

Examen transmission de l'information

J.-M Friedt, 14 mars 2016

Pour chaque question **en caractères gras**, vous ferez valider votre réponse par une démonstration rapide à un enseignant. Si vous le désirez, un document numérique pourra être transmis par courrier électronique à jmfriedt@femto-st.fr. Vous rédigerez pour toutes les questions un texte donnant les réponses et les calculs sur une feuille.

Tous les documents papier et électroniques autorisés, accès aux téléphones portables proscrit.

1 Identification de signaux transmis sur porteuse radiofréquence

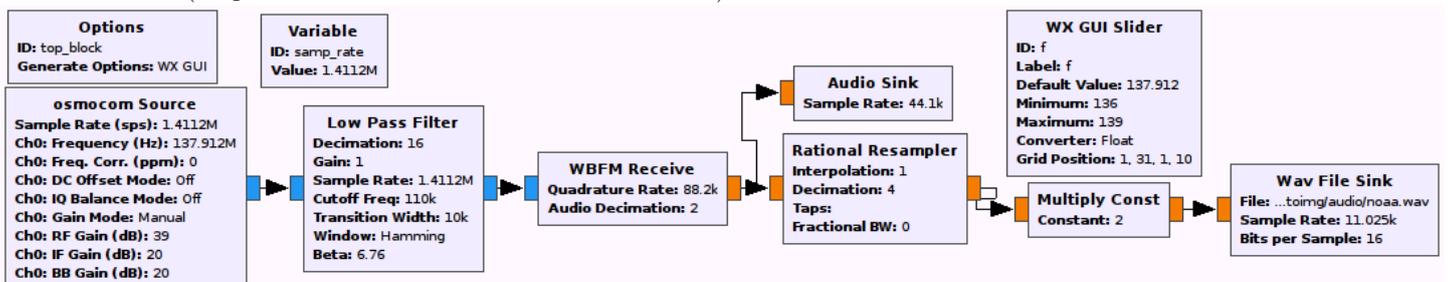
Deux signaux sont émis sur des porteuses dans la bande industrielle, scientifique et médicale comprise entre 433,05 et 434,79 MHz. Deux porteuses, à des fréquences différentes, sont modulées par le même signal numérique mais selon des modes de modulation différents.

1. Après analyse des informations fournies par GNURadio lors de l'acquisition de signaux issus d'un récepteur de télévision numérique terrestre, identifier les deux fréquences des deux porteuses qui nous intéressent.
2. En comparant les structures spectrales autour de ces porteuses, identifier lequel des deux modes de modulation utilise un encombrement spectral le plus important.
3. Rappeler les trois modes de modulation possibles, et proposer deux modes possibles de modulation compte tenu de leur encombrement spectral relatif.
4. **Valider le mode de modulation de la porteuse de fréquence la plus basse en proposant un schéma de démodulation dans gnuradio-companion, et afficher l'évolution temporelle du signal.**
5. **Valider le mode de modulation de la porteuse de fréquence la plus haute en proposant un schéma de démodulation dans gnuradio-companion, et afficher l'évolution temporelle du signal.**
6. Le même signal numérique attaque les canaux de modulation externes de deux synthétiseurs radiofréquences émettant des porteuses de fréquence différentes : est-on capables de démoduler *simultanément* les deux canaux de communication ?
7. Si oui, vérifier que les signaux en sortie des deux démodulateurs (pour les deux modes de modulation différents) portent la même information.
8. Quel est le débit des informations numériques transmises ?
9. Bonus : décoder le message transmis, sachant qu'un encodage compatible RS232 est utilisé.

2 Décodage d'une image de satellite météorologique

Un satellite météorologique en orbite polaire basse transporte une information sur une double-porteuse : la porteuse radiofréquence est modulée en fréquence par un signal audiofréquence qui est lui-même modulé en amplitude. L'amplitude de ce signal audiofréquence est représentative de l'intensité lumineuse d'un photodétecteur embarqué sur le satellite. La rotation du satellite, au rythme de 2 tours/seconde, sur son axe parallèle à sa trajectoire induit le balayage selon l'abscisse, et l'avancée du satellite le long de sa trajectoire induit le balayage selon l'ordonnée de l'image.

