

Radio club de CAEN

Un début en DATV (Digital Amateur TéléVision)

par Yves, F4HSL

Le projet d'expérimenter une liaison en télévision numérique est né il y a quelques mois, sous l'impulsion de Philippe (F6FZU) déjà bien avancé sur le sujet.



L'enjeu majeur de la DATV amateur est de pouvoir transmettre de la vidéo de qualité sur les bandes amateurs en utilisant le moins de bande passante possible au moindre coût, irréalisable en télévision analogique. Des expériences en cours ont montré la possibilité de transmettre de la qualité VHS avec une largeur de bande d'environ 100 kHz, ce qui ouvrira la possibilité d'utiliser la bande VHF.

L'objectif, atteint à l'heure où j'écris cet article, était d'établir dans un premier temps une liaison vidéo sur la bande 437 MHz entre nos deux QRA (Bernières sur Mer et Courseulles sur Mer) distants de quelques Km, en utilisant au maximum le « home made », et à un coût minimum.

Il a donc fallu dans un premier temps se documenter sur le sujet et force a été de constater qu'il existe une certaine effervescence dans le domaine, et ce depuis pas mal d'années. Il a donc été nécessaire de choisir un angle d'attaque parmi les nombreuses réalisations effectuées ou en cours.

Mais où trouver les informations nécessaires? Comme d'habitude à notre époque sur Internet... Quantité de sites proposent des infos sur le sujet.

Voici quelques références de qualité :

- Le forum VivaDATV.org contient une mine d'informations sur la DATV, les acteurs principaux étant Jean-Pierre (F6DZP), Évariste (F6OEO), Lucien (F1TE), Jean-Yves (F1DJO) et beaucoup d'autres.
- Le groupe Yahoo ATV (il faut être abonné à la liste pour recevoir les infos par mail)
- Le site anglais du BATC (British Amateur Télévision Club)
- Le site de HB9AFO
- Il faut aussi citer l'article de Jean-Pierre paru récemment dans la revue du REF de septembre 2017

La DATV utilise des techniques pour transmettre des informations numériques par radio. Il existe actuellement plusieurs standards, dont le DVB-T, utilisé pour transmettre la télévision numérique terrestre (TNT), et le DVB-S pour transmettre la télévision par satellite. Les informations à transmettre peuvent être de la vidéo, ou de façon plus générale, des données informatiques quelconques (pour transmettre des fichiers par exemple).

Nous nous intéresserons ici à la transmission de la vidéo, c'est à dire d'images animées avec une certaine fluidité.

Le codage du signal vidéo

Un signal vidéo numérique brut (issu d'une caméra par exemple) est composé d'une succession de blocs de pixels codés en binaire (octets), représentant des images se succédant à une certaine cadence (25 images par seconde par exemple). La transmission de ce flux brut représente une très grande quantité d'information, que l'on s'efforcera de réduire par des procédés de compression. La compression MPEG est l'une des techniques utilisées, qui consiste en partie à découper les images en petits carrés, et à ne transmettre que les mouvements de ces carrés entre deux images successives. On réduit ainsi considérablement la quantité d'information nécessaire par un facteur de plusieurs dizaines. Cette opération est effectuée par logiciel, à l'aide de CoDec (Codeur-Décodeur) spécifiques. Différents débits sont alors possibles suivant la qualité d'image désirée et le standard adopté (MPEG2, H264, H265...).

La transmission du signal vidéo encodé sur un canal radio

Une fois le flux vidéo encodé (auquel on peut rajouter du son et d'autres infos) il faut le transmettre sur un canal radio. En prévision d'erreurs de transmission sur ce canal (QRM, QSB...), on va rajouter des blocs de données supplémentaires qui vont servir à corriger ces erreurs durant le décodage. Ces blocs de données sont composés par un algorithme spécifique ; le FEC (Forward Correction Error) exprime le rapport entre les blocs utiles (contenant le flux vidéo par exemple) et le total des blocs transmis.

Le FEC peut ainsi prendre des valeurs comme $1/2$, $3/4$, $7/8$ etc... Un FEC de $1/2$ apportera une meilleure correction d'erreurs qu'un FEC de $7/8$ mais un « rendement » inférieur.

Le tout forme le TS (Transport Stream). Ce TS va être transmis en série, c'est à dire que l'on va convertir les blocs d'octets en une suite de 0 et de 1, transmis les uns après les autres.

La transmission de données sur un canal radio se heurte toujours aux mêmes difficultés ; Il faut transmettre un maximum de données le plus vite possible, avec un minimum de largeur de bande occupée, et la plus forte immunité aux parasites. Comme d'habitude il faudra faire des compromis.

