



Réalisation d'un modem de fréquence

ARNOLD Cyprien
HABRE Joe

Département Sciences du Numérique - Première année
2020-2021

Table des matières

1	Introduction	3
2	Modem de fréquence - Démodulation par filtrage	3
2.1	Construction du signal modulé en fréquence	3
2.1.1	Génération du signal NRZ	4
2.1.2	Génération du signal modulé en fréquence	5
2.2	Canal de transmission à bruit additif, blanc et Gaussien	7
2.3	Démodulation par filtrage	7
2.3.1	Synthèse du filtre passe-bas	7
2.3.2	Synthèse du filtre passe-haut	8
2.3.3	Filtrage, tracés, commentaires sur les résultats obtenus	9
2.3.4	Détection d'énergie	10
2.4	Application de la recommandation V21	11
3	Modem de fréquence V21 - Démodulateur FSK	11
3.1	Démodulateur FSK - Contexte de synchronisation idéale	11
3.2	Démodulateur FSK avec gestion d'une erreur de synchronisation de phase porteuse	13
4	Conclusion	14
5	Références	14

Table des figures

1	Signal modulé en fréquence	3
2	Signal NRZ pour 50 bits.	4
3	Densité spectrale de puissance de NRZ entre $-f_e/2$ et $f_e/2$ (échelle log)	4
4	Densité spectrale de puissance de NRZ zoomée au centre.	5
5	Signal modulé en fréquence sur matlab.	5
6	Tracé du signal $x(t)$ en fonction du temps (s) pour 20bits.	5
7	Densité spectrale de puissance de $x(t)$ en fonction de la fréquence (Hz) (en échelle log).	6
8	Démodulation par filtrage.	7
9	Réponse impulsionnelle et en fréquence du filtre passe-bas.	8
10	Superposition de la densité spectrale de puissance de $x(t)$ et de la réponse fréquentielle du passe-bas.	8
11	Calcul de la réponse impulsionnelle du passe-haut sous Matlab.	8
12	Réponse impulsionnelle et en fréquence du filtre passe-haut.	9
13	Superposition de la densité spectrale de puissance de $x(t)$ et de la réponse fréquentielle du passe-haut.	9
14	Signal bruité avant et après filtrage pour 10 bits et une puissance de bruit de 15 dB	10
15	Densité spectrale de puissance normalisé de $x(t)$ après filtrage.	10
16	Solution Matlab pour la détection d'énergie.	11
17	Image reconstituée.	11
18	Démodulation FSK. Synchronisation supposée idéale.	12
19	Solution Matlab pour le démodulateur FSK en synchronisation idéale.	13
20	Démodulation FSK - Gestion d'une erreur de phase porteuse.	14

1 Introduction

L'objectif de ce projet est de concevoir un modem suivant la recommandation V21 du CCIT. Un modem est utilisé, d'une part, pour transformer une information numérique à transmettre en un signal susceptible de se propager sur le support physique à traverser entre l'émetteur et le récepteur (modulation), d'autre part, pour retrouver, à partir du signal abimé par le lien physique de communication, l'information binaire transmise (démodulation).

Ici, il s'agit d'un modem, avec pour support de communication la ligne téléphonique, travaillant à un débit maximal de 300 bits par seconde. Elle présente, en effet, l'avantage de générer des signaux à enveloppes constantes, ce qui est très intéressant notamment en présence de non linéarités dans une chaîne de transmission. La modulation de fréquence présente dans la recommandation V21 est une modulation de fréquence à 2 états (2-FSK). Les bits 0 et 1 à transmettre sont transformés en morceaux de cosinus de fréquences différentes pour former le signal transmis.

La fréquence moyenne nominale est de $F_c = 1080Hz$. L'excursion de fréquence de $\pm\Delta f = \pm 100Hz$. La fréquence la plus élevée ($F_0 = F_c + \Delta f = 1180Hz$) devra correspondre à l'état binaire 0, tandis que la plus faible ($F_1 = F_c - \Delta f = 980Hz$) devra correspondre à l'état binaire 1. La première étape consistera à former le signal 2-FSK à transmettre à partir d'un fichier d'information binaire (modulation). A partir d'un signal reçu abimé par un canal à bruit additif blanc et Gaussien, il s'agira ensuite de mettre en place le modem de réception permettant de retrouver l'information binaire.