Le récepteur radiofréquence se charge d'éliminer la modulation FM sur la porteuse radiofréquence : le résultat de cette réception est un signal audiofréquence à télécharger à http://jmfriedt.sequanux.org/160306_14h29_noaa19.wav échantillonné à 11025 Hz que nous allons analyser. L'enregistrement est formé d'échantillons codés sur 16 bits signés organisés en little endian (on pourra le vérifier avec la commande `file`).



suite au verso...

1. Quelle est la fréquence du signal audiofréquence qui porte l'information d'amplitude? Justifier/démontrer. Le fichier sera chargé pour ce traitement dans GNU/Octave par `d=fread(f,inf,'int16')`; après création du descripteur de fichier `f`.
2. Sur combien d'échantillons du signal issu du fichier `.wav` une période de ce signal audiofréquence est-il encodé?
3. Comment implémenter un démodulateur d'amplitude sur ce signal?
4. La fonction `reshape(data,M,N)` permet, sous GNU/Octave ou au moyen de `numpy` sous Python¹, de prendre un vecteur de $M \times N$ éléments (`data`) et de former une matrice de M lignes et N colonnes. Sachant que le satellite tourne sur lui-même au rythme de 2 Hz, transformer la séquence d'amplitudes du signal en une image.
5. Justifier de la forme de l'image (plusieurs causes possibles).

3 Questions de cours

1. Connaissant l'ordre de grandeur de la fréquence des signaux de la bande FM commerciale, quelle est la taille "idéale" d'antenne sur un toit de véhicule pour recevoir un tel signal?
2. Un signal à 10 kHz est échantillonné à 2 Méchantillons/seconde. Quelle est l'abscisse de la raie correspondante dans une transformée de Fourier discrète sur 1000 points? Combien de temps a-t-il fallu pour acquérir ces 1000 points?
3. Quelles sont les quatre méthodes de partage du spectre entre plusieurs interlocuteurs?
4. Quelle architecture de circuit permet de démoduler un signal FM?
5. Quelle opération mathématique permet de s'affranchir de la modulation BPSK pour retrouver la porteuse (plusieurs réponses possibles)?
6. Quelles opérations mathématique permettent de retrouver le signal modulé en AM?
7. Un oscillateur local à 104,4 MHz présente une sensibilité thermique de 10 ppm/K. Quelle est sa variation de fréquence si la température varie de 10 °C?
8. Comparer cette variation de fréquence à la bande passante d'une modulation de la bande FM commerciale : est-ce qu'un démodulateur FM sera capable de rattraper une telle erreur? Justifier.
9. Sur combien d'octets est codée une adresse IP (deux réponses possibles)? une adresse matérielle (MAC)?
10. Quelle est la différence entre les protocoles de transport UDP et TCP?

1. syntaxe `reshape(data, (N,M))`

Identification de signaux transmis sur porteuse radiofréquence

Sans entrer dans les détails de chaque question qui visait à proposer une démarche étape par étape pour identifier deux porteuses radiofréquences à **434,0** et **434,6** MHz, les blocs de traitement de la Fig. 1 permettent de facilement visualiser le spectre et observer les deux porteuses sous forme de raies fines dans la bande [433,05; 434,79] MHz. On notera qu'il est peu judicieux de décimer la sortie du récepteur DVB-T lors de la visualisation du spectre : la fréquence d'échantillonnage de 2 MS/s permet de visualiser directement tous les messages compris dans la bande ISM en se centrant sur 434 MHz (qui s'avère être une des porteuses).

En zoomant sur chacune de ces raies et en activant la persistance pour mémoriser le spectre des paquets d'informations transmises, nous constatons sur la Fig. 1 à gauche une porteuse unique (décalée de -35 kHz en l'absence d'oscillateur commun entre émetteur et récepteur) décorée de part et d'autres de raies additionnelles laissant penser à une modulation d'amplitude (faible encombrement spectral : une porteuse et quelques raies autour). Au contraire, la porteuse à plus haute fréquence (Fig. 1, droite) présente une seconde composante décalée de 100 kHz sous la porteuse émise en continu. L'encombrement spectral de ce second mode est par ailleurs bien plus important : cela laisse penser à une modulation de phase ou de fréquence. La seconde option sera la bonne. Comme l'état au repos d'un signal RS232 est le niveau haut, nous émettons majoritairement sur la fréquence modulant l'état 1 (porteuse + 100 kHz) et verrons passer quelques bits à 0 lors des transmissions (ici encore, le récepteur DVB-T est décalé d'environ -35 kHz par rapport au synthétiseur de fréquence).

