

Traduction Google du fichier BATC de M0EYT « Custom DATV Firmware for the Pluto »

Avant-propos

Ceci est un aperçu d'un article destiné à être publié dans CQ-TV 266 afin de promouvoir les premières expérimentations.

Merci à **Paul M0EYT** pour l'avoir écrit.

Avant de charger le micrologiciel personnalisé, il peut s'avérer nécessaire d'étendre la plage de fréquences du Pluto. Les instructions à ce sujet se trouvent sur cette page Web:

<https://wiki.analog.com/university/tools/pluto/users/customizing>.

Faites défiler l'écran vers le bas jusqu'à la section "Mise à jour de l'AD9364".

Vous voudrez peut-être également activer le deuxième cœur de l'UC, comme décrit ici:

<https://www.ph4x.com/pluto-sdr-hack-2nd-cpu-core/>

Vous pouvez télécharger le firmware à partir du forum VIVA DATV, après l'enregistrement, à l'adresse <http://www.vivadatv.org/viewtopic.php?f=60&t=692>.

Introduction

Tout d'abord, je ne suis pas un expert en DATV, mais j'ai joué avec un certain nombre de DTS au cours des années qui se sont écoulées depuis le début de ma première SDR-14 en 2005. J'ai touché au GHz pour avoir une idée approximative de ce que je fais mais c'est tout! J'aime m'assurer d'apprendre chaque jour quelque chose de nouveau en rapport avec la technologie. DATV et le PLUTO SDR s'en chargent. J'ai eu le PLUTO après en avoir emprunté un à Jules, GONZO et en être très impressionné, car son prix est irréprochable. Il fonctionne hors de la boîte, a TX + RX et avec quelques minutes de travail, sa gamme de fréquences peut être étendue pour couvrir la plage de 70 MHz à 6 GHz, pas mal du tout, en plus, il tourne Linux en interne (Linux plutôt 4.14.0-g387d584d434e). Le récepteur est génial et j'ai utilisé l'unité de prêt pour écouter toutes les balises Bell Hill jusqu'à 5,7 GHz avec un dispositif semblable à un trombone enfoncé dans le port RX.



C'est l'unité en question, illustrée ci-dessus, très compacte à 5 "X3" X1 "avec des prises SMA séparées pour la transmission et la réception et une paire de prises micro USB pour les E/S et l'alimentation. Elle consomme de 400 à 420 mA en mode veille. Lorsque le transpondeur à bande étroite QO100 était utilisé avec des PLUTO SDR pour la transmission, il y avait une certaine dérive de fréquence: il apparaît que le TCXO intégral n'est pas particulièrement performant, il est donc intéressant de le remplacer par un TCXO décent tel que ASTX. 13-C-40.000MHz-I05-T que vous pouvez obtenir auprès de Mouser, ce qui résout tous les problèmes de stabilité de fréquence (enfin la plupart ...). Pour une stabilité optimale, une référence externe à GPSDO est toujours recommandée.

Premiers pas

J'ai construit le filtre / modulateur Portsdown pour DVBS qui fonctionne très bien, mais je recherchais une méthode non-Lime-SDR (c'est une longue histoire) pour générer DVBS28PSK et 16 / 32APSK DATV. J'ai vu sur Twitter que OM Evariste, F5OEO (@ F5OEOEvariste) de la renommée de RPiTX était en train de développer un micrologiciel de remplacement pour le PLUTO, qui fournirait des fonctionnalités DATV ainsi que d'autres bonus. firmware. Le firmware est arrivé rapidement. Le fichier 'pluto.frm' est copié dans le répertoire racine de la mémoire de stockage de masse PLUTO, où se trouvent les fichiers de configuration. Une fois copié, éjectez le périphérique, ne débranchez pas l'USB mais utilisez l'éjection du logiciel. La LED bleue 1 du PLUTO clignote

rapidement pendant 3 ou 4 minutes, puis redémarre et le périphérique de stockage de masse réapparaît. Pour plus de détails sur ce processus, consultez :

<https://wiki.analog.com/university/tools/pluto/users/firmware>

- ne le débranchez pas pendant le processus de flash pour des raisons évidentes; vous pourriez le détériorer. Le SDR étant toujours branché sur votre PC, vous pouvez naviguer vers le serveur Web interne en pointant un navigateur sur <http://192.168.2.1>. Tout devrait bien se passer, vous devriez voir quelque chose de similaire à la capture d'écran suivante:



Après avoir mis à jour le micrologiciel et confirmé son fonctionnement, nous avons ensuite entamé une courbe d'apprentissage très lourde! De quoi aurais-je besoin pour générer un «flux numérique» contenant de la vidéo? Quel logiciel devrait être utilisé? Comment le flux passe-t-il du PC au SDR? Comment définir tous les paramètres nécessaires à la génération de DATV? Comment puis-je transférer des vidéos de la caméra de mon téléphone vers le PC? La liste des questions s'allongeait à mesure que j'examinais.

