSIF / ELSE / ENDIF 11
   4.4. LOOP / ENDLOOP 12
   4.5. MODE 12
   4.6. RESTART 12
   4.7. SEND 12
   4.8. SET 13
   4.9. TUNE 13
   4.10. TIMESTAMP 13
5. Variables 14
   5.1. Déclaration et utilisation de variables 14
   5.2. Variables prédéfinies 14
   5.3. Formatage d'une variable 15
6. Expressions 17
7. Comparaisons et opérateurs logiques 18
8. Réalisation du script 19
   8.1. Création du script 19

8.2. Script simple 19
8.3. Script plus évolué....19
8.4. Script avec test conditionnel....20
8.5. Script avec synchronisation à la minute 20
9. Générateur externe 21
10. Schéma électronique 22
11. Connecteurs et jumpers....23
12. Interface DDS AD9954 24
13. Remerciements....26

1. Démarrage

FbBalise est configuré par des pages WEB accessibles par Wifi. A la mise sous tension, quelques secondes sont nécessaires pour que le Wifi soit activé.

Depuis votre ordinateur personnel ou votre smartphone, cherchez le réseau Wifi « FbBalise » et connectez-le.

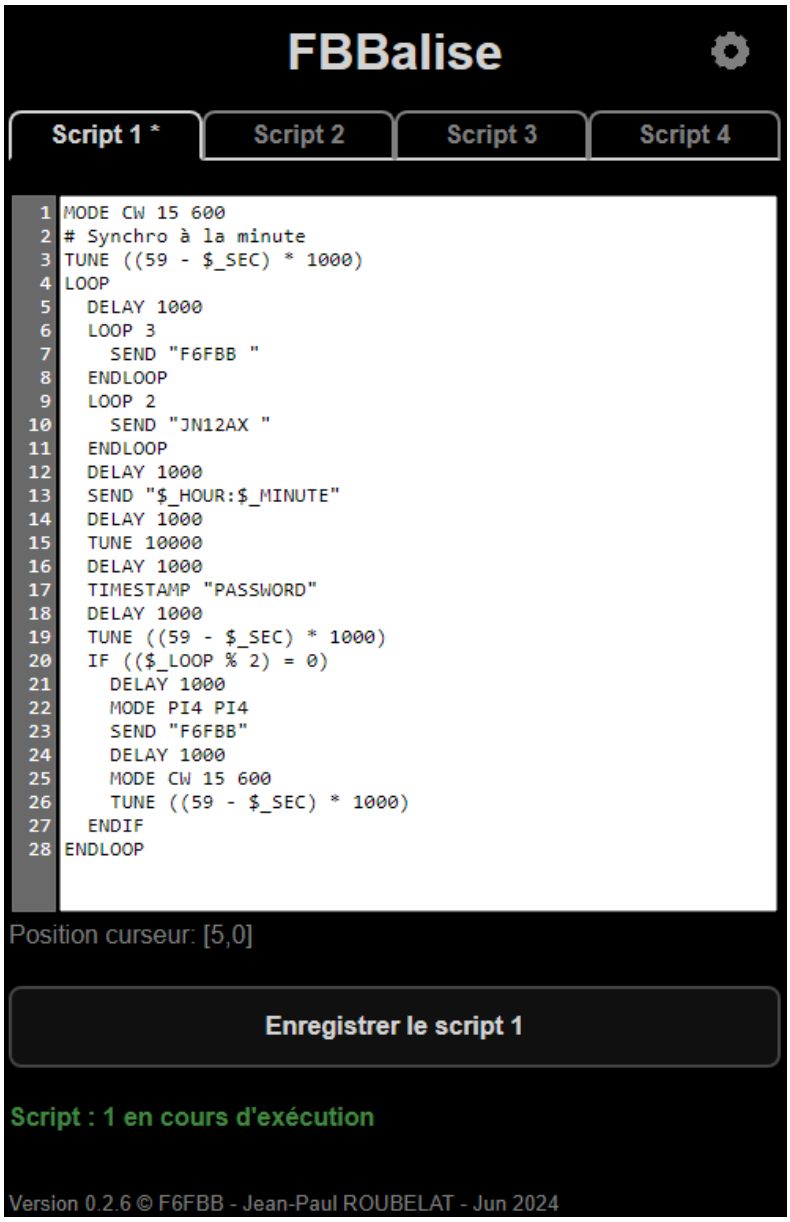
Après la connexion au Wifi « FbBalise », entrez le mot de passe si nécessaire (« 12345678 » par défaut), une fenêtre d'accueil s'affiche au bout de quelques secondes. Cette fenêtre va permettre d'accéder par un menu aux différentes pages de l'application.

Si vous êtes bien connecté au Wifi « FbBalise » mais la fenêtre ne s'affiche pas, vous pouvez la connecter à l'adresse http://192.168.4.1

2. Page accueil

La page d'accueil propose un accès à :

- La configuration par la roue dentée en haut à droite de la fenêtre,
- Quatre onglets permettant de gérer et exécuter quatre scripts différents.
    - Si un nom d'onglet est suivi d'un astérisque, celui-ci a été modifié.
- Une fenêtre texte éditable qui permet de créer ou modifier le script, avec la position du curseur sous la fenêtre.
- Une ligne affichant un éventuel message d'erreur ou une ligne de débug.
- Un bouton permettant de démarrer, redémarrer ou sauvegarder un script suivant le contexte
- Une ligne de status donnant l'état d'exécution du script.



3. Page Paramètres

Un appui sur la roue dentée en haut à droite permet d'aller sur la page des paramètres.

Cette page permet de modifier la configuration par défaut et a quatre onglets :

- Configuration WiFi,
- Paramètres généraux,
- SPI
- mise à jour.



Toute modification devra être enregistrée par le bouton « Validation »

3 boutons en bas de page vous permettent de choisir un type de redémarrage :

- « Redémarrer » : le FbBalise est redémarré, les paramètres sont conservés, aucune modification n'est faite au PWS.
- « Effacer la flash » : les scripts contenus dans la mémoire flash sont effacés . Le FbBalise redémarre et la configuration est conservée.
- « Paramètres usine » : tous les paramètres et scripts sont effacés, le FbBalise redémarre en configuration par défaut.