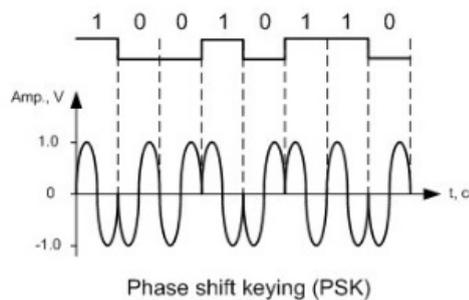
La technique utilisée en DATV DVB-S est la modulation de phase PSK (Phase Shift Keying), et plus particulièrement ses déclinaisons ; la modulation BPSK (Binary Phase Shift Keying) et QPSK (Quadrature Phase Shift Keying).

La modulation BPSK

Comme son nom l'indique, la modulation BPSK consiste à faire varier la phase d'une porteuse en fonction des données binaires à transmettre. On peut ainsi transmettre la

porteuse (une sinusoïde à 437 MHz par exemple) pendant la durée d'un bit 1, et transmettre la même porteuse en opposition de phase pendant la durée d'un bit 0. Dans la pratique on transforme le bit 1 en une tension (1 volt par exemple), et le bit 0 en une tension opposée (-1 volt dans notre exemple). Ensuite on multiplie le signal de la porteuse par 1 ou -1 suivant la valeur du bit à transmettre. Les matheux vous montreront que multiplier un signal sinusoïdal (la porteuse) par -1 revient à inverser sa phase.

Schéma d'un signal BPSK :



La largeur de bande occupée par une telle modulation est celle d'une porteuse modulée en amplitude en double bandes latérales par un signal rectangulaire puisque l'on utilise un multiplieur pour la générer.

Le SR (Symbol Rate) représente le débit des symboles à transmettre, un symbole représente un état de phase de la porteuse. En BPSK ce débit est égal au débit binaire car il y a deux états de phase possible pour la porteuse (0 ou π). Dans notre exemple (un débit binaire de 2 Mb/s) le SR sera donc égal à 2 Ms/s.

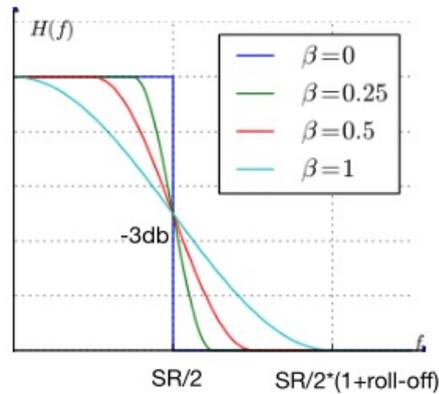
Le spectre de la porteuse modulée en BPSK est ici en théorie infini car le signal rectangulaire modulant contient lui aussi une infinité d'harmoniques. Il est donc impératif de le filtrer.

Le filtrage du signal modulant

C'est un point très important car cela conditionne la largeur de bande occupée du signal HF modulé. Si l'on filtre radicalement (avec un filtre passe-bas) un signal rectangulaire représentant par exemple un débit de 2 Mb/s, de façon à ne garder au minimum que la fondamentale de 1 MHz, la bande occupée sera au mieux de 2 MHz (on a deux bandes latérales). La difficulté (voir la figure ci-dessous) est de réaliser ce filtre passe-bas sans déformer les signaux entrants dans la bande passante du filtre, et en maîtrisant les sur-oscillations au passage des fronts du signal rectangulaire, ce qui permet d'éviter une confusion éventuelle de symboles successifs au moment du décodage. Ce filtre idéal est appelé filtre de Nyquist.

En pratique on va choisir la pente de coupure du filtre (cette pente est quantifiée par un coefficient que l'on appelle le roll-off) de façon à être dans un compromis entre la bande passante et la forme des sur-oscillations (la réponse impulsionnelle du filtre). La fréquence de coupure (la bande passante) du filtre à -3db sera égale à $SR/2$ et la **largeur de bande égale à $SR/2*(1+roll-off)$** .

Schéma de la courbe de réponse du filtre passe-bas de Nyquist pour différents coefficients de roll-off :



Le symbole Bêta en légende représente le coefficient de roll-off

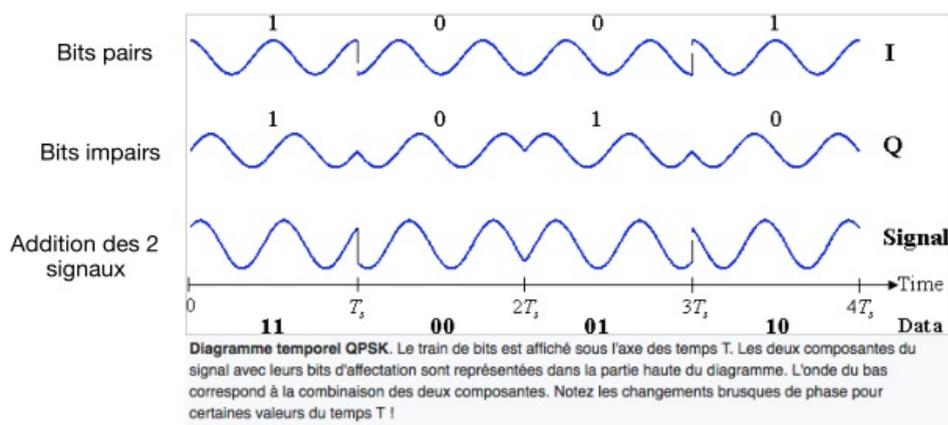
Un coefficient roll-off de 0,35 est un bon compromis. La largeur de bande occupée au minimum par le signal modulé est égale à deux fois la largeur de bande du filtre, puisque nous avons deux bandes latérales c'est à dire à $SR*(1+0,35)$. Rappelons que le SR représente le débit des symboles/seconde. Dans notre exemple précédent de BPSK où le SR est égal à 2 Ms/s la largeur de bande occupée sera donc de 2,7 MHz, si le filtre est bien réalisé...ce qui n'est pas simple à faire, surtout en analogique.