Je connais John, GI7UGV, qui travaille dans le même secteur et qui s'intéresse beaucoup à la DATV. J'ai donc eu une discussion avec lui et quelques minutes plus tard, VMIX (<https://www.vmix.com/>) a été installé. La méthode initiale la plus simple pour faire ce dont j'avais besoin; faire un PC générer un flux pour contrôler le PLUTO. C'était plutôt intuitif et au bout d'une demi-heure, j'avais une source de carte de test, qui envoyait les données pertinentes vers PLUTO. Cela a été effectué en définissant une cible de flux RTMP externe avec les paramètres suivants:

URL : `rtmp://192.168.2.1:7272/,437,DVBS2,QPSK,333,23,Pass : ,M0EYT,`

Les paramètres ci-dessus font partie de l'URL et sont analysés par le micrologiciel F50EO pour définir les différents paramètres de transmission DATV:

Frequency in MHz: 437

Mode (DVBS/DVBS2): DVBS2

Constellation (QPSK,8PSK,16APSK): QPSK (only QPSK is valid in DVBS)

SymbolRate in KS (33-2000): 333

FEC (12,23,34,67,78...): 23

CALLSIGN: M0EYT

Il est particulièrement important d'examiner la syntaxe de définition du flux RTMP. Il est probablement préférable de copier et coller l'URL / texte de passage ci-dessus, puis de le modifier pour répondre à vos propres exigences. Avec le PLUTO SDR branché sur le port USB du PC sous VMIX, cela fonctionnait immédiatement et un opérateur QPSK était généré en 437 MHz, pouvant être reçu sur le Minitiouner. J'avais remarqué que VMIX n'était pas gratuit, alors plutôt que de passer des heures avec IDA, et après avoir noté ce que John 'UGV avait dit, j'ai décidé de désinstaller VMIX et d'essayer OBS (Open Broadcaster Software

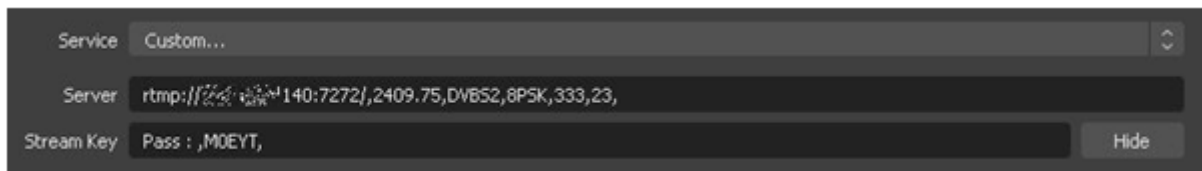
<https://obsproject.com/download> .

Comme il s'agit d'une source ouverte, il n'y a pas de problème de licence et bien que ce ne soit

pas aussi perfectionné que VMIX, il est entièrement fonctionnel et fonctionne parfaitement.

OBS Basics

Dans OBS, la première chose à faire est de définir le flux de sortie afin qu'il pointe vers la carte SDR de PLUTO. Allez donc dans Paramètres, Flux et tapez ce qui suit; évidemment, vous adaptez l'adresse IP, les paramètres de modulation et l'indicatif à votre environnement:



Vous pourrez voir une boîte de dialogue "Commandes" ancrée au bas de la fenêtre OBS. C'est à cet endroit que vous appuyez sur "démarrer la diffusion" pour activer la sortie DATV de PLUTO. Un bloc vert doit apparaître dans la barre d'état pour indiquer que la diffusion vers PLUTO est en cours.

Avant de vous lancer et d'appuyer sur «démarrer la diffusion», vous devez définir le débit en flux continu pour éviter tout débordement entre OBS et PLUTO.