3.1. Configuration WiFi



Paramètes WiFi

- Nom du réseau WiFi (SSID) pour le point d'accès (AP).
- Mot de passe pour l'accès à ce réseau WiFi. Par défaut « 12345678 ».

Option « Délai WiFi »

- Le Wifi s'active à la mise sous tension de FbBalise.
- Si l'option est active, le Wifi s'arrêtera automatiquement au bout du temps d'inactivité spécifié en minutes.
- Le délai minimal est de 5 minutes. Toute action de changement de page ou de sauvegarde redonne le délai spécifié.
- Si l'option est validée, le point d'accès WiFi ne sera de nouveau accessible qu'après coupure et remise sous tension ou par « activation ».

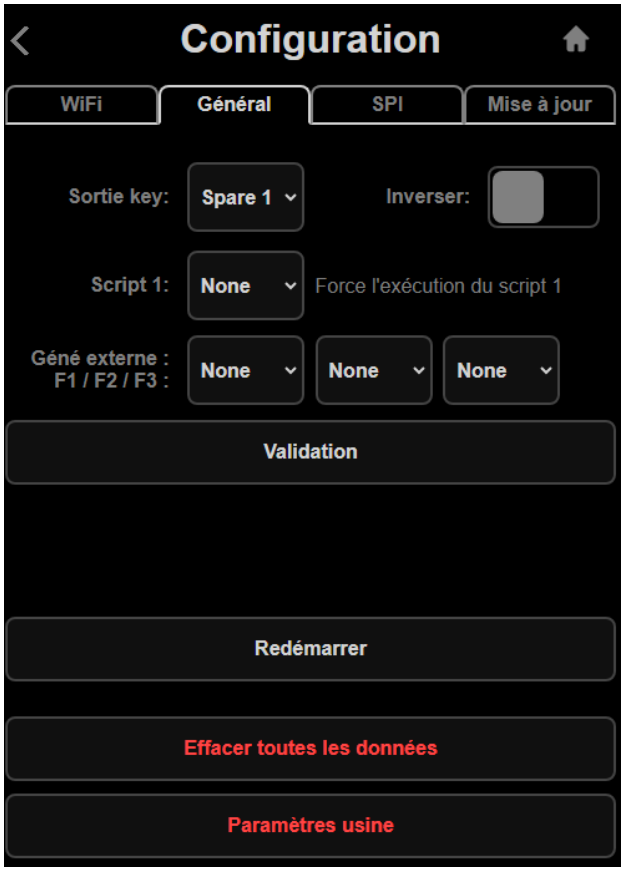
Activation

- Il est possible de dévalider ou valider l'accès WiFi à l'aide d'un niveau 0 ou 3.3v sur l'entrée spare spécifiée.

Accès par WiFi internet

- Si un réseau WiFi et son mot de passe sont spécifiés, FbBalise tentera de se connecter à ce réseau. S'il échoue, il lancera son propre accès WiFi.

3.2. Paramètres Généraux



Sortie du signal KEY télégraphie

- Sélection du port de sortie CW Key (spare 1 à 6)

Inversion signal KEY télégraphie

- Inverse le signal de sortie CW

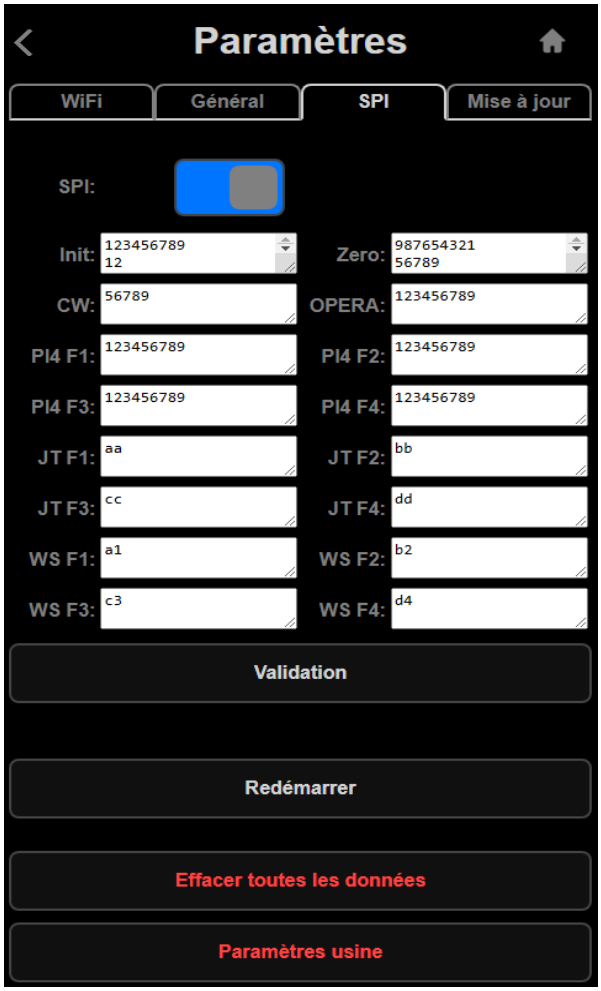
Script 1

- Permet de sélectionner un spare qui verrouillera la balise sur le script 1.

Géné externe

- spécifie trois spares qui donneront la phase utilisée pour la configuration d'un générateur externe. Voir paragraphe Générateur externe.

3.3. Configuration SPI



Le switch « SPI » permet de valider / dévalider l'usage du bus SPI. Ce bus permet de programmer un synthétiseur ou DDS en protocole SPI.