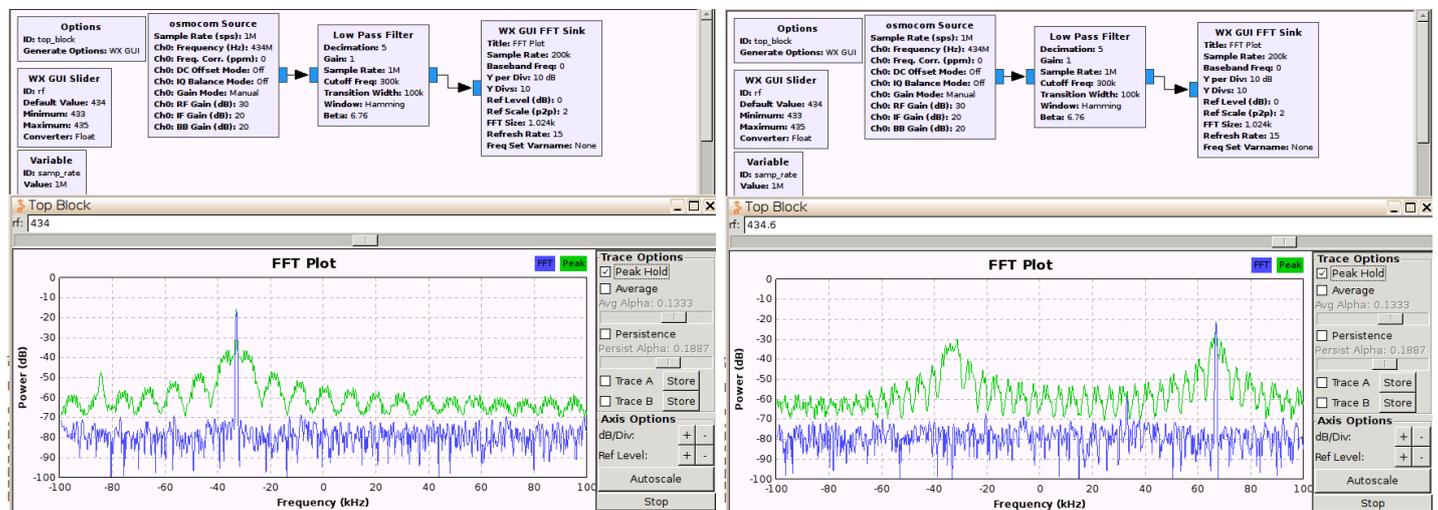


FIGURE 1 – Analyse préliminaire des deux porteuses et, par mémorisation du maximum de chaque spectre, de leur encombrement spectral.

Fort de ces observations, il semble raisonnable de penser qu'un signal RS232 issu d'un microcontrôleur module les deux entrées externes de deux synthétiseurs radiofréquences de porteuse 434,0 et 434,6 MHz. En effet, le montage de la Fig. 2 (droite) illustre un tel dispositif. Sans passer par l'étape des démodulations individuelles, le schéma de traitement de la Fig. 2 (gauche, haut) permet de démoduler *simultanément* les deux signaux par transposition de fréquence d'une des deux porteuses (Xlating FIR Filter pour ramener la porteuse en bande de base). Nous constatons (Fig. 2, gauche en bas) que les deux modes de modulation portent bien la même modulation. Pour bien visualiser, le déclenchement (*Trigger*) est placé en mode Normal et légèrement décalé par rapport à l'origine des ordonnées. Le filtre passe-haut (ou DC blocker) permet de facilement visualiser les deux signaux simultanément, qui sont sinon affectés d'un biais qui rend leur visualisation moins confortable. Noter que la configuration par défaut du bloc de démodulation AM – avec une bande passante de 5 kHz et une fréquence de coupure de 5,5 kHz – ne fournit pas une bande passante suffisante pour bien définir les bits, avec des changements d'état s'apparentant plus à des segment de sinusöide que des crénaux. Nous augmentons donc la bande passante de démodulation à 15 ou 25 kHz.

Constatant (Fig. 2, gauche en bas) que 10 bits sont contenus dans un intervalle de 1 ms (par exemple en observant le signal entre 7 et 8 ms), nous pouvons penser que le signal est transmis autour de 10 kbauds, et sachant que le protocole est compatible RS232, nous pouvons proposer un débit de 9600 bauds.