Rendez-vous sur :

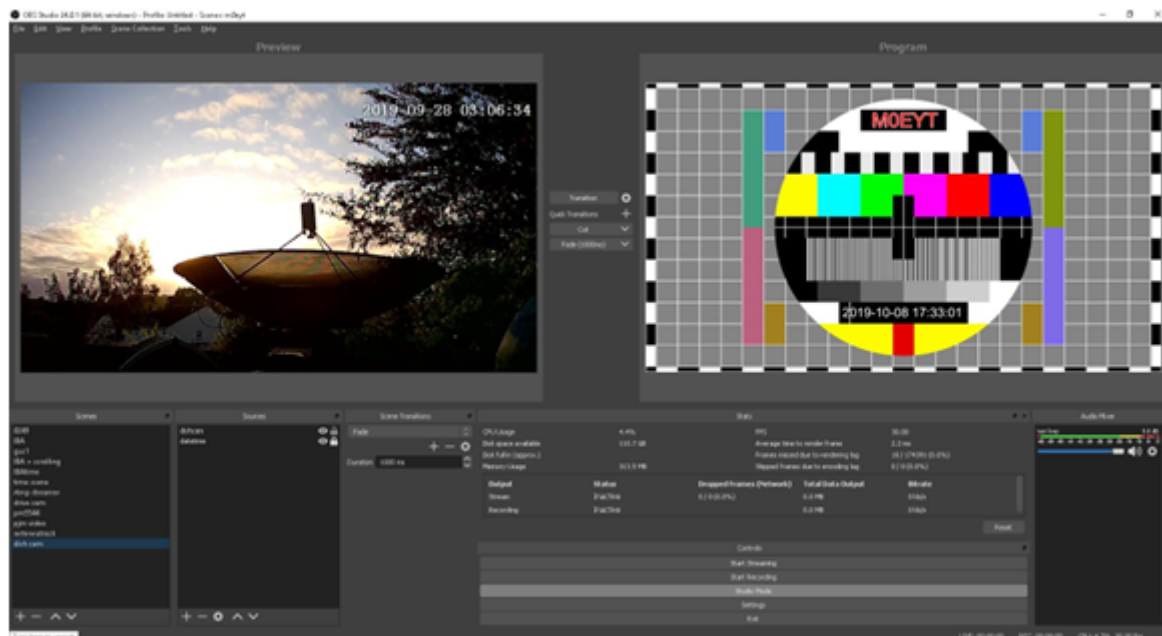
http://www.satbroadcasts.com/DVB-S_Bitrate_and_Bandwidth_Calculator.html,

saisissez vos paramètres DVBS/S2, appuyez sur calculer, puis notez le débit «Netto TS bitrate»: vous souhaitez définir le débit binaire de votre flux à environ 65 % à 70% de ce chiffre. Par conséquent, si le débit binaire Net TS est de 440 kbps, vous souhaitez définir le débit binaire de votre vidéo sur 286 kbps. Cela signifie que la surcharge de la vidéo et du transport ne causera pas de débordement lors de la diffusion de données dans PLUTO. Une fois que vous êtes familiarisé avec les différents débits et vos réglages favoris, vous pourrez jauger le débit de la vidéo dans OBS. Il est défini via "paramètres", puis "sortie", puis dans la section "diffusion en continu", tapez votre débit. Mon encodeur est réglé sur la compression x264 et mon débit audio sur 64Kbps. Avec ces paramètres, il n'y a aucune interruption dans le flux audio et tout fonctionne de manière fluide.

Ensuite, vous aurez besoin d'une source d'image. La méthode la plus simple dans OBS consiste alors à vous rendre dans le dock «scènes», à appuyer sur +, à entrer un nom pour votre scène, tel que «carte de test». Ensuite, dans le dock "sources", appuyez sur +, sélectionnez le périphérique de capture vidéo, créez-en un nouveau, tapez un nom et appuyez sur OK. Une carte de test 'Barre de couleur / Fondu gris / Barre' devrait apparaître dans la fenêtre 'Aperçu'. Assurez-vous que dans le dock «Contrôles», vous avez appuyé sur «Studio Mode» pour afficher les fenêtres Aperçu et Programme. Ce que vous voyez dans la fenêtre "Programme" correspond à la vidéo en streaming sur votre PLUTO.