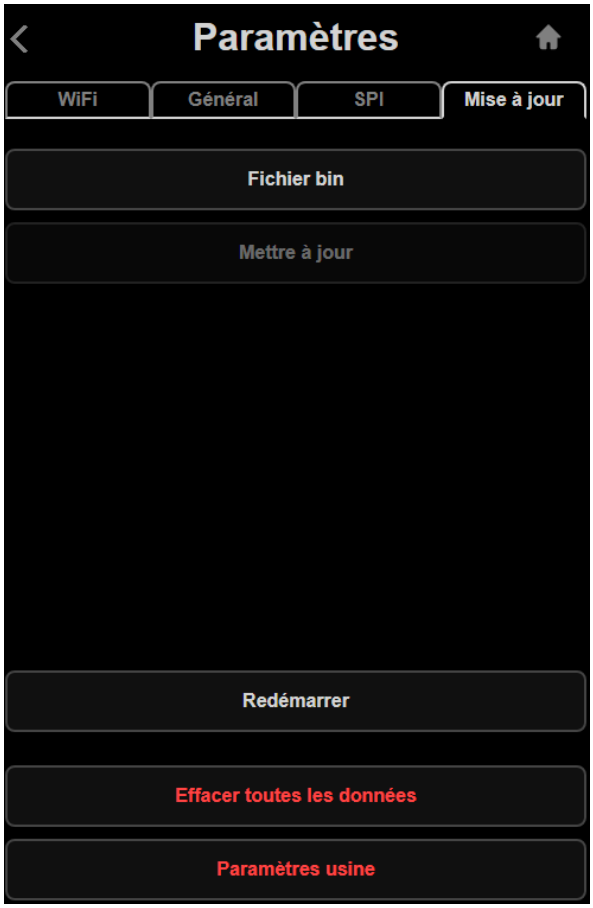
Si le bus SPI est activé, les SPARES 5 et 6 sont réservés :

- SPARE 5 devient OOK si validé et sortie KEY (open drain) si SJ2 est fermé.
- SPARE 5 devient SPI/OSK si validé et si SJ4 est fermé.
- SPARE 6 devient SPI/RST

8 fenêtres texte permettent de définir une séquence hexadécimale envoyée au synthétiseur/DDS lors des différentes phases :

- Init : lors du démarrage du script (configuration synthétiseur/DDS)
- Zero : Lors d'une pause (fréquence porteuse)
- CW : fréquence « CW ON »
- OPERA : fréquence « OPERA ON »
- PI4 F1 à F4: fréquence 1 à 4 en mode PI4
- JT F1 à F4: fréquence 1 à 4 en mode JT4A à JT4G
- WS F1 à F4: fréquence 1 à 4 en mode WSPR

3.4. Mise à jour



Cette page permet de mettre à jour le logiciel avec un nouveau logiciel. Le fichier est de la forme « nom_de_fichier.bin ».

Après avoir saisi le nom de fichier, appuyer sur le bouton « mettre à jour ». Le fichier sera transmis au module balise pour sa mise à jour. En fin de mise à jour, le module sera redémarré.

Ne pas débrancher le module pendant la phase de mise à jour.

4. Mots clé du script

- Un paramètre entre crochets est optionnel
- Une expression peut être
  - une valeur numérique
  - une chaine de caractères entre guillemets
  - une expression entre parenthèses, contenant des expressions, variables, valeurs numériques, parenthèses et signes +, -, /, *, %
  - une condition entre parenthèses, contenant des opérations, variables, valeurs numériques, parenthèses et signes >, >=, <, <=, =

4.1. DEBUG

- DEBUG a un paramètre texte affiché sur la page d'accueil, sous la fenêtre d'édition.
- Utiliser des guillemets pour définir l'argument car il peut contenir des espaces.
- Si l'argument contient une variable, son contenu sera transformé et affiché en texte.
- Une variable peut être formatée cf Formatage d'une variable.

```
DEBUG "TENSION: $($_A2 * 5.12):1 VOLTS UPTIME : $_UPTIME"
```

- Dans cet exemple, on suppose un pont diviseur externe d'un rapport de 5.12. La variable sera toujours affichée avec une décimale.

4.2. DELAY

- DELAY a un paramètre expression spécifiant le délai de pause en millisecondes

```
DELAY (EXPRESSION)
```

4.3. IF / ELSIF / ELSE / ENDIF

- IF ou ELSIF ont un paramètre entre parenthèses définissant la condition.
  - Si la condition est vraie, seule la partie IF/ELSIF sera exécutée, Un seul IF/ELSIF sera exécuté, si aucun IF ou ELSIF n'est vrai, le ELSE sera exécuté.
  - Il peut y avoir plusieurs blocs ELSIF.
  - Si la condition est égale à 0, seule la partie ELSE sera exécutée.
- Un ENDIF doit obligatoirement spécifier la fin du test conditionnel.

```
IF (CONDITION)
      ...
ELSIF (CONDITION)
      ...
ELSE
      ...
ENDIF
```

4.4. LOOP / ENDLOOP

- LOOP peut avoir un paramètre optionnel spécifiant le nombre de boucles
- Si le nombre de boucles n'est pas spécifié, la boucle sera infinie
- Un ENDLOOP doit obligatoirement spécifier la fin de boucle.

```
LOOP [(EXPRESSION)]
     ...
ENDLOOP
```

4.5. MODE

- Mode attend un ou plusieurs paramètres spécifiant les options du mode
  - CW (vitesse en mots / minute) [tonalité]
    - Tonalite 800Hz par défaut
  - OPERA PERIOD [tonalité]
    - PERIOD parmi OPERA05, OPERA1, OPERA2, OPERA4, OPERA8, OPERA32, OPERA65, OPERA2H
    - Tonalite 1500Hz par défaut
  - PI4 Type [tonalité]
    - TYPE parmi PI4, PI4-80, PI4-96, PI4-120
    - Tonalite de base 800Hz par défaut
  - JT Type [tonalité]
    - TYPE parmi JT4A->JT4G, JT65A->JT65C, JT9A->JT9H et WSPR
    - Tonalite de base 800Hz par défaut pour JTxx, 1500Hz pour WSPR

```
MODE CW (EXPRESSION) [(EXPRESSION)]
MODE OPERA OPERA05 [(EXPRESSION)]
MODE PI4 PI4 [(EXPRESSION)]
MODE JT JT4G [(EXPRESSION)]
```

4.6. RESTART

- No parameter requested,
- Restart FbBalise. Configuration and scripts are not lost.

