

Démodulation et décodage de trames AX25 par DSPIC

pour la localisation d'un ballon sonde météo

dans le cadre d'une action "Planète Sciences"

Thierry Capitaine, Valéry Bourny, Mohamed Hamzaoui, Aurelien Lorthois, Jacky Senlis

¹ Université de Picardie Jules Verne, Laboratoire MIS, Faculté des Sciences, 80000, thierry.capitaine@u-picardie.fr
INSSET, 48 rue Raspail BP422 02100 Saint Quentin, jacky.senlis@u-picardie.fr

RESUME

Depuis huit ans, l'Institut Supérieur des Sciences et Techniques [1] propose une formation Master dans le domaine des systèmes embarqués. Lors des phases de travaux pratiques et des périodes de projets suivies par nos étudiants, nous privilégions l'implantation des connaissances théoriques dispensées dans le domaine des microcontrôleurs, DSP et FPGA, par une exploitation optimisée des ressources disponibles sur les cibles. Cette démarche, dans le cadre d'un projet de balises APRS [5] à embarquer dans un ballon, a été menée par un groupe d'étudiants de M1 SET. Dans cet article, nous décrivons un élément de ce projet : l'implantation de la démodulation FSK et du décodage des trames AX25 [4] dans un DSP.

Mots clés : Systèmes embarqués, dsp, traitement numérique du signal, AFSK, trames packet AX25.

1 CONTEXTE

Cette réalisation s'inscrit dans le cadre d'un projet piloté par "Planète Sciences", destiné aux élèves de troisième du collège de Gauchy dans l'Aisne.

1.1 Description du projet "ballon sonde"

Il s'agit de lancer et de suivre en temps réel le déplacement d'un ballon sonde météo équipé d'une carte de télémesures.

Cette carte dispose de huit entrées analogiques et d'une sortie VHF pour renvoyer les mesures à une station réceptrice au sol.

Le ballon est susceptible de monter à 30km d'altitude et de parcourir 300km.

1.2 Les intervenants et leur rôle

Le responsable de "Planète Sciences" a mis en relation les divers intervenants et a effectué les démarches administratives pour obtenir les autorisations nécessaires et régler les problèmes d'assurance. Le matériel (ballon sonde, hélium, carte de télémesures) a été fourni par le C.N.E.S.

Les élèves du collège, dans le cadre de leurs cours de technologie avaient pour tâche d'installer et de vérifier le bon fonctionnement des capteurs sur la carte de télémesures.

La problématique principale de ce type d'opération est la difficulté de poursuite (pointage antenne du récepteur de télémesures) et surtout la détermination du lieu de chute du ballon. Pour répondre à ces

objectifs, un groupe de cinq étudiants du master SET de l'INSSET a développé un module de géolocalisation et a formé les élèves à son intégration dans la nacelle du ballon.

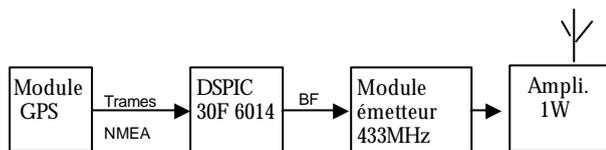
1.3 Objectif du module embarqué

Ce module aura en charge la transmission au format APRS des données de localisation reçues par un GPS embarqué ainsi que le décodage des ordres envoyés depuis le sol afin de déclencher le largage de la nacelle qui ouvrira le parachute.



2 LE SYSTEME DE GEOLOCALISATION

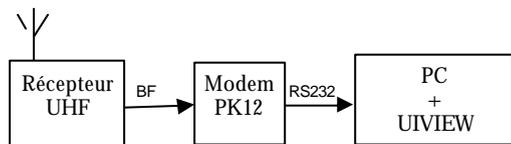
2.1 Description de la partie émettrice



Le système de localisation comporte un module émetteur UHF qui envoie les coordonnées GPS du ballon, longitude, latitude à intervalle de temps régulier.

Le programme du DSPIC doit extraire ces informations des trames NMEA [7] provenant du module GPS connecté sur son port série et les mettre en forme en respectant le protocole AX25 [4] pour moduler le signal d'entrée BF de l'émetteur. Le choix du protocole AX25 a été conditionné par la disposition de logiciels gratuits (décodage de trames et affichage graphique de position sur une carte) développés par la communauté radio amateur.