Vous pouvez définir un certain nombre de «scènes» afin de pouvoir les sélectionner, les fondre ou les couper rapidement. Si vous avez des fichiers vidéo de bureau, ceux-ci sont faciles à ajouter. Vous pouvez créer une scène supplémentaire et y insérer par exemple une image JPG / PNG ou ajouter une vidéo de bureau. J'ai constaté que mon caméscope avait vidé sa vidéo dans un fichier .VRO, mais personne n'en a jamais entendu parler, mais OBS pourrait l'ingérer et la diffuser correctement, y compris les pistes audio stéréo.

Vous pouvez également ajouter facilement des messages texte défilants pour superposer vos images, diverses horloges analogiques et numériques, des entrées depuis des webcams, des caméras de vidéosurveillance RTSP, des surimpressions de poulet / chat dansant, etc. Il existe de nombreux choix. Vous voudrez probablement passer quelques heures à parcourir les différents menus pour avoir une idée des options du logiciel et de ce qu'il peut faire. Je trouvais assez facile de configurer des sources et de pouvoir couper et modifier les paramètres tout en regardant le flux DVB-S2 sur un autre ordinateur portable. Dans OBS, il est également intéressant de regarder les différentes extensions/add-ons écrites par d'autres pour la plate-forme. Il s'agit essentiellement de fonctionnalités supplémentaires que vous pouvez utiliser. Vous allez vous retrouver avec quelque chose de similaire à mon exemple:



Dans mes scènes OBS, j'ai une entrée 'streamer rtmp', cela me permet d'utiliser la caméra / le microphone de mon téléphone mobile Android, avec un logiciel appelé Larix Broadcaster

https://play.google.com/store/apps/details?com.wmspanel.larix_broadcaster

- cela permet de diffuser la vidéo à partir du téléphone, mais vous ne pouvez pas l'intégrer directement dans OBS, car vous avez besoin d'un serveur de diffusion RTMP.

Vous pouvez utiliser cette application mobile pour diffuser directement sur PLUTO SDR, mais toutes les fonctionnalités de traitement vidéo de OBS sont alors perdues. Heureusement, il existe un fichier à

<https://forum.batc.org.uk/viewtopic.php?f=69&t=6179>

détaillant ce qu'il faut faire pour créer un tel serveur, vous pouvez le déposer sur l'un de vos PI Raspberry. et cela consomme très peu de CPU. Fondamentalement, il utilise le serveur HTTP NGINX avec un plug-in de streamer RTMP et fonctionne simplement. Pointez votre téléphone et OBS sur l'IP du PI avec le port défini dans le fichier de configuration, appuyez sur les différents boutons Go et la vidéo/audio sera diffusée du téléphone vers OBS. Si vous pouvez vous passer de toute source vidéo analogique dans votre configuration, votre sortie de flux globale sera numérique, du capteur à l'affichage à l'autre bout de votre QSO. Cela signifie que la qualité sera maintenue et que le son sera de qualité médiocre avec des boucles terrestres/bourdonnements ou des vidéos présentant des artefacts analogiques typiques.

Si vous souhaitez diffuser depuis le bureau Windows sur le serveur RTMP, vous pouvez également le configurer facilement. Tout d'abord, téléchargez FFMpeg à l'adresse :

<https://ffmpeg.zeranoe.com/builds/>.

Vous aurez ensuite besoin du lecteur de capture que vous pouvez obtenir à l'adresse :

<http://www.umediaserver.net/components/index.html>.

Recherchez UScreenCapture et téléchargez-le. version appropriée. Une fois les deux packages installés, ouvrez un shell de commande et exécutez les opérations suivantes:

```
ffmpeg -f dshow -i video = "UScreenCapture" -r 10 -c: v libx264 -b: v 300k -preset ultra-rapide -b 300k -s 1280x800 -x264opts keyint = 50 -g 25 -pix_fmt yuv420p -f flv "rtmp://1.2.3.4:1936/live1/desktop "
```

les paramètres -b: v et -b correspondent au débit vidéo. -s est une taille de capture et -r est de 10 images par seconde, ajustez-le en fonction de votre configuration particulière. J'ai testé cela avec 333Ksps QPSK et il rend bien. Dans OBS, vous devez configurer une source pointant vers

rtmp: //1.2.3.4:1936/live1/desktop

pour pouvoir afficher l'image de votre bureau.