4.7. SEND

- SEND permet d'envoyer un texte passé en argument en utilisant le mode précédemment défini par le mot-clé MODE.
- Utiliser des guillemets pour définir l'argument car il peut contenir des espaces.
- Si l'argument contient une variable, son contenu sera transformé en texte
- Une variable peut être formatée cf Formatage d'une variable.
- Le contenu peut être limité par le mode utilisé.
- Dans le cas de OPERA, seul un indicatif valide sera accepté en argument.
- Dans le cas du JT, l'indicatif et le qth locator peuvent être envoyés, et les messages type 1, 2 ou 3 pour le WSPR.

```
SEND  "F6FBB JN12BT"
SEND  "TENSION $($_A2 * 4.1):1"      # toujours 1 chiffre après la virgule
```

4.8. SET

- SET permet de définir ou redéfinir la valeur d'une variable.
- Une variable doit commencer par le caractère $.
- Une variable est exclusivement numérique.
- Une variable peut être utilisée dans une expression.
- SET a deux paramètres :
    - Un paramètre spécifiant le nom de la variable. Ce nom doit obligatoirement commencer par une consonne et être exclusivement alphanumérique.
    - un paramètre expression spécifiant la valeur numérique affecté à la variable

```
SET $NOMDEVAR ($AUTREVAR / 12.5)
SET $NOMDEVAR 12.5
SET $NOMDEVAR ($NOMDEVAR + 1)        # incrémente $NOMDEVAR
```

4.9. TUNE

- TUNE a un paramètre expression spécifiant le temps du signal en millisecondes

```
TUNE (EXPRESSION)
```

4.10. TIMESTAMP

- TIMESTAMP a un paramètre texte spécifiant le mot de passe utilisé pour le calcul de la clé. Ce mot de passe est **OBLIGATOIREMENT** en majuscules. Il suit la spécification de G4JNT.
- Le mode en cours doit être CW.
- Cette commande ne fonctionne que si le GPS est connecté.
- L'heure utilisée pour le calcul est l'heure GMT fournie par le GPS.
- Si l'heure n'est pas valide (i.e. absence de GPS) le message « TIME ? » sera transmis.

```
TIMESTAMP MOTDEPASSE
```

5. Variables

5.1. Déclaration et utilisation de variables

Les variables sont obligatoirement numériques. Elles sont définies ou redéfines par le mot-clé « SET ». Une variable spéciale **$(EXPRESSION)** n'a pas besoin de déclaration, contient l'expression à évaluer et représente son contenu.

Si la variable est utilisée dans une ligne « SEND », sa valeur sera éventuellement formatée et transformée en texte. Le nombre de variables est limité à 10.

5.2. Variables prédéfinies

Plusieurs variables relatives au système sont prédéfinies. Il s'agit toujours de variables numériques. Leur nom commence toujours par un underscore (tiret bas) excepté pour $(expression). Certaines sont en lecture seule (pas d'initialisation nécessaire), d'autres en écriture seule.

Variable	Valeur	Commentaire	R/W
$_I1 A $_I6	0 ou 1	Etat du spare en entrée 0=LOW 1=HIGH	R
$_O1 A $_O6	0 ou <> 0	Positionne le spare 0=LOW sinon HIGH	W
$_A1 A $_A3	0 à 3,3	Tension en entrée sur l'analog en volts	R
$_PTT	0 ou 1(PTT on)	Positionne la sortie PTT	W
$_HOUR	0 à 23 ou -1	Donnera -1 si le GPS n'est pas actif	R
$_MINUTE	0 à 59 ou -1	Donnera -1 si le GPS n'est pas actif	R
$_SECOND	0.000 à 59.999 ou -1	Donnera -1 si le GPS n'est pas actif. Avec ms	R
$_DAY	1 à 31 ou -1	Donnera -1 si le GPS n'est pas actif	R
$_MONTH	1 à 12 ou -1	Donnera -1 si le GPS n'est pas actif	R
$_YEAR	yyyy ou -1	Donnera -1 si le GPS n'est pas actif	R
$_UPTIME	Nombre d'heures	Nombre d'heures depuis le démarrage (flottant)	R
$_MILLIS	millisecondes	Millisecondes depuis le démarrage du script	R
$_LOOP	Nombre de boucles	Nombre d'itérations dans la boucle active	R
$_POWER	Tension	Donne la tension d'alimentation du module	R
$_TEMPE	Température	Ou -1 si le BME280 non connecté	R
$_HUMID	Humidité	Ou -1 si le BME280 non connecté	R
$_BAROM	Pression barom.	Ou -1 si le BME280 non connecté	R
$_BALTI	Altitude mètres	Correction de pression barométrique.	RW
$_LOCATOR	Qth Locator	Texte uniquement !	R
$_TIMEST[.]	Time stamp	Texte uniquement, Password entre crochets.	R

La variable $_TIMEST[PASSWORD] contient le timestamp au format spécifié par G4JNT, I.E. la minute en 2 chiffres, un espace et 3 lettres. Le password entre crochets est OBLIGATOIREMENT en caractères majuscules.

La variables $_BALTI a la valeur 0 par défaut, donc pas de correction de la pression. La valeur devra être initialisée avec l'altitude pour la correction.

Les variables $_TEMPE, $_HUMID et $_BAROM nécessitent la présence d'un BME280 connecté sur le port I2C. La détection d'adresse est automatique.