2.2 Description de la partie réceptrice



La sortie BF du récepteur radio est connectée à un modem PK12 qui démodule et décode les trames AX25. Le contenu de ces trames est envoyé par un port série au PC équipé du logiciel UIVIEW pour afficher automatiquement la position du ballon sur une carte.

3 DESCRIPTION DES TRAMES "AX25" [4] [5]

3.1 Modulation FSK Bell 202

En VHF, sur la fréquence 144.800MHz, les trames modulées en FSK sont audibles sur la sortie BF du poste récepteur.

Les fréquences BF utilisées sont $f_0 = 1200\text{Hz}$ et $f_1 = 2200\text{Hz}$.

Le débit binaire est de 1200 bit/s. La durée d'un bit correspond donc à une période du signal f_0 et à un peu moins de 2 périodes du signal f_1 .

Le codage est conçu pour être indépendant de la polarité du signal: un changement de fréquence correspond à un '0' logique et il n'y a pas de changement de fréquence pour les '1'.

"01111110" est un octet qui indique le début ou la fin d'une trame ("fanion"), il contient six '1' consécutifs.

Une trame ne doit donc pas comporter six '1' consécutifs, pour cela, on ajoute un '0' après une suite de cinq '1', ce '0' doit être supprimé à la réception ("bit stuffing").

Description d'une trame:

Fanion	Adresse	Contrôle	Information	FCS	Fanion
01111110	14 à 70 Octets	1 Octet	256 octets maxi.	2 octets	01111110

4 DEMODULATION DES TRAMES AX25

4.1 Objectifs

Démoduler et décoder les trames AX25 fournies par la sortie BF du poste récepteur et envoyer les informations sur un port série.

4.2 Contraintes

Au niveau de l'entrée BF, le dispositif ne doit pas être sensible à l'amplitude du signal (100mV à 2V) ni à l'offset. Il doit détecter les signaux en présence de bruit et utiliser le moins de composants possibles.

4.3 Le microcontrôleur

Nous avons choisi un DSPIC 30f6014 [6] produit par la société Microchip parce qu'il contient des fonctionnalités DSP (MAC en virgule fixe), des ports de communication et un convertisseur analogique numérique. Les outils de développement sont gratuits.

4.4 Les phases de développement

Dans un premier temps, nous avons recueilli des informations sur la couche physique du protocole AX25, puis nous avons mis au point et testé les algorithmes de démodulation et de décodage avec des enregistrements (fichiers wav) sous matlab. Cette étape étant validée, nous avons ensuite travaillé à l'implantation des algorithmes dans le DSPIC pour obtenir un fonctionnement en continu, la puissance de calcul du DSPIC permet de démoduler et reconstruire les trames au fur et à mesure de l'arrivée des échantillons.

Les paragraphes suivants décrivent pour chacune de ces étapes, les parties théoriques, algorithmiques et l'implantation logicielle.

5 ALGORITHME DU DEMODULATEUR FSK[3]

5.1 Principe

Le signal est échantillonné à la fréquence: f_e . On utilise la forme discrétisée de la transformée de Fourier sur une fenêtre glissante de largeur T_b , durée d'un bit, pour calculer l'amplitude des composantes fréquentielles f_0 et f_1 .

Soit C_0 et C_1 ces amplitudes, pour détecter un changement de fréquence (un 0 logique) il suffit de tester le signe de $DCC = C_0 - C_1$ après chaque calcul.

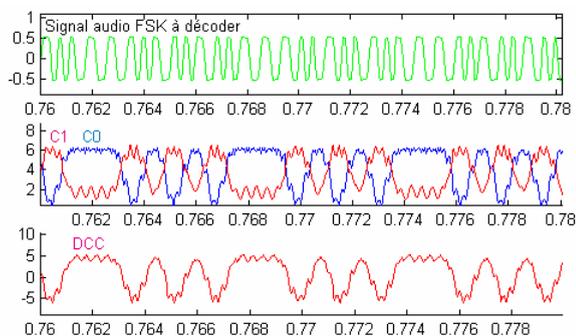
Soit N le nombre d'échantillons par bit : $N = \frac{T_b}{T_e} = \frac{f_e}{f_b}$

$$C_0(n) = \left| \sum_{i=0}^{N-1} x(n-i) \cdot \exp(j \cdot (n-i) \cdot \omega_0 \cdot T_e) \right|$$