En RS232, nous avons un premier passage de haut à bas pour le bit de start, suivi de 7 ou 8 bits de données, éventuellement 1 bit de parité et 1 ou 2 bit(s) d'arrêt avec passage au niveau haut. Les données sont transmises avec le bit de poids faible en premier. Ainsi pour le premier caractère, 00010010 avec le bit de poids faible en premier devient 01001000=0x48='H' et le bit qui suit est bien à 1, indiquant le bit de stop. Le bit suivant passe à 0 interdisant deux bits de stop, donc nous sommes

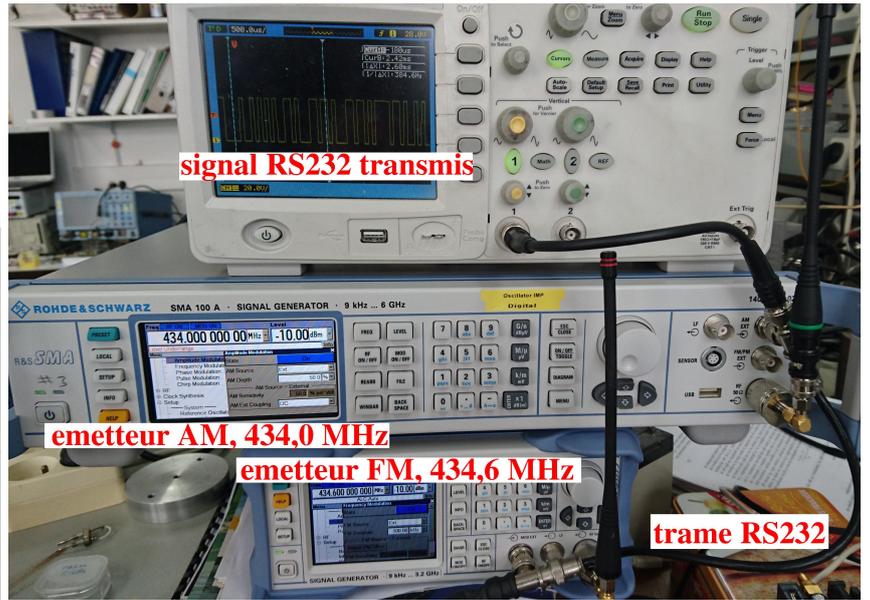
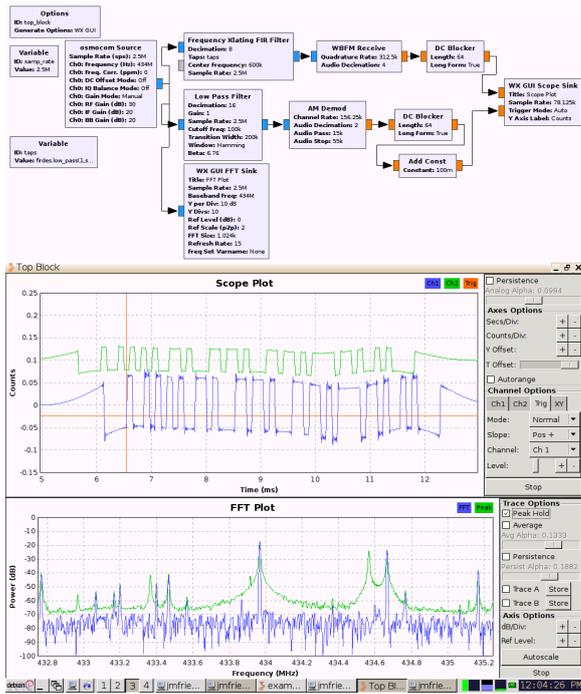


FIGURE 2 – Droite : montage expérimental. Gauche : schéma des traitements (haut) et signaux résultants (bas). Dans ce schéma de traitement, les filtres passe-haut (*DC-blocking*) sont inclus afin d'éliminer le biais sur les courbes et les ramener sur une ordonnée commune. Leur rôle est uniquement de simplifier la visualisation sur un unique bloc oscilloscope à deux voies et éventuellement aider à la détection automatique de changement d'état par passage par 0. Ces blocs *DC-blocking* ne seront pas utiles si deux graphiques distincts représentent les résultats de démodulation AM et FM (Fig. 3).

repartis pour une transaction avec ce bit de start, et la séquence 10100110 et un niveau 1 indiquant le stop. Le code ASCII de valeur 0x65 est 'e' et nous avons alors de fort soupçons que le message transmis est "Hello" suivi d'un retour à la ligne, tel que nous pourrions le vérifier en finissant cette analyse (Fig. 3).