⚠ **Les variables $_I1 à $_I6 et $_A1 à $_A3 positionnent le port spare en entrée. Les variables $_O1 à $_O 6 positionnent le port spare en sortie. Si le port spare concerné est utilisé pour un autre usage, cela peut affecter son fonctionnement.**

⚠ **La tension en entrée sur un port spare ne doit pas dépasser 3,3v sous peine de détruire le microcontroleur. Les mesures de tension supérieures peuvent se faire en utilisant un pont diviseur externe. La valeur peut alors être multipliée en utilisant une expression.**

Exemples :

```
SET $_O2 1   # Positionne le port spare 2 au niveau haut

SET $_O2 0   # Positionne le port spare 2 au niveau bas

IF ($_I1)    # Vrai si le port spare 1 est à niveau haut

IF ($_A3 > 2.1)    # Vrai si le port analog 3 est à une tension supérieure à 2,1v
```

5.3. Formatage d'une variable

Lors de l'utilisation d'une variable sous forme textuelle (dans DEBUG ou SEND par exemple), il est possible de formater la variable en utilisant le format suivant :

```
$VAR:N,S
```

- N est le nombre de décimales. La valeur sera arrondie en fonction de ce nombre.
- S est le caractère de séparation qui remplacera le point décimal.
- Le formatage est optionnel, mais devra être utilisé dans cet ordre

Exemples :

```
SET $PI 3.1415926536     # Initialise la variable avec la valeur  3.1415926536

SEND "$PI:3,V"           # Enverra le texte 3V142

SEND "$PI:,V"            # Enverra le texte 3V14159
```

Variables tension d'alimentation et météo pour PI4:

```
SEND "/_PS$_POWER:1,V" # Envoie la tension 12.4v : '/_PS12V4'

SEND "/_HUMI$_HUMI:0"  # Envoie 'hygrométrie 56% : '/_HUMI56'

IF ($_TEMPE >= 0)          # Cas pour +26° et -12°
  SEND "/_TMPO$_TEMPE:0"       # Envoie la température 26° : '/_TMPO26'
ELSE
  SEND "/_TMNE$(-$_TEMPE):0" # Envoie la température -12° : '/_TMNE12'
ENDIF
```

SET $_BALTI 400 # Correction de la pression barom avec l'altitude (400m)
SEND "/_BARO$($_BAROM%100):0" # Envoie la pression 1016hPa : '/_BARO16'

6. Expressions

Une expression est de la forme « A opérateur B » où :

- A est une valeur, une variable ou une expression.
- Un opérateur du tableau ci-dessous,
- B est une valeur, une variable ou une expression.

. Les opérations sont réalisées en virgule flottante.

Les opérateurs sont :

Opérateur	Action	Commentaire
+	Addition	
-	Soustraction	
*	Multiplication	
/	Division	Quotient de la division flottante
%	Modulo	Reste de la division flottante

Les multiplications, divisions et modulo sont effectuées en priorité, de gauche à droite.

Les additions et soustractions sont effectuées en deuxième priorité, également de gauche à droite.

Une expression peut être entourée de parenthèses, auquel cas les expressions entre parenthèses sont effectuées en priorité.

Exemples :

`(3 + 5 * 2)` => 3 + 10 => 13

`((3 + 5) * 2)` => 8 * 2 => 16

`(3 * 5 + 2)` => 15 + 2 => 17

7. Comparaisons et opérateurs logiques

Une comparaison est de la forme « A Comparateur B » où :

- A est une valeur, une variable ou une expression.
- B est une valeur, une variable ou une expression.

Les calculs sont faits en virgule flottante. L'égalité peut donc être difficile à trouver. Le résultat d'une comparaison est égal à 1.0 si le résultat est vrai sinon 0.0 si le résultat est faux.

Les opérateurs logiques fonctionnent également sur des valeurs égales à 0 (0 logique) ou différentes de 0 (1 logique)

Les opérateurs sont :

Comparateur	Action	Commentaire
`<`	Inférieur	
`>`	Supérieur	
`<=`	Inférieur ou égal	
`>=`	Supérieur ou égal	
`!= ou <>`	Différent	
`== ou =`	Egalité	
`&`	Et logique	
`|`	Ou logique	
`^`	Ou exclusif logique	

Exemples :

```
IF ($_A3 > 2.1)     # Vrai si le port analog 3 est à une tension supérieure à 2,1v
```

Exécute si analog3 supérieur à 2.1v

```
ELSE
```

Exécute si le le port analog 3 est à une tension inférieure ou égale à à 2,1v

```
ENDIF
```

```
IF ($_T & 2.1)      # Vrai si la valeur de $_T est différente de 0
```

8. Réalisation du script

8.1. Création du script

Le script est entré au clavier (ou collé) dans la fenêtre d'édition. Lorsque l'édition est terminée, le bouton de sauvegarde va envoyer et sauvegarder le script dans la flash de l'ESP32.

- Le script sera interprété en majuscules et ne fait pas de différences entre majuscules et minuscules.
- Toute ligne devra commencer par un mot clé, suivi de ses arguments éventuels.
- Toute ligne vide est ignorée.
- Toute fin de ligne sera ignorée après le caractère #

8.2. Script simple

Ce script va permettre de :

- Sélectionner le mode CW à 15 mots/minute, son à 800Hz
- Envoyer l'indicatif et le qth locator suivi d'une pause de 1s, tune de 5s, pause de 1s
- Le tout en boucle infinie.

```
MODE CW 15 800
LOOP
      SEND  "F6FBB JN12BT"
      DELAY 1000
      TUNE  5000
      DELAY 1000
ENDLOOP
```

8.3. Script plus évolué

Ce script va générer :

- Un mode CW 15 mots minute à 800Hz,
- trois fois l'indicatif, le qth locator et une pause de 500ms,
- ensuite, pause de 500ms puis tune de 5 secondes puis pause de 1 seconde
- il bouclera indéfiniment…

```
MODE CW 15 800
LOOP
      LOOP 3
            SEND  "F6FBB JN12BT"
            DELAY 500
      ENDLOOP
      DELAY 500
      TUNE  5000
      DELAY 1000
ENDLOOP
```

8.4. Script avec test conditionnel

Ce script va générer :