La modulation QPSK (Quadrature Shift Keying)

Nous avons vu précédemment le principe de la modulation BPSK, qui utilise deux états de phase de la porteuse (0 et pi). La modulation QPSK utilise 4 états de phase ($\pi/4$, $3\pi/4$, $5\pi/4$ et $7\pi/4$) de la porteuse, générés à l'aide de deux canaux en parallèle (comme en stéréo). Comme un état de phase correspond à un symbole, on aura ici 4 symboles possibles, représentés par les couples de bits : 11, 00, 01, et 10. L'idée est de scinder en deux le flux binaire à transmettre, bits pairs et bits impairs.

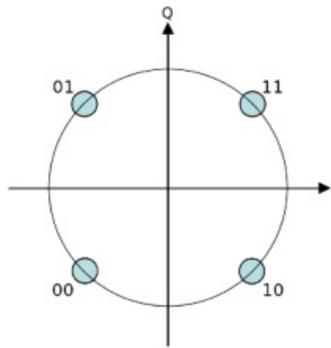
On utilisera pour le premier canal une modulation de porteuse BPSK sur les bits pairs comme décrite plus haut, et pour le deuxième canal, une modulation BPSK sur les bits impairs, mais sur la porteuse déphasée de 90 degrés par rapport à la première (en quadrature). Le premier canal est généralement appelé I (comme In phase) et le deuxième Q (comme Quadrature phase).

En additionnant les deux signaux correspondants on obtient le signal QPSK à 4 états de phase comme le montre le schéma ci-dessous :



Ts représente la durée d'un symbole

On représente souvent le signal sous forme d'une constellation de symboles (ici le signal idéal) :



Quel est l'intérêt de la modulation QPSK ?

La largeur de bande occupée par la modulation PSK ne dépend pas du nombre d'états de phase de la porteuse, mais de la rapidité avec laquelle ils se succèdent (c'est à dire de la fréquence des symboles transmis, le SR).

On voit donc que la QPSK permet de diviser le SR par 2 pour un même débit binaire, puisque chaque symbole représente 2 bits et non 1 comme en BPSK, et donc d'occuper 2 fois moins de largeur de bande. Ainsi pour transmettre un flux binaire en QPSK de 2 Mbit/s, on aura besoin d'un SR de 1 Ms/s et donc théoriquement d'une bande de 1,35 MHz, si on utilise un filtre de Nyquist avec un roll-off de 0,35).

On doit bien sûr y perdre quelque part... hélas oui !, car plus on augmente le nombre d'états de phase de la porteuse, donc de symboles possibles, plus il est difficile de les discriminer à la réception en présence de bruit. Il est tout de même possible d'utiliser des modulations 8PSK (8 états de phase, un symbole représente alors 3 bits), 32 PSK (un symbole représente 5 bits), etc... suivant le bruit du canal de transmission.

Il existe d'autres types de modulation encore plus performants, utilisés en particulier dans les radiocommunications professionnelles, mais c'est un autre sujet (bien passionnant) qui fait appel à des astuces techniques, et à des concepts mathématiques très élaborés qui sortent du cadre du radio-amateurisme.

Une largeur de bande de 1,35 MHz est utilisable sans problème en UHF, en revanche ce n'est pas possible en VHF ou en HF. Il faudra donc diminuer considérablement le débit binaire et le SR pour pouvoir transmettre de la vidéo sur ces bandes, au risque bien entendu de dégrader considérablement la qualité vidéo. Des expériences sont en cours avec des SR très bas et des résultats encourageants.

La réalisation d'un émetteur

Le choix s'est porté sur le projet de F1DJO/F5OEO, pour sa simplicité et son faible coût. Il existe bien entendu d'autres réalisations plus complexes mais plus performantes.