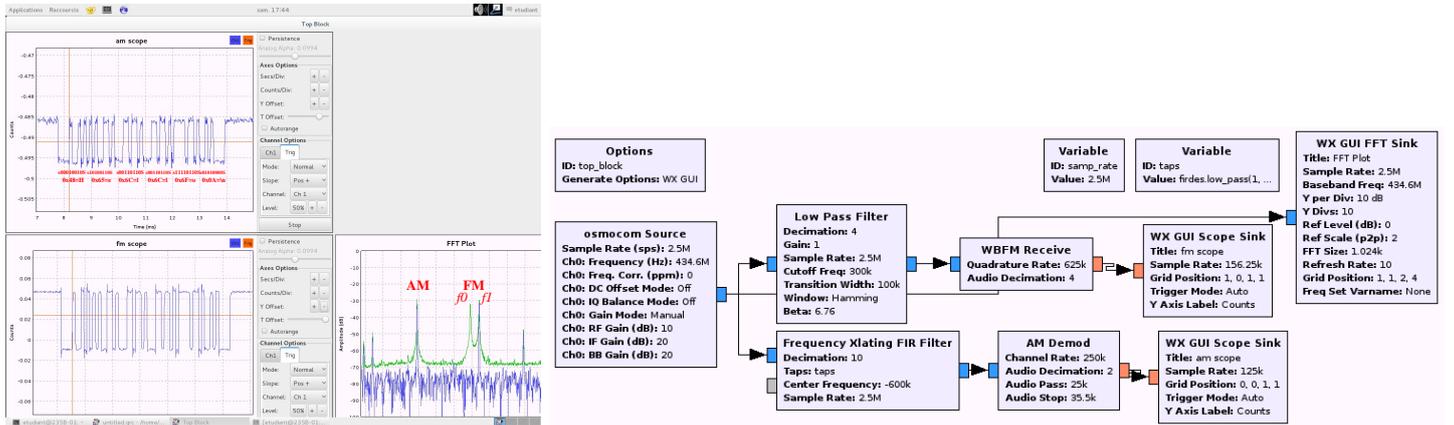


FIGURE 3 – Acquisition en salle 235B (gauche) au moyen du schéma de traitement proposé sur la figure de droite. La position des figures est explicitement indiquée pour tenter de faire tenir tous les graphiques à l'écran simultanément.

Décodage d'une image de satellite météorologique

1. Une fois le fichier binaire chargé en mémoire (`fread` sous GNU/Octave, `fromfile` avec Numpy pour Python), une transformée de Fourier du signal indique une raie parfaitement visible autour de **2400 Hz**. Les deux pièges sont d'une part de ne pas se contenter de regarder le tout début du fichier (qui comporte l'entête du format wav) mais de chercher un peu plus loin, et ne pas se tromper sur l'axe des abscisses. La FFT de N points distribue le résultat sur un axe des fréquences compris entre $-11025/2$ et $11025/2$ (après `fftshift` qui place la fréquence nulle au milieu du graphique, ou de 0 à 11025 en l'absence de `fftshift`).
2. Une période du signal audiofréquence modulé en amplitude dure $11024/2400=5$ points.
3. Une détection d'amplitude s'obtient par redressement et filtre passe bas. Dans sa version numérique, le redressement s'obtient après avoir retranché la valeur moyenne (biais) du signal et pris la valeur absolue. Le filtre passe bas est ici réduit à sa plus simple expression de moyenne glissante sur 5 points. Afin que le filtre conserve l'énergie, nous divisons le vecteur de 1 par le nombre d'éléments.
4. Avec une rotation au rythme de 2 Hz, nous avons $11025/2=5512,5$ points sur chaque ligne. Si en plus de la sortie du filtre passe bas nous décimons le signal (par exemple d'un facteur 5), nous aurons environ 1102,5 points par ligne.