Evidemment, définissez l'adresse IP du serveur RTMP cible sur l'adresse appropriée. Si vous utilisez le serveur nginx RTMP de John GI7UGV, vous pouvez ajouter une autre directive pour prendre en charge plusieurs flux. Votre fichier `/etc/nginx/nginx.conf` pourrait ressembler à ceci:

```
worker_processes auto;

rtmp_auto_push on;

events {}

rtmp {
    server {
        listen 1935;

        application live {
            live on;

            record off;
        }

        listen 1936;

        application live1 {
            live on;

            record off;
        }
    }
}
```

Systeme d'intégration

Après avoir brièvement réfléchi à la façon d'intégrer PLUTO dans mon système à bande étroite QO100 existant, j'ai décidé qu'il serait préférable de placer tous les équipements nécessaires dans l'ODU, car il existe déjà un réseau, une référence de 10 MHz, un réseau local, une fréquence de 70 cm, etc. connexions à la boîte extérieure.



J'ai trouvé un vieil adaptateur USB <> Ethernet et un adaptateur OTG, branché sur le port USB IO du PLUTO, et une batterie de chargeur de téléphone dans le port USB du PSU. a démarré.

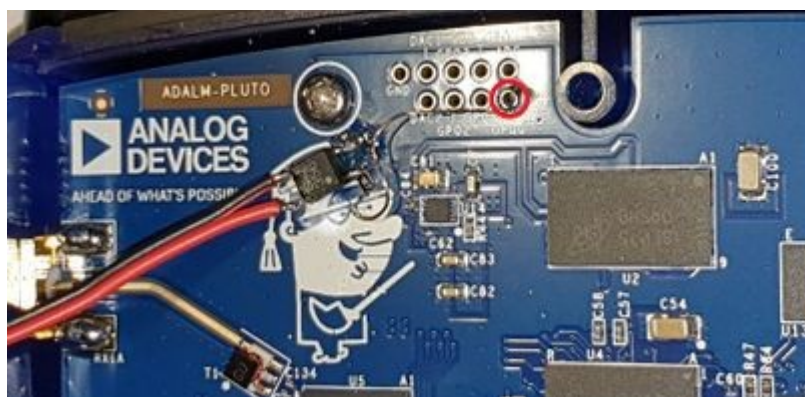
Le fonctionnement en réseau est vraiment la solution, car il élimine de nombreux problèmes USB et permet à plusieurs sources d'utiliser le SDR sans devoir déranger en permanence avec des câbles USB et des connecteurs micro-USB fragiles.

Matthias DD1US a écrit ses expériences sur PLUTO LAN sur qui méritent d'être digérées.

<https://tinyurl.com/y4qtcmau>

Lors de l'itération finale du système, ici, à la station terrestre MOEYT, j'utiliserai un bon bloc d'alimentation 5 V débranché, le PLUTO ira dans un boîtier métallique à des fins de stabilité mécanique et je monterai des prises SMA avec des connecteurs dos à dos qui ont le même espacement que le SDR, cela devrait garantir que rien ne soit cassé.

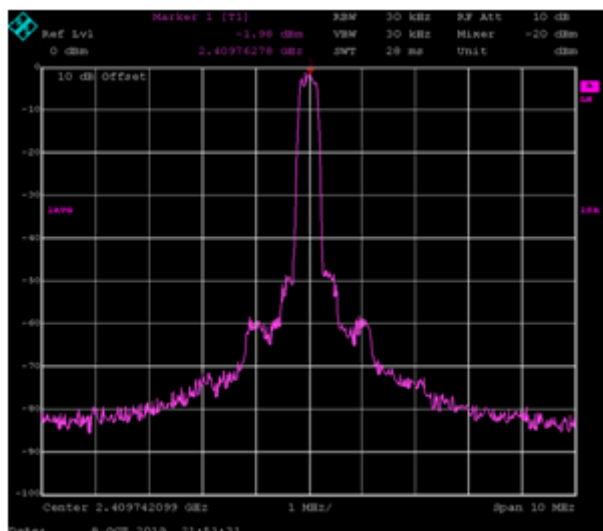
Le micrologiciel PLUTO prend désormais en charge une sortie PTT de sorte que, lorsqu'un démarrage de flux RTMP valide commence, la broche PTT change d'état pour permettre la modulation des amplificateurs, etc. PTT vient de GPO0 entouré en rouge ci-dessous;