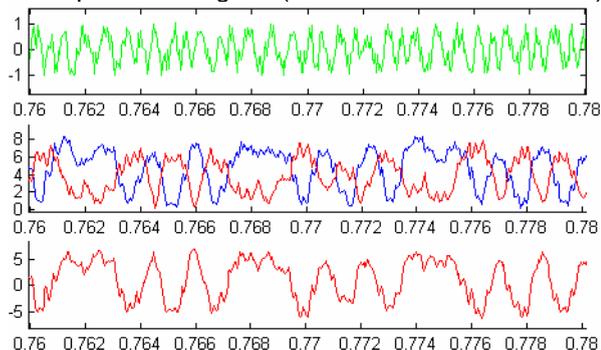
$$C_1(n) = \left| \sum_{i=0}^{N-1} x(n-i) \cdot \exp(j \cdot (n-i) \cdot \omega_1 \cdot T_e) \right|$$

5.2 Tests avec matlab

Les figures ci-dessous montrent les résultats obtenus avec matlab pour un fichier enregistré à 22kHz.



Le signal précédent peut encore être décodé malgré l'ajout d'un bruit aléatoire d'amplitude égale à l'amplitude du signal (fonction random de matlab)



5.3 Algorithme de décodage des trames

On repère les changements de fréquence ('0' logique) et on compte les bit à '1' entre ces changements, six bits à "1" consécutifs correspondent au début de la trame, les octets sont reconstruits au fur et à mesure de l'arrivée des bits selon l'algorithme décrit dans le fichier :

<http://igsenlis.free.fr/ax25/organigramme.doc>

5.4 Remarques pour la construction de la trame:

Les octets sont envoyés en commençant par le bit de poids faible (lsb) sauf pour le FCS

Les premiers octets reçus après le fanion correspondent au champ d'adresse, les bits de poids faibles de ces octets sont à '0' et les codes ASCII sont donc sur 7 bits et sont décalés de un bit à gauche...

Pour le reste de la trame les codes ASCII sont codés normalement avec huit bits.

Liens vers les fichiers matlab:

http://igsenlis.free.fr/ax25/ax25_matlab.zip

Contenu obtenu après décodage du fichier ax25_05_11.wav :

```
APRS pF6KNP 4F5SNV sF1ZRG
sTRACE6j_0!4658.33N/00553.19# Relais
APRS REF 39 Mt Poupet 850m 7W UiDigi
1.9b1_β~
```

6 IMPLANTATION DANS LE DSP

6.1 Choix de la fréquence d'échantillonnage

Les essais sous matlab avec différents enregistrements montrent que les résultats sont meilleurs avec une fréquence d'échantillonnage élevée. Il faut cependant tenir compte du temps de calcul qui augmente avec le nombre d'échantillons et de la vitesse maximale du convertisseur.

Nous avons choisi un multiple commun à f_0 et f_1 pour avoir un nombre entier d'échantillons par période.

Avec $f_e = 26400\text{Hz}$ le dsp dispose de $38\mu\text{s}$ pour calculer C_0 et C_1 et faire le décodage.

Les calculs de C_0 et C_1 nécessitent $4 \cdot N_b$ multiplications avec $N_b = \frac{f_e}{f_b} = 22$ échantillons par bit.

Pour faire 88 multiplications, en utilisant les buffers circulaires, le dsp met environ $88 \cdot T_{cy}$ soit moins de $3\mu\text{s}$; ce qui laisse suffisamment de temps pour reconstruire dynamiquement la trame.

Remarque: La fréquence d'échantillonnage maximale est limitée par le temps de calcul et par la vi-

tesse du convertisseur. Les relations suivantes doivent être vérifiées:

$$T_e \geq T_{calcul} \quad \text{et} \quad f_e < f_{max_convertisseur}$$

En tenant compte uniquement des multiplications,

$$\text{on trouve } f_e = \frac{\sqrt{f_{cy} * f_b}}{2} = 94 \text{ kHz.}$$

Cette valeur ne tient pas compte des autres opérations: appel de l'interruption, sauvegarde et restitution du contexte, reconstructions des octets ni de l'envoi sur le port série. C'est pourquoi la fréquence $f_e = 26.4 \text{ kHz}$ a été retenue après quelques essais.