- Un mode CW 15 mots minute à 800Hz,
- trois fois l'indicatif, une pause de 1s puis deux fois le qth locator,
- ensuite, pause de 500ms puis tune de 5 secondes puis pause de 1 seconde
- Toutes les 4 boucles il enverra l'indicatif en mode OPERA puis retour en mode CW
- il bouclera indéfiniment…

```
MODE CW 15 800
LOOP
      LOOP 3
            SEND "F6FBB "
      ENDLOOP
      DELAY 1000
      LOOP 2
            SEND "JN12BT "
      ENDLOOP
      DELAY 1000
      TUNE  5000
      IF (($_LOOP % 4) = 0)
            DELAY 1000
            MODE OPERA OPERA05 1500
            SEND "F6FBB"
            MODE CW 15 800
      ENDIF
      DELAY 1000
ENDLOOP
```

8.5. Script avec synchronisation à la minute

Ce script va permettre d'envoyer 3 fois l'indicatif suivis d'une pause d'une seconde, puis un tune jusqu'à la seconde 59, faire une pause de 1 seconde puis d'envoyer l'indicatif en mode OPERA . Il nécessite la présence du module GPS pour avoir l'information horaire.

```
MODE CW 15 800
LOOP
      LOOP 3
            SEND "F6FBB "
      ENDLOOP
      DELAY 1000
      TUNE ((59 - $_SEC) * 1000)
      DELAY 1000
      MODE OPERA OPERA05 1500
      SEND "F6FBB"
      DELAY 1000
      MODE CW 15 800
ENDLOOP
```

9. Générateur externe

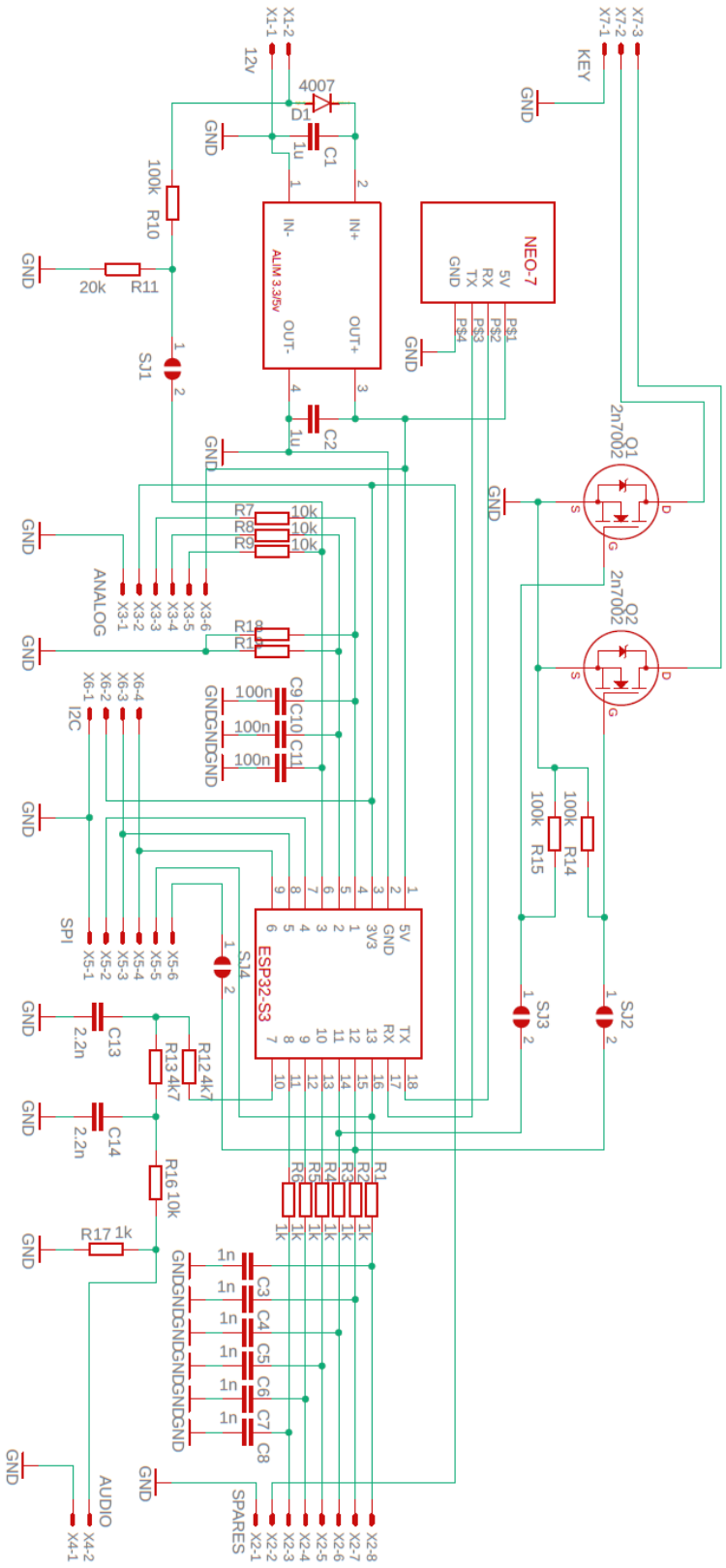
Dans l'onglet « Fréquences » du menu configuration, il est possible de sélectionner 3 spares pour activer un générateur externe OOK ou de fréquences.

Le niveau des spares attribués à F1, F2 et F3 suivra la logique suivante :