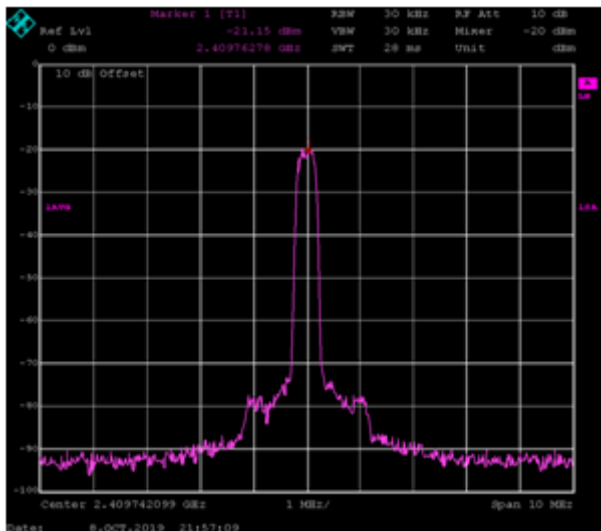
La sortie de niveau logique est 0 / 1,2V, ce qui suffit pour piloter un optocoupleur. Elle assure une isolation par décantation entre le PLUTO et ce que vous contrôlez. J'ai utilisé un optocoupleur '356' récupéré d'un vieux PCB. Remarque: Au cours du processus de démarrage PLUTO, la broche GPO0 est basculée du plus bas au plus élevé, y reste pendant 5 secondes, puis bascule à nouveau. Aucune sortie RF n'a été vue pendant cette bascule, etc.

Sujets RF

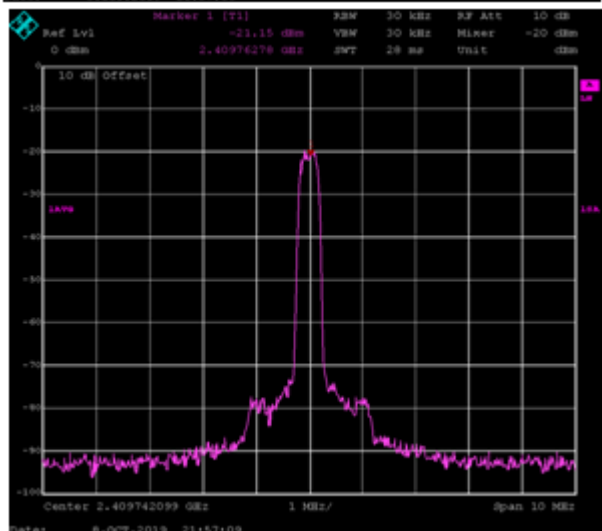
Le niveau de sortie RF du PLUTO est assez bas, environ -15 dBm à 2,4 GHz. Il est donc nécessaire de procéder à une amplification avec quoi que ce soit d'utile. le spectre de sortie, en particulier la repousse spectrale / les épaules. Toutes les mesures ci-dessous sont prises avec le système PLUTO générant DVBS2 à 2409,750 MHz, 8PSK, 333Ksps, 2/3 FEC.