Ces concepts s'implémentent sous GNU/Octave par

```
% $ file 160221_noaa18_noaa.wav
% 160221_noaa18_noaa.wav: RIFF (little-endian) data, WAVE audio, Microsoft PCM, 16 bit, mono 11025 Hz
fs=11025; N=4096;
f=fopen('160306_14h29_noaa19.wav'); d=fread(f,inf,'int16'); fclose(f); % 160221_noaa18_noaa.wav

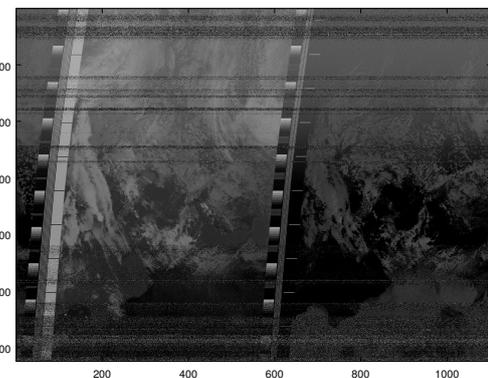
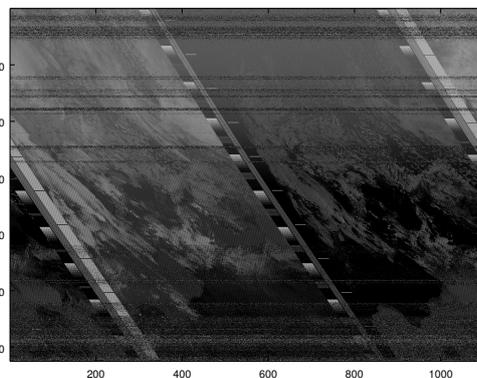
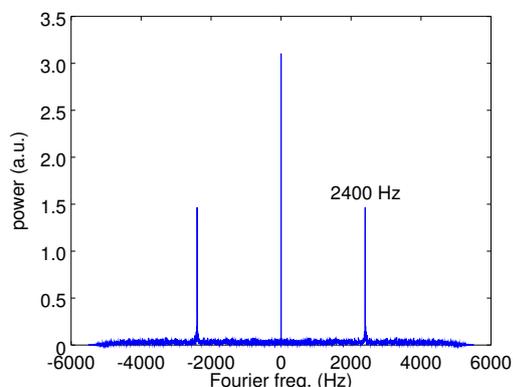
% afficher la FFT du signal pour trouver la fr\`equence de la porteuse audiofr\`equence
freq=linspace(-fs/2,fs/2,N);
df=abs(fft(d(1e5:1e5+20000),N)); % on trouve une porteuse \`a 2400 Hz
plot(freq,fftshift(df)); xlabel('Fourier freq. (Hz)');ylabel('power (a.u.)')

% retrancher la valeur moyenne et valeur absolue pour redresser le signal
moyenne=mean(d(1e5:1e5+20000));
dd=abs(d-moyenne);
x=ones(5,1)/5; % filtre passe bas pour lisser et garder l'amplitude
ddf=conv(dd,x); ddf=ddf(1:5:end); % ... sur 5 points car 11025/2400=5

% transformer le vecteur en image et afficher : 11025/5/2=1102..1103
image=reshape(ddf(1:1103*1250),1103,1250);
figure; imagesc(flipud(image'),[500 2000]); colormap gray % [0 1000] pour 160221

% pour vraiment bien faire, on alterne 1102 et 1103
l=270 % l'image ne commence pas en 1, on coupe le debut
for k=1:2:1250
    image(1:1102,k)=ddf(1:l+1102-1)'; l=l+1102;
    image(1:1103,k+1)=ddf(1:l+1103-1)'; l=l+1103;
end
figure; imagesc(flipud(image'),[500 2000]); colormap gray % [0 1000] pour 160221
```

Une fois le signal redressé acquis, le vecteur d'amplitudes (intensité lumineuse du photodétecteur) est réorganisé sous forme d'image : le satellite tourne sur lui même en 2 secondes ou $11025/2=5512,5$ points. Étant donné que nous avons décimé le vecteur d'amplitudes en ne gardant que un point sur cinq (*i.e.* une estimation d'amplitude par période), il ne reste que $5512,5/5=1102,5$ points/ligne.



L'image n'est pas droite car

1. la fréquence d'échantillonnage n'est pas paire, et nous n'avons pas un nombre constant d'échantillons par ligne alors que la satellite tourne au rythme de 2 Hz (image du milieu – cet artefact est corrigé sur l'image de droite),
2. l'horloge qui cadence la carte son n'est pas synchrone avec l'horloge embarquée sur le satellite. Pour ajuster la forme de l'image, nous devons nous recaler sur la frise de couleurs qui fournit justement ce signal de synchronisation.