Schéma fonctionnel de l'ensemble retenu :



Le raspi est un module Raspberry Pi 3 fonctionnant sous Linux, équipé d'un écran LCD s'enfichant directement sur le module, et d'une caméra que l'on peut se procurer chez Amazon par exemple. Le modulateur QPSK est conçu par F1DJO. Le logiciel, écrit par

F5OEO, permet d'utiliser l'ensemble comme un émetteur (avec un modulateur QPSK extérieur), ou un récepteur (si l'on connecte une clé SDR adéquate sur l'un des ports USB). Le logiciel complet est téléchargeable sur internet sous forme d'une image, que l'on peut ensuite facilement transférer sur une micro carte SD enfichable dans le raspi et qui remplacera celle d'origine.

On peut télécharger cette image ici :

<http://www.f5oeo.fr/>

Il sera alors possible de télécharger les mises à jour éventuelles en se connectant au raspi via l'utilitaire Windows Putty et la connexion Ethernet du raspi. La même connexion peut servir aussi à télécharger des images de mires à transmettre dans le raspi via l'utilitaire Windows Winscp.

Le menu du logiciel (cf photo de l'émetteur ci-dessous) propose essentiellement :

- De choisir une source de données (caméra ou mires animées)
- De choisir une fréquence de réception (raspi en mode réception avec clé SDR)
- De choisir le SR parmi plusieurs valeurs prédéfinies
- De choisir un FEC (Forward Error Correction)

Le logiciel génère alors directement les signaux binaires I et Q sur le port GPIO du raspi. Ces signaux sont ensuite envoyés au modulateur QPSK (voir le schéma ci-dessus).

Le niveau du signal HF en sortie de l'étage amplificateur du modulateur est de l'ordre de 16 dbm, il est donc nécessaire de l'amplifier si l'on veut une portée suffisante. Des essais concluants ont été effectués avec un ampli linéaire de 1 watt HF sur plusieurs Km, en utilisant des antennes rudimentaires.

Le modulateur utilise un circuit intégré U2790B et un étage d'amplification. Une partie des composants (circuits imprimés, U2790, Quartz pour le 437 MHz) est disponible chez Cimko, le reste se trouve chez Farnell par exemple. L'oscillateur (à l'origine utilisé dans les télécommandes 433 MHz) est disponible sur Ebay.

La documentation complète concernant le projet de F1DJ0 se trouve ici :

(<http://www.vivadatv.org/download/file.php?id=956>)

Schéma fonctionnel du modulateur QPSK :

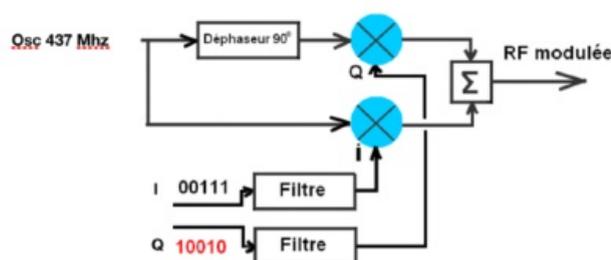
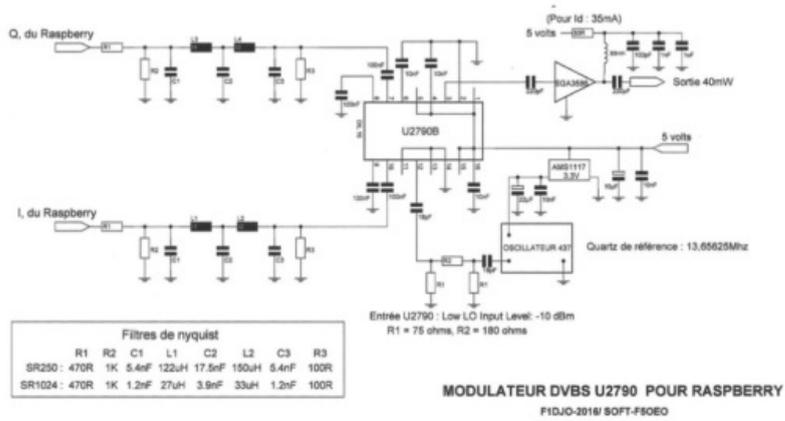
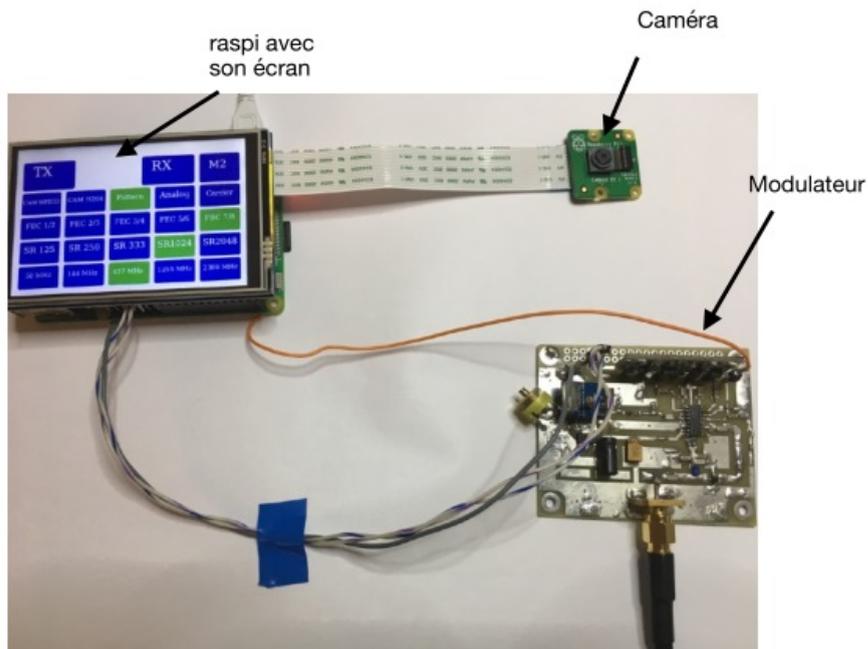