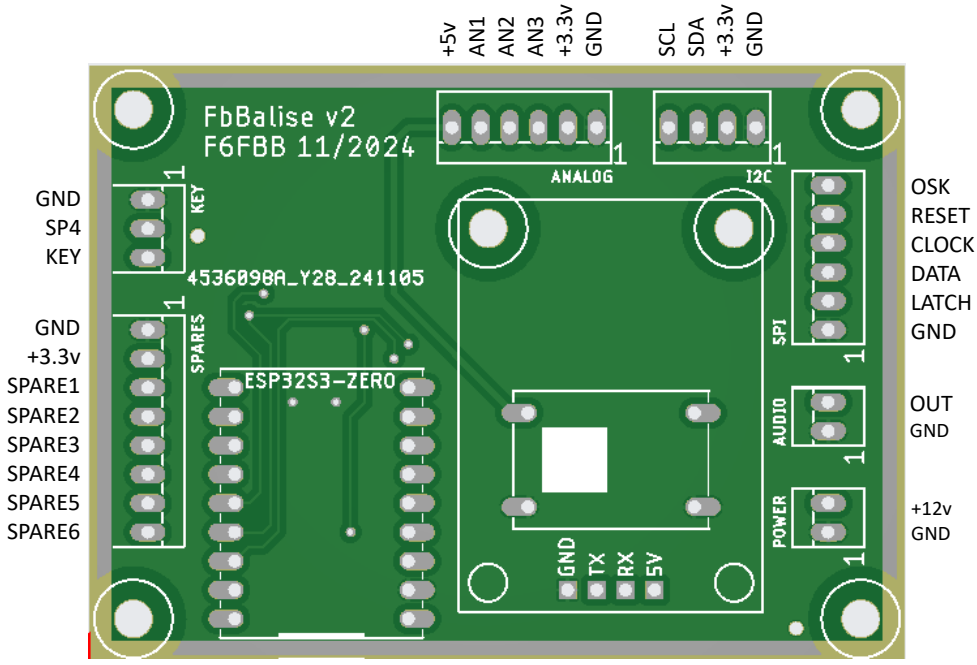
F3	F2	F1	Fréquence à générer	
0	0	0	CW_OFF / SPACE	Génération OOK ou shift suivant hardware
0	0	1	CW_ON / MARK	Génération OOK ou shift suivant hardware
0	1	0	OPERA_OFF / SPACE	Génération OOK ou shift suivant hardware
0	1	1	OPERA_ON / MARK	Génération OOK ou shift suivant hardware
1	0	0	PI4/0	PI4 fréquence 0
1	0	1	PI4/1	PI4 fréquence 1
1	1	0	PI4/2	PI4 fréquence 2
1	1	1	PI4/3	PI4 fréquence 3

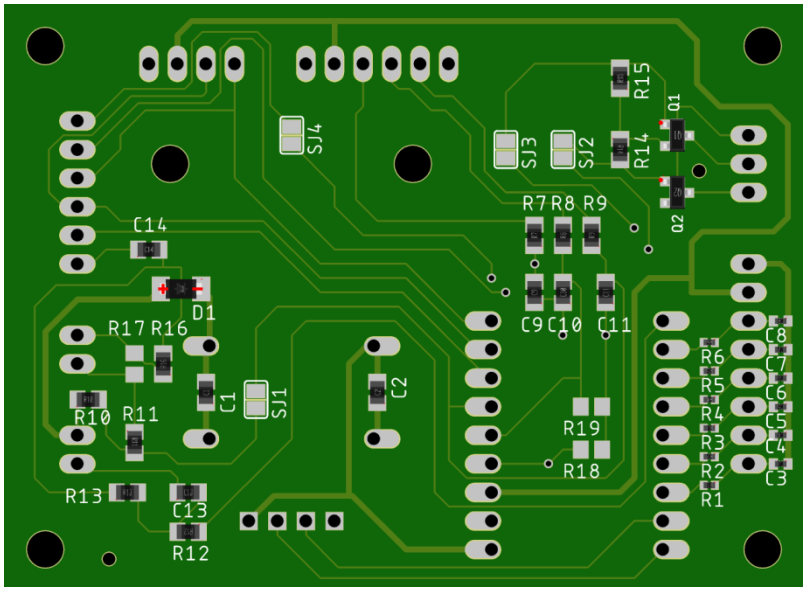
- La sélection des 3 spares n'est pas obligatoire, Il est possible de sélectionner tout ou partie de ces spares.,
    - F1 => « KeyUp / KeyDown. » dans le cas d'une balise CW uniquement
    - F2 => CW /OPERA si PI4 n'est pas utilisé
    - F3 => mode PI4

10. Schéma électronique



11. Connecteurs et jumpers





SJ1 : mesure de la tension d'alimentation sur AN1

SJ2 : sortie OOK open drain

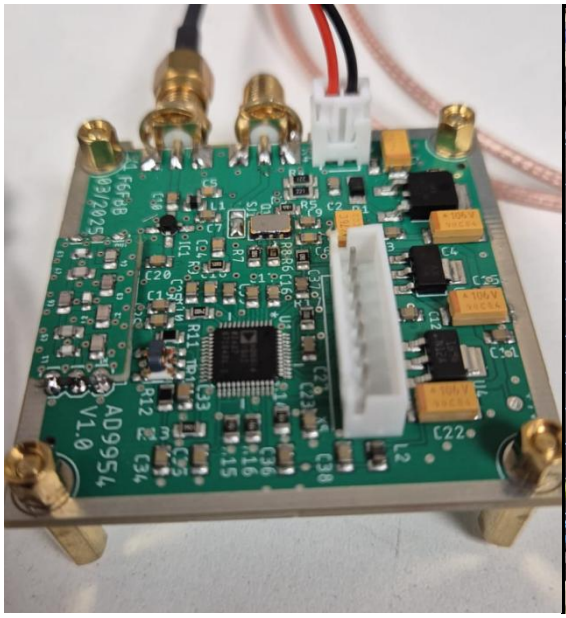
SJ3 : sortie PTT open drain

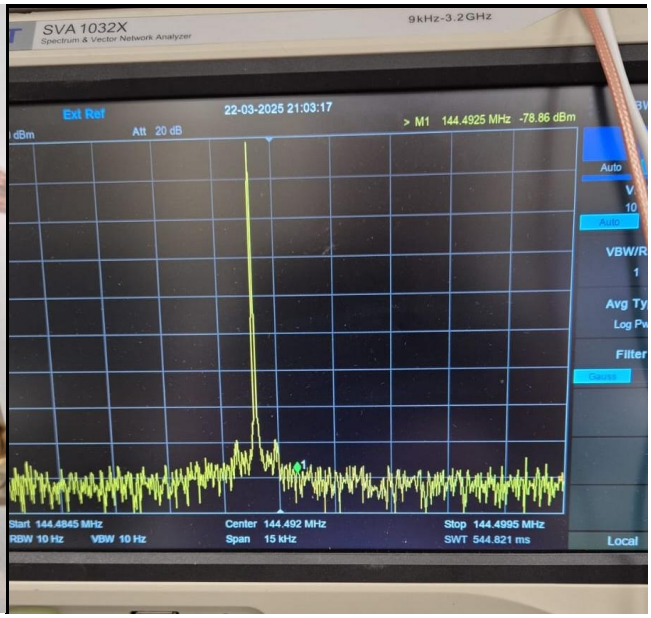
SJ4 : sortie OOK SPI

12. Interface et carte DDS AD9954

Une carte basée sur le circuit AD9954 de Analog Devices peut être directement pilotée via le bus SPI.