6.2 Utilisation du convertisseur

La conversion est déclenchée automatiquement par un TIMER lors de son passage par la valeur de son registre de période, ceci garantit un échantillonnage précis à $1 T_{cy}$ près ($0.03 \mu\text{s}$). La fin de conversion produit une demande d'interruption qui lance automatiquement la routine de calcul de C0, C1 ainsi que la reconstruction et l'envoi des octets obtenus sur le port série.

6.3 Utilisation des fonctionnalités DSP pour optimiser le calcul de C0 et C1

A l'initialisation, on précalcule les coefficients $\cos(n \cdot w_0 \cdot T_e)$, $\sin(n \cdot w_0 \cdot T_e)$, $\cos(n \cdot w_1 \cdot T_e)$ et $\sin(n \cdot w_1 \cdot T_e)$ pour $n=0, 1, \dots, N-1$. Ces coefficients sont codés en virgule fixe 1.15 en les multipliant par 2^{15} et sont stockés dans un buffer circulaire de la RAM_X du DSPIC.

Au fur et à mesure de leur arrivée, les échantillons, sont stockés dans un buffer circulaire, de taille N dans la RAM_Y. L'échantillon qui vient d'arriver remplace simplement le plus ancien. Il n'y a pas d'opération de décalage d'échantillons en mémoire. Le travail le plus délicat consiste à bien gérer les registres qui pointent sur les coefficients et les échantillons lors des calculs des parties réelles et imaginaires de C0 et C1

Chacun des termes à calculer étant une somme de produit, l'instruction "MAC" (multiplication accumulée) associée à un registre "REPEAT" permet d'optimiser le temps de calcul: l'instruction qui suit "REPEAT n" est décodée une seule fois et exécutée n+1 fois.

6.4 Algorithme de reconstruction de trames

Il s'agit du même algorithme que celui utilisé avec matlab. Cependant la reconstruction de trame se fait en temps contraint, au fur et à mesure de l'arrivée des échantillons, et les octets reconstruits ne sont pas stockés en mémoire mais envoyés directement sur la sortie série du DSPIC (connecter un terminal ASCII sur la sortie série N°2 pour lire les messages décodés).

6.5 Tests

Les résultats sont satisfaisants avec des signaux d'amplitude 0.2V à 2V, même bruités, sans modifier le programme.

Toutes les trames sont décodées et envoyées sur le port série, même les trames non valides car le FCS n'est pas testé.

Ce mot de contrôle pourrait être calculé au fur et à mesure de la réception de la trame, de façon à détecter les erreurs de transmission.

6.6 Les fichiers sources

http://jgsenlis.free.fr/ax25/ax25_30f6014.zip

decode_ax25.c: Routines de démodulation et de décodage

gen_AX25.c: Pour fabriquer des trames AX25 modulées en FSK sur une sortie PWM (à compléter par le calcul et l'envoi du FCS si on souhaite contrôler la validité des trames)

port_serier.c: Pour gérer le port série

menu.c: Contient le "main" et l'interface utilisateur

7 CONCLUSION

Les étudiants en plus de réaliser un projet technique, ont participé activement à l'encadrement des élèves sur la partie "intégrations des capteurs" et ont participé au lâcher du ballon.

La partie modulation / démodulation numérique des signaux FSK leur a permis de mettre en application de façon très concrète leurs connaissances en traitement du signal.

L'utilisation de l'instruction MAC associée aux buffers circulaires est bien adaptée aux calculs à réaliser et aux performances à atteindre.

Bibliographie/liens internet

- 1.INSSET, formations Master et Licence3 « Systèmes Embarqués dans les Transports » www.insset.u-picardie.fr/enseignement/iupset/iupset.php
2. Planète Sciences <http://www.planete-sciences.org>
- 3.Electronique "Systèmes de communication et de filtrage" F.Manneville J.Esqueieu
- 4.Description du protocole AX25 <http://www.tapr.org/pdf/AX25.2.2.pdf>
- 5.Site sur l'APRS <http://packetradio.free.fr/aprs.htm>
- 6.Datasheet du 30f6014 : <http://www.microchip.com/>
7. Trames N.M.E.A <http://fr.wikipedia.org/wiki/Nmea/>