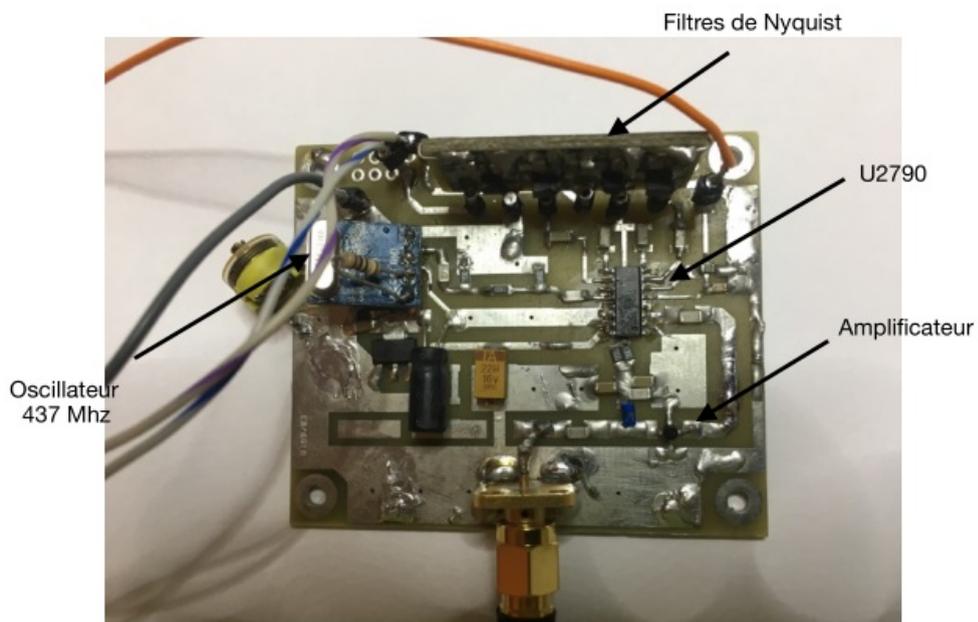
Schéma électrique du modulateur QPSK (avec l'étage amplificateur) :



Réalisation complète de l'émetteur :



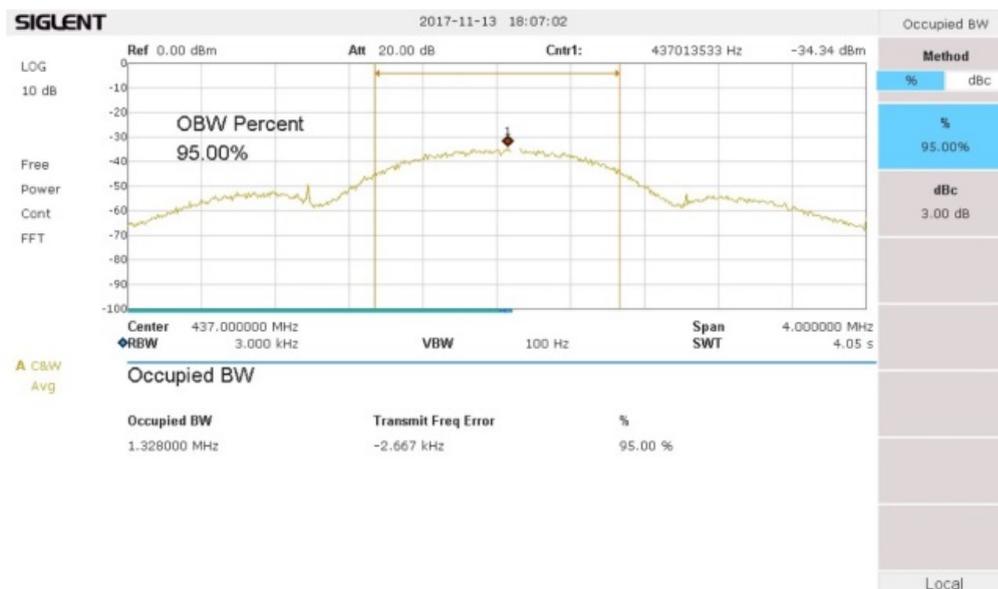
Zoom sur le modulateur :



Une des mires émise par le raspi :



Largeur de bande occupée en sortie du modulateur (pour un SR de 1024 Ms/s) :



On est pas très loin de la largeur idéale...les épaules latérales sont un peu hautes, cela est dû à l'imperfection des filtres de Nyquist utilisés ici pour filtrer les signaux IQ du raspi.