Si le switch SPI et le switch AD9954 sont activés, une fenêtre spécifique permet de configurer le circuit et de générer le signal HF en FSK ou OOK suivant les modes et configurations.



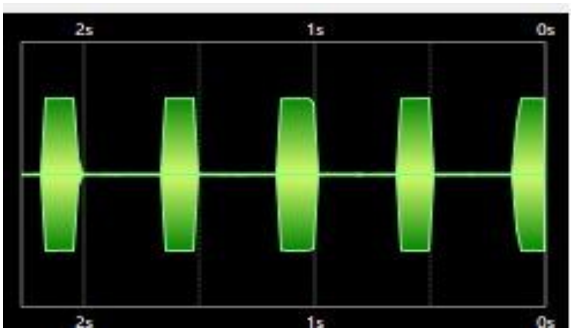




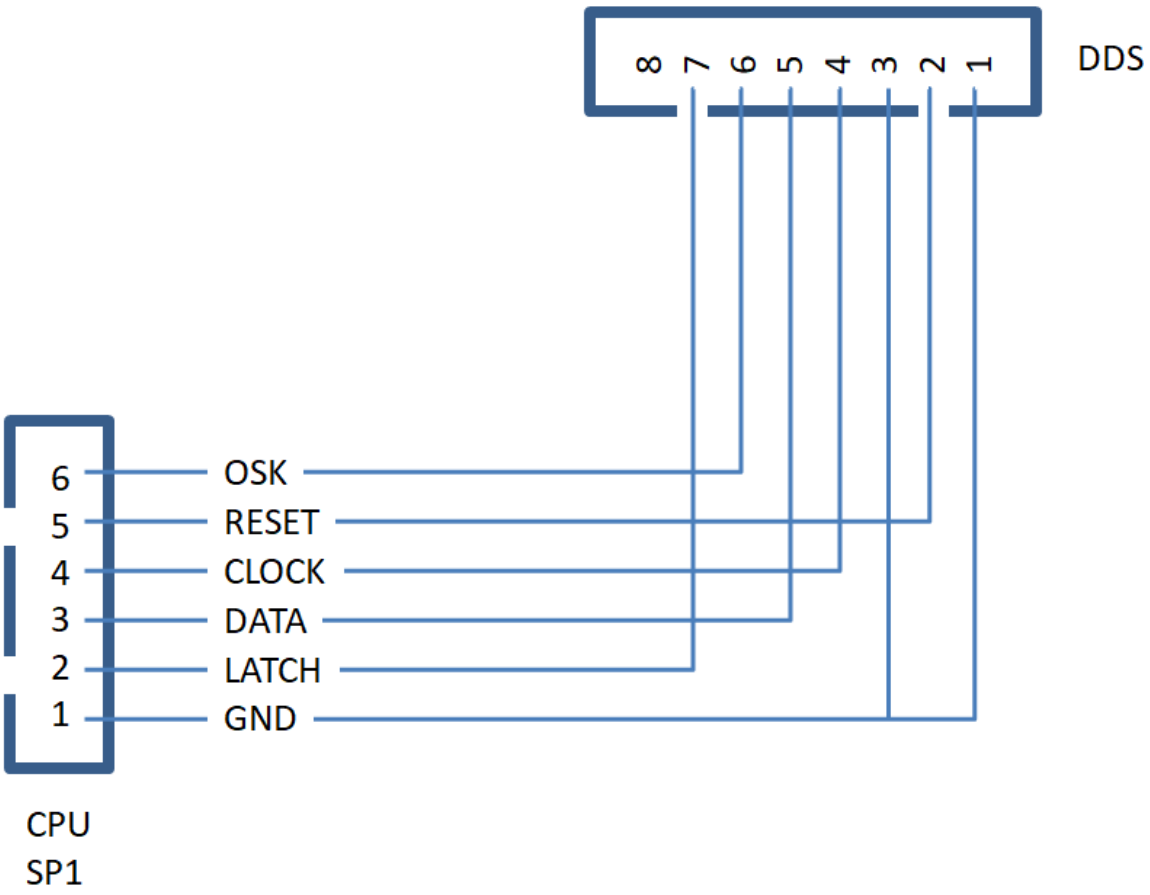
La valeur de la fréquence de base doit être renseignée. La fréquence maximale est de 160 000 000 Hz.

Toutes les valeurs de shift et de fréquences sont exprimées en Hz.

Le multiplicateur permet d'adapter les valeurs de décalage des modes de modulation multifréquences en fonction d'une multiplication de la fréquence générée. I.E. dans le cas d'une génération 144MHz multipliée par 3 pour générer du 432MHz, le décalage de 315Hz en JT4G sera ramené à 105Hz, de façon à retrouver la valeur nominale après les étages de multiplication de l'émetteur. Les autres modes PI4 et WSPR auront le même fonctionnement.



Génération de la modulation OOK avec fronts « anti-claquements »



Câble FbBalise vers connecteur DDS

13. Remerciements

La réalisation logicielle est de ma conception. Ce logiciel fonctionne sur une carte FbBalise et pourra utiliser une carte fille GPS (pour la mise à l'heure)

Le logiciel est écrit en C/C++ sous l'environnement « platformio / vscode » et inclut plusieurs librairies tierces ESP32 .

Merci à F5DJL et F1HDI pour leurs idées, remarques et soutien !

Jean-Paul ROUBELAT.