Plusieurs versions du récepteur devront être implantées. Une première version procédera par filtrage. Cette version nécessite l'utilisation de fréquences suffisamment éloignées pour coder les bits 0 et 1. Elle n'est donc pas utilisable dans le contexte de la norme V21 souhaitée. On implantera donc une deuxième version de récepteur étant capable de travailler avec des fréquences proches (occupation spectrale plus faible pour le signal généré).

2 Modem de fréquence - Démodulation par filtrage

2.1 Construction du signal modulé en fréquence

La première étape du projet consiste à réaliser la modulation de fréquence, i.e. transformer l'information binaire à transmettre en un signal modulé en fréquence (exemple sur la figure 3). Le

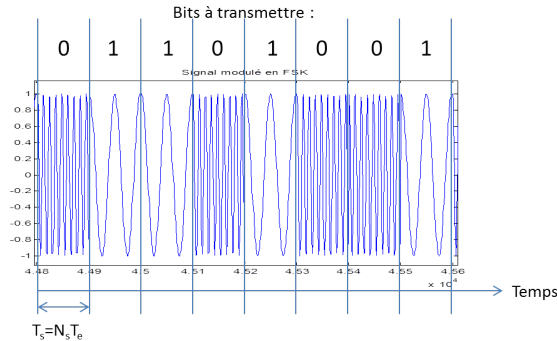


FIGURE 1 – Signal modulé en fréquence

signal modulé en fréquence $x(t)$ est généré de la manière suivante :

$$x(t) = (1 - NRZ(t)) \times \cos(2\pi F_0 t + \phi_0) + NRZ(t) \times \cos(2\pi F_1 t + \phi_1) \quad (1)$$

où $NRZ(t)$ est un signal de type NRZ polaire formé à partir de la suite de bits à transmettre en codant les 0 et les 1 par des niveaux 0 et 1 de durée T_s secondes. ϕ_0 et ϕ_1 sont des variables aléatoires indépendantes uniformément réparties sur $[0, 2\pi]$ qui peuvent être obtenues sous matlab en utilisant $rand*2*pi$.

2.1.1 Génération du signal NRZ

1. Nous voulons un débit $f_s = 300\text{bits/s}$. Ainsi, le nombre d'échantillons N_s à générer par niveau 0 ou 1 de durée $T_s = 1/f_s$ vaut :

$$N_s = \frac{f_e}{f_s} = \frac{48000}{300} = 160$$

2. Signal NRZ pour 50 bits.

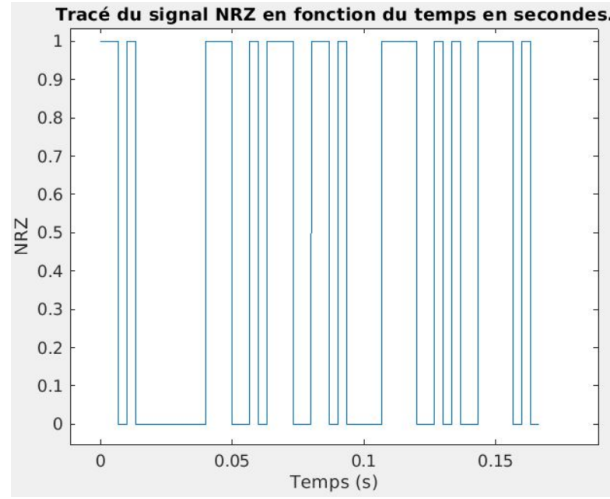


FIGURE 2 – Signal NRZ pour 50 bits.

3. Estimation et tracé de la densité spectrale de puissance du signal NRZ, sachant que la densité spectrale de puissance théorique d'un tel signal est donnée par :

$$S_{NRZ}(f) = \frac{1}{4}T_s \text{sinc}^2(\pi f T_s) + \frac{1}{4}\delta(f)$$

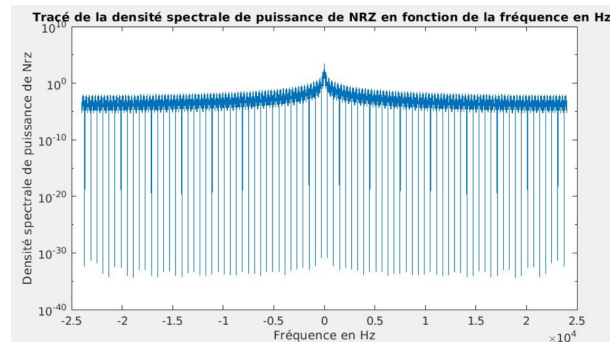


FIGURE 3 – Densité spectrale de puissance de NRZ entre $-f_e/2$ et $f_e/2$ (échelle log)

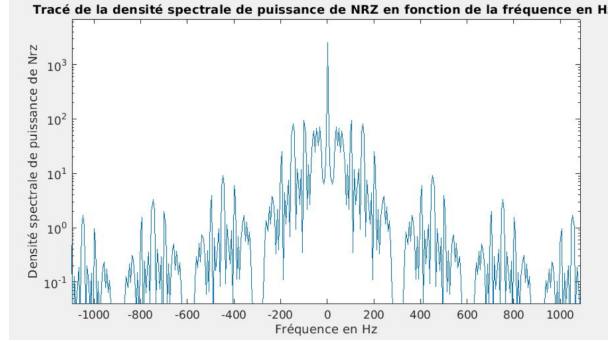


FIGURE 4 – Densité spectrale de puissance de NRZ zoomée au centre.

Nous remarquons à travers la figure 3, un pic à 0 Hz qui correspond au dirac de la formule admise.

Nous remarquons la forme d'un sinus cardinal à travers la figure 4. La déformation de ce tracé est dû au carré présent sur le sinus cardinal dans la formule.

2.1.2 Génération du signal modulé en fréquence

1. Le signal modulé en fréquence $x(t)$ est généré de la manière suivante en matlab :

```
phase0=rand*2*pi;
phase1=rand*2*pi;
x=(1-Nrz).*cos(2*pi*F0*temps+phase0)+Nrz.*cos(2*pi*F1*temps+phase1);
```

FIGURE 5 – Signal modulé en fréquence sur maltab.

2. Tracé du signal $x(t)$ avec une échelle temporelle en seconde.

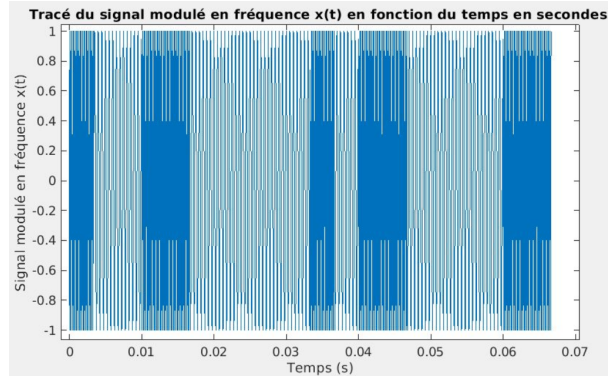


FIGURE 6 – Tracé du signal $x(t)$ en fonction du temps (s) pour 20bits.

Comme on peut le remarquer, la présence de bleu foncé sur le tracé correspond au haute fréquence $f_0 = 6000Hz$, représentant le bit 0, tandis que la partie plus claire représente la fréquence $f_1 = 2000Hz$, correspondant au bit 1.

3. Calculez la densité spectrale de puissance du signal modulé en fréquence $x(t)$ en fonction de celle du signal NRZ que l'on notera $S_{NRZ}(f)$.

On calcule d'abord la fonction d'autocorrélation du signal, en utilisant la formule ci-dessous :

$$R_x(\tau) = E[x(t)x^*(t - \tau)] \quad (2)$$

$$\begin{aligned}
R_x(\tau) &= E[(1 - NRZ(t)) \cos(2\pi f_0 t + \phi_0) + NRZ(t) \cos(2\pi f_1 t + \phi_1)] \\
&\quad * [(1 - NRZ(t - \tau)) \cos(2\pi f_0(t - \tau) + \phi_0) + NRZ(t - \tau) \cos(2\pi f_1(t - \tau) + \phi_1)] \\
&= E\left[\frac{1}{2}(1 - NRZ(t))(1 - NRZ(t - \tau)) \cos(2\pi f_0 \tau)\right] + \\
&\quad E[(1 - NRZ(t))NRZ(t - \tau) (\cos(2\pi f_0 t - 2\pi f_1(t - \tau) + \phi_0 - \phi_1)) + (\cos(2\pi f_0 t - 2\pi f_1(t - \tau) + \phi_0 + \phi_1))] \\
&\quad E[NRZ(t)(1 - NRZ(t - \tau)) (\cos(2\pi f_1 t - 2\pi f_0(t - \tau) + \phi_1 - \phi_0)) + (\cos(2\pi f_1 t - 2\pi f_0(t - \tau) + \phi_1 + \phi_0))] \\
&\quad + E\left[\frac{1}{2}NRZ(t)NRZ(t - \tau) \cos(2\pi f_1 \tau)\right]
\end{aligned}$$

Sachant que la moyenne d'un cosinus est nulle sur une période, la moyenne de NRZ(t) vaut $E[NRZ(t)] = \frac{1}{2}$ et que :

$$R_{NRZ}(\tau) = E[NRZ(t)NRZ(t - \tau)] \quad (3)$$

On applique la transformée de Fourier à l'autocorrélation pour obtenir la DSP du signal :

$$S_x(f) = TF[R_x(\tau)] \quad (4)$$

D'où,

$$\begin{aligned}
S_x(f) &= \frac{1}{4}S_{NRZ}(f) * [\delta(f + f_1) + \delta(f - f_1) + \delta(f + f_2) + \delta(f - f_2)] \\
&= \frac{1}{4} \times (S_{NRZ}(f + f_1) + S_{NRZ}(f - f_1) + S_{NRZ}(f + f_0) + S_{NRZ}(f - f_0))
\end{aligned}$$

On constate donc grâce à l'expression de $S_{NRZ}(f)$ que la densité spectrale de $x(t)$ est composée de 4 diracs au fréquence $f_1, -f_1, f_0, -f_0$ et de sinus cardinal au carré.

4. Tracé de la densité spectrale de puissance du signal modulé en fréquence, en utilisant un périodogramme et avec une échelle fréquentielle en Hz :

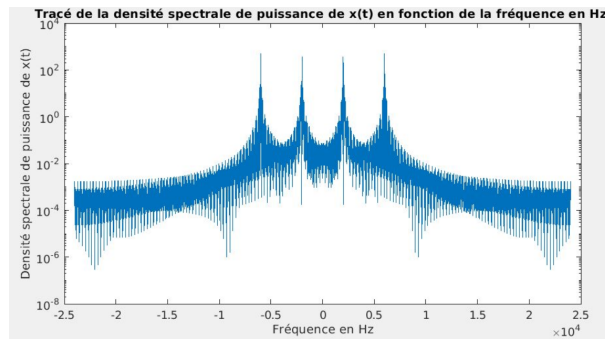


FIGURE 7 – Densité spectrale de puissance de $x(t)$ en fonction de la fréquence (Hz) (en échelle log).

On peut observer ci-dessus 4 diracs en $f_0, -f_0, f_1$ et $-f_1$. Lorsqu'on zoom, on pourra remarquer le même tracé observé dans la figure 4. Ces observations sont donc en total adéquation avec ce que l'on a calculé précédemment.

2.2 Canal de transmission à bruit additif, blanc et Gaussien

Nous allons considérer que le canal de propagation ajoute au signal émis un bruit que l'on suppose blanc et Gaussien et qui modélise les perturbations introduites.

La puissance du bruit Gaussien à ajouter devra être déduite du rapport signal sur bruit (SNR : Signal to Noise Ratio) souhaité pour la transmission donné en dB :

$$SNR_{dB} = 10 \log_{10} \frac{P_x}{P_b}$$

où P_x représente la puissance du signal modulé en fréquence et P_b la puissance du bruit ajouté.

2.3 Démodulation par filtrage

La figure 8 présente le récepteur implanté pour retrouver, à partir du signal modulé en fréquence bruité, le message binaire envoyé.