La réception DATV

Principe de la réception QPSK

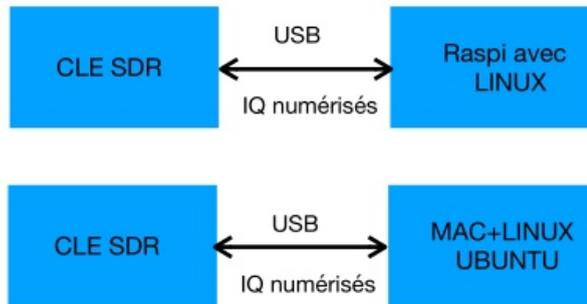
Il est nécessaire de récupérer les signaux I et Q à partir du signal HF modulé. Le principe est le même qu'à l'émission (inversé), on multiplie le signal reçu par la fréquence de la porteuse, ce qui permet de récupérer le canal I après filtrage passe-bas pour éliminer les fréquences images. On fait la même opération avec la porteuse déphasée de $\pi/2$ (en quadrature) pour récupérer le canal Q. Les signaux IQ sont ensuite échantillonnés et disponibles sous forme numérique. La suite du traitement se fait en numérique, par logiciel ou DSP (Digital Signal Processor), pour récupérer les signaux rectangulaires représentant les 1 et les 0 des deux canaux, et reconstituer le flux binaire initial. A partir de ce flux on va ensuite reconstituer le TS, en appliquant la correction d'erreurs éventuelles, extraire l'information vidéo, et la traiter (décompression) pour

l'afficher sur un écran.

La mesure de la qualité de réception se fait à plusieurs niveaux ; on a bien sûr le classique rapport S/N du canal radio, on utilise aussi le BER (Bit Error Rate) qui traduit le rapport entre le nombre de bits erronés et les bits reçus, et le MER (Modulation Error Rate) qui traduit les erreurs de positionnement des symboles sur la constellation (erreurs de phase et d'amplitude).

Trois types de récepteurs ont été testés, deux utilisant une clé SDR relié à un ordinateur avec LINUX (Raspi et MAC avec UBUNTU), et un autre utilisant une carte spécifique (le minitiouner) conçue par Jean-Pierre (F6DZP) connectée à un ordinateur sous Windows.

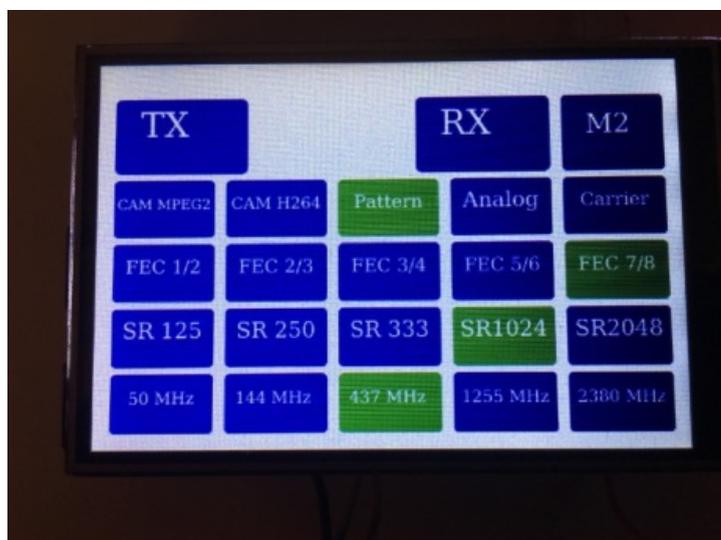
Schéma fonctionnel des récepteurs avec clé SDR :



La clé SDR comprend un circuit tuner et un circuit démodulateur générant les signaux IQ numérisés. Cette dernière fonction doit utiliser le circuit RTL2832U pour pouvoir être programmée par les logiciels utilisés sous LINUX. Ce type de clé, construit initialement pour recevoir la TNT, se trouve facilement pour une vingtaine d'euros.

Une fois numérisés par la clé SDR les signaux IQ sont envoyés à l'ordinateur par une liaison USB. Ils sont ensuite traités par logiciel pour compléter la démodulation QPSK et reconstituer le flux H264 ou MPEG2 selon le codec utilisé à l'émission. Ce flux peut alors être ouvert pour afficher la vidéo en utilisant VLC par exemple.