La même chose s'obtient en **Python** par

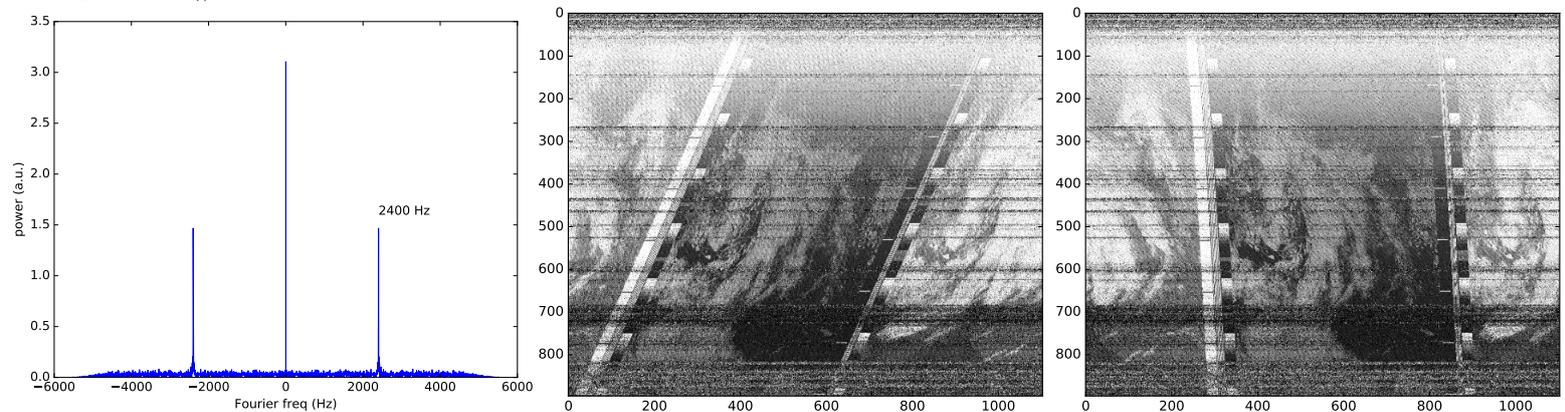
```
import numpy as np
import scipy as sy
import matplotlib.pyplot as plt

# 160221_noaa18_noaa.wav: RIFF (little-endian) data, WAVE audio, Microsoft PCM, 16 bit, mono 11025 Hz
fs = 11025; N = 4096;
d = np.fromfile('160227_noaa19_orleans.wav', dtype='<i2')

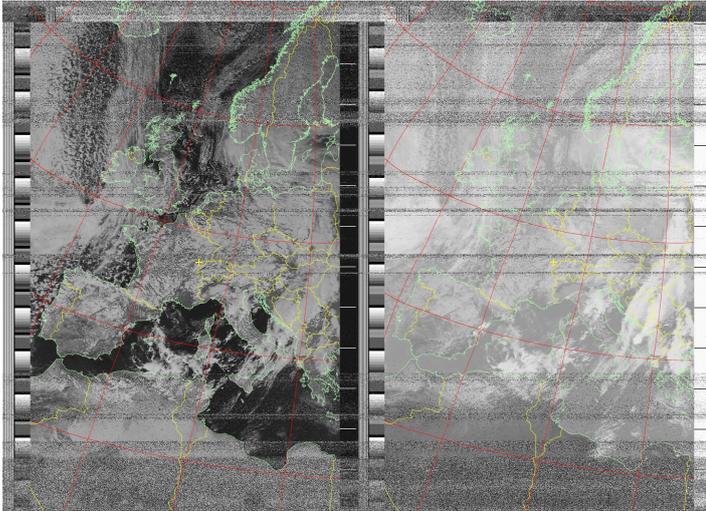
# afficher la FFT du signal pour trouver la fr'equence de la porteuse audio fr'equence
yf = sy.fft(d[1e5:1e5+200000], N); yf = np.fft.fftshift(yf)
freq = sy.linspace(-fs/2, fs/2, num=N) # on trouve une porteuse \a 2400 Hz
plt.plot(freq, np.abs(yf)/1e6); plt.ylabel('power (a.u.)'); plt.xlabel('Fourier freq (Hz)')
plt.text(2400, 1.6, '2400 Hz');

# retrancher la valeur moyenne et valeur absolue pour redresser le signal
moyenne = np.mean(d[1e5:1e5+200000]);
dd = np.abs(d - moyenne);
x = np.ones(5)/5; # filtre passe bas pour lisser et garder l'amplitude
ddf = sy.convolve(dd, x, 'same'); ddf = ddf[0:-1:5]; # ... sur 5 points car 11025/2400=5
# transformer le vecteur en image et afficher : 11025/5/2=1102.1103
image = np.reshape(ddf[0:1103*897], (897, 1103)); # print ddf.shape pour verifier combien on a de points
plt.figure(); plt.imshow(np.fliplr(np.flipud(image)), clim=(500, 1200), cmap='gray'); # plt.colorbar()

# pour vraiment bien faire, on alterne 1102 et 1103
l = 269 # l'image ne commence pas en 1, on coupe le debut
image2 = np.empty_like(image); image2[:] = image
for k in range(0, 896, 2):
    image2[k, 0:1102] = ddf[l:l+1102]; l = l+1102
    image2[k+1, 0:1103] = ddf[l:l+1103]; l = l+1103
plt.figure(); plt.imshow(np.fliplr(np.flipud(image2)), clim=(500, 1200), cmap='gray')
plt.show()
```