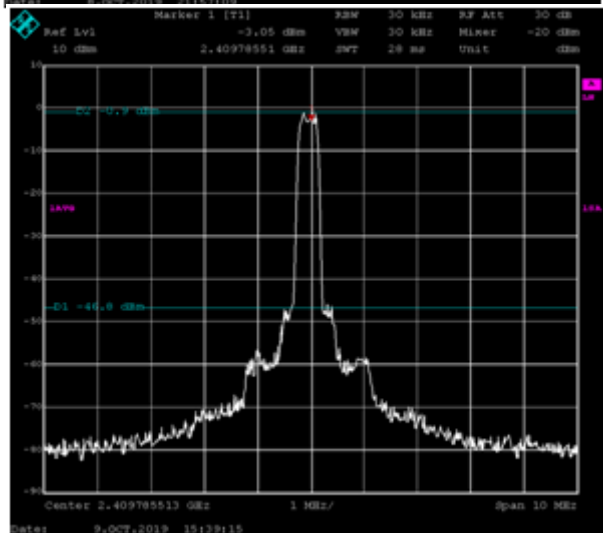
Avantek SA82-2340



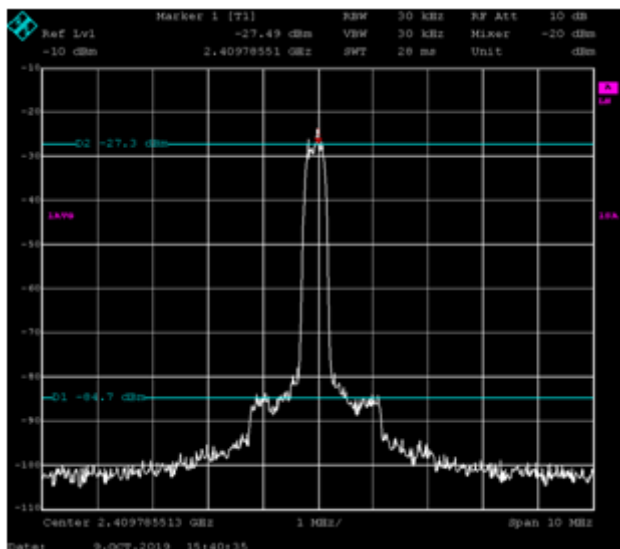
SPF5043 + 10dB Pad on o/p



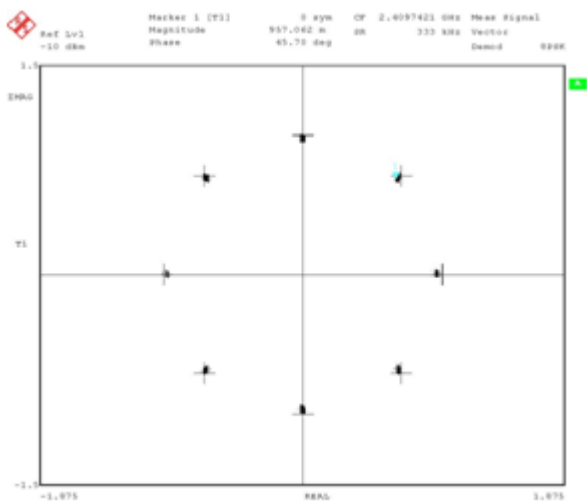
LNA4ALL + 10dB Pad on o/p



3 X amp 30~dB gain (+ 10dB Pad on o/p)



PLUTO direct output 8PSK baseline



PLUTO 8PSK baseline

D'après les tests effectués par les LNA pour obtenir un peu plus de sortie PLUTO, il apparaît que les dispositifs les plus modernes n'ajoutent pas d'IMD significatif à la xPSK, à condition qu'ils ne soient pas trop sollicités. Ma gamme actuelle de TX expérimentales est un préampli FET double prenant la sortie PLUTO à +5 dBm, un amplificateur de son secondaire évalué à 20 watts fonctionnant à 30 dB, puis un amplificateur Spectrian, modifié conformément à l'article de CQ-TV pour offrir une résolution de 45 dB. 30 watts. En regardant la constellation PSK, on constate qu'elle est relativement propre et que les épaules de chaque côté de la 8PSK sont à 35 dB, ce qui devrait être à peu près acceptable. Je vais essayer de trouver l'un des amplificateurs Axis-NT car cela résoudrait tous les problèmes d'alimentation.

Finalité

Le PLUTO SDR avec le micrologiciel F50EO offre certainement un moyen simple de générer une DATV de VHF jusqu'aux bandes hyperfréquences inférieures. Pour le QO100 avec une chaîne d'amplis appropriée, c'est l'idéal et le nombre d'utilisateurs apparaissant sur le transpondeur à large bande devrait augmenter. Il n'y a pas de porteuses d'étalonnage pré-transmission gênantes à projeter sur le transpondeur, ce qui est agréable. Un grand crédit doit être attribué à Evariste F50EO pour son travail remarquable sur ce firmware, et je vous recommanderais de faire un don pour ses efforts via <https://www.paypal.me/f50eo> - de quoi bien boire une bouteille de vin ou deux par exemple! Je suis sûr que plusieurs centaines d'heures de travail ont été consacrées à ce projet, un support ne risque donc pas de vous échapper et pourrait même encourager de nouvelles améliorations. à bientôt sur le transpondeur!

Merci à John GI7UGV pour la santé mentale de vérifier cette écriture ;-)

Traduction Google et mise en page par F6HDW Glomski Jean Claude