- Avec le Raspi, le menu du logiciel de F5OEO permet de choisir une fréquence de réception et les différents paramètres QPSK (SR, FEC, Source vidéo...) :



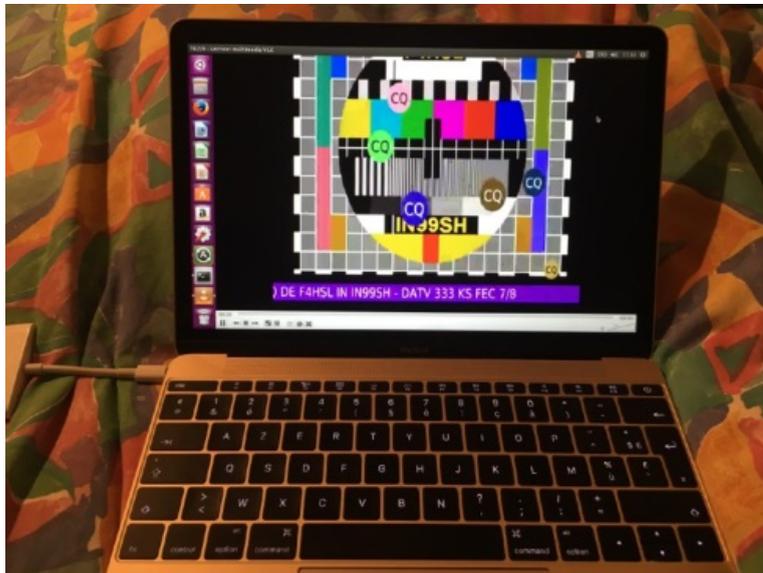
- Avec Ubuntu il faudra utiliser une ligne de commande (c'est plus compliqué mais cela permet des réglages fins) :

```
parallels@ubuntu:~$ rtl_sdr -f 437000000 -s 1500000 -g 15 - | leandrb/src/apps/  
./leandrb --sr 333000 --cr 7/8 -f 1500000 --qui | vlc -
```

On distingue sur la ligne ci-dessus :

- La commande rtl-sdr (rtl-sdr est un logiciel faisant partie du package gnu-radio de Linux) qui va piloter la clé SDR. Ici on fixe la fréquence centrale de réception à 437 MHz (option -f), la cadence d'échantillonnage des signaux IQ donc la bande reçue par la clé à 1,5 MHz (option -s), et le gain de la clé à 15 dB (option -g).
- La commande leandrb (leandrb est un logiciel Linux qui permet la démodulation QPSK on le trouve sur le site pabr.org). On notera l'option qui permet d'afficher à l'écran les résultats de la démodulation QPSK (constellation par exemple).
- Vlc qui permet d'afficher la vidéo dans une fenêtre.

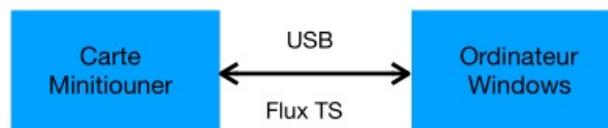
La réception avec un Mac portable (Ubuntu avec Parallel Desktop) :



La réception avec un raspi et une clé SDR :

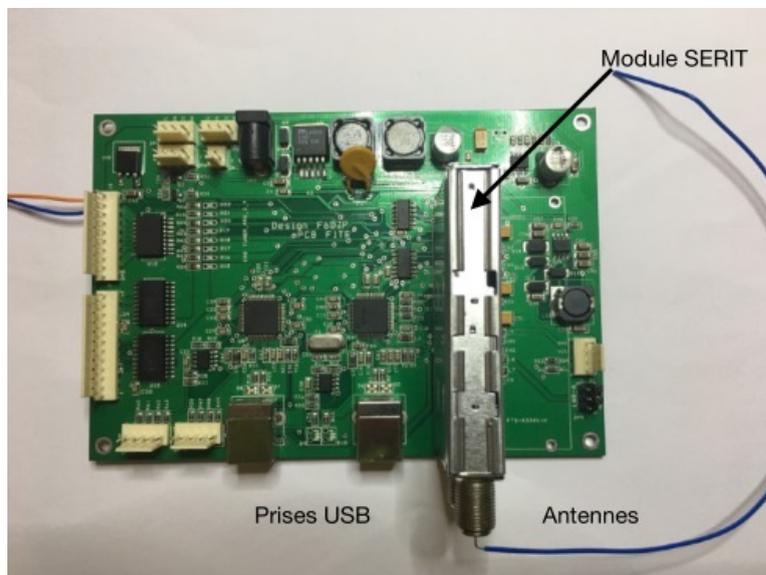


Schéma fonctionnel du récepteur avec le minitiuner :

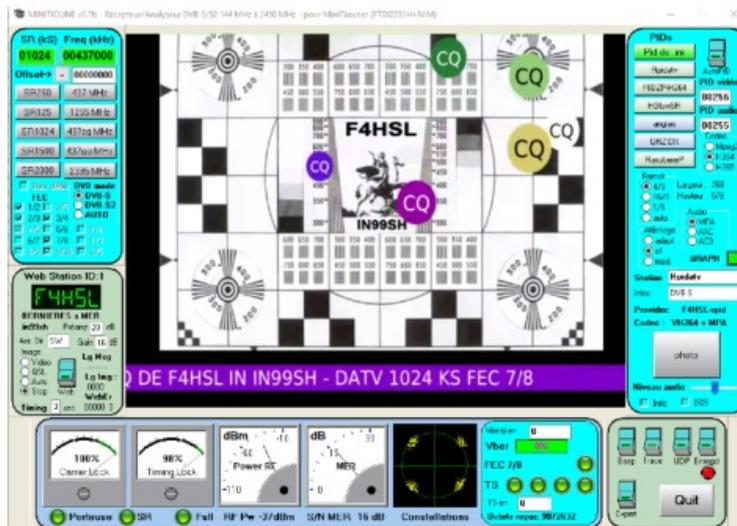


Le minitiuner est une carte comprenant un tuner et un démodulateur complet QPSK (elle remplit donc les fonctions d'une clé SDR et du logiciel leandvb expliquées plus haut). Le module utilisé est un tuner-démodulateur SERIT entièrement programmable par logiciel via un bus série dédié (I2C). Ce module professionnel est utilisé dans les décodeurs TV DVB-S du commerce et peut recevoir deux fréquences indépendamment (deux entrées tuner). La carte communique avec l'ordinateur en USB et délivre le flux TS numérisé par le module SERIT, ainsi que les différentes informations liées à ce flux (MER, taux d'erreurs, niveaux IQ, etc...). Ce flux peut ensuite être décodé et affiché par un logiciel dédié (Tutoune) sur l'ordinateur Windows. Le tout est très bien documenté sur le forum VivaDATV.org.

Le minitiuner tel qu'on peut le commander sur le site du REF :

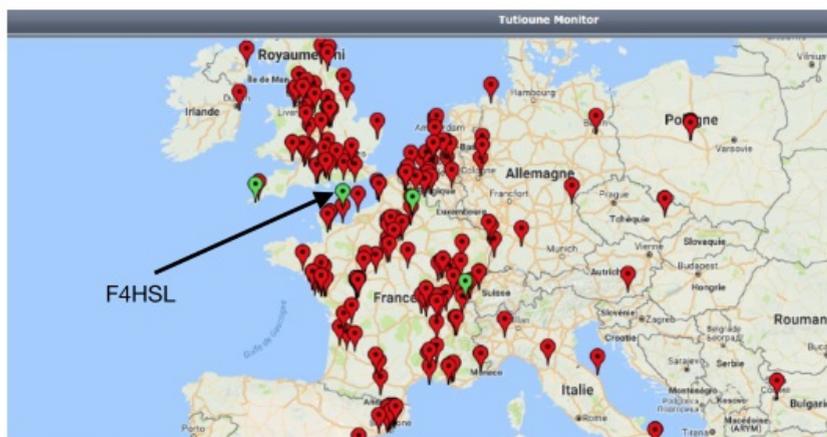


L'écran du logiciel du minitiouner (sous Windows10 sur un Mac avec Parallel Desktop) :



Cette copie d'écran montre la réception de la mire émise par mon émetteur raspi. Le logiciel Tutuione pilote la carte minitiouner et permet de choisir en particulier la fréquence de réception et le SR. Le type de codage vidéo (MPEG, H264...) est en principe trouvé automatiquement avec les PID (Packet Identifier) du flux TS. Le logiciel propose différentes mesures, dont le niveau de signal reçu, le MER, etc.... On a aussi la possibilité d'envoyer ces informations, et partiellement la réception vidéo, sur un site Web qui permet de voir en temps réel les stations connectées en DATV. Ce site est accessible à partir de l'écran d'accueil du forum VivaDATV.org.

Le site TutuioneMonitor (sur Internet) :



Conclusion

Tout ceci est une première étape réalisée avec pas mal de difficultés, en particulier dues à mon inexpérience dans la soudure des composants CMS... (merci à Philippe, F6FZU, pour son aide et sa disponibilité), mais aussi à des problèmes logiciels avec Ubuntu et Windows dus à ma configuration matérielle et logicielle (Mac). Les étapes suivantes seront les mises en boîtiers, les tests de transmission du son en utilisant par exemple une caméra sonore, et bien sûr la réalisation de liaisons à plus grande distance avec des OM QRV en DATV, ce qui nécessitera d'utiliser des antennes UHF performantes, que je ne possède pas actuellement.

Il peut être aussi envisagé d'améliorer l'émetteur, en particulier le filtrage de Nyquist pour diminuer la largeur de bande occupée, et aussi d'expérimenter d'autres types de modulation plus économes en largeur de bande (comme la 8PSK utilisée en DVB-S2).

73 d'Yves, F4HSL

[Retour vers l'accueil](#)

Mis à jour le 04/12/2017

Xiti