Pour comparaison, le fichier proposé pour traitement au cours de l'examen génère l'image de gauche ci-dessous après traitement par `wxtoimg` (disponible à <http://www.wxtoimg.com/>) qui s'asservit sur le code de synchronisation fourni en chaque début de ligne. Pour comparaison, l'image de droite est issue du site Meteosat (image infra-rouge) après téléchargement à la même date.



Acquisition DVB-T/taitement wxtoimg

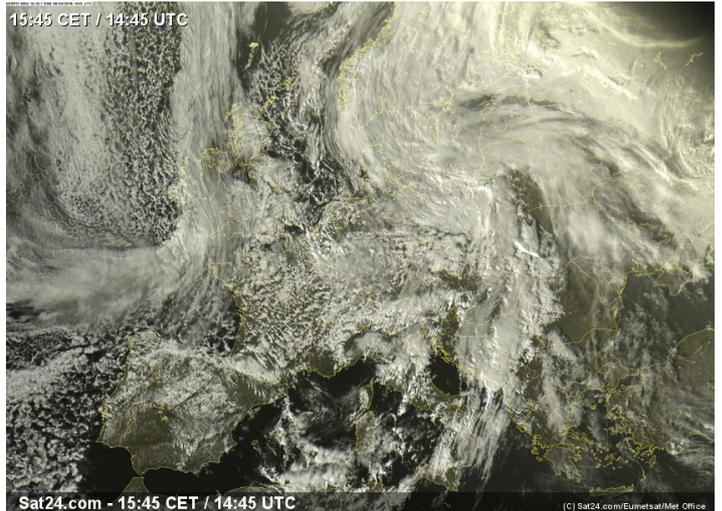


Image Meteosat

Questions de cours

1. $\lambda = 300/100 = 3$ m donc $3/4=0,75$ m.
2. 2 MHz de bande sur 1000 points donc le spectre s'étend de -1 MHz à +1 MHz en 1000 points. L'axe des abscisses va donc de 0 à 1 MHz sur 500 points et l'abscisse représentant le signal de 10 kHz sur trouve à $1/100$ de la bande soit l'abscisse d'indice 5. Il faut $500 \text{ ns} \times 1000 \text{ points} = 500 \mu\text{s}$ pour acquérir ces 1000 points.
3. {TFSC}DMA (temps, fréquence, espace, code)
4. boucle à verrouillage de phase (PLL)
5. $\arctan(Q/I)$ ou mise au carré du signal
6. redressement donc retrancher la valeur moyenne et valeur absolue suivie d'un passe-bas
7. $104,4 \times 10 = 1,044 \text{ kHz/K}$ donc 10,44 kHz pour une variation de température de 10 K.
8. La bande FM commerciale propose des stations séparées d'au moins 200 kHz donc la dérive en température est aisément rattrapable.
9. 4 octets pour IP v4, 16 octets pour IP v6. L'adresse MAC est sur 6 octets.
10. TCP est un protocole de transport connecté dans laquelle l'intégrité des paquets est garantie. UDP est un mode non-connecté qui ne garantit pas l'intégrité des transactions : c'est éventuellement à la couche applicative de se soucier de la perte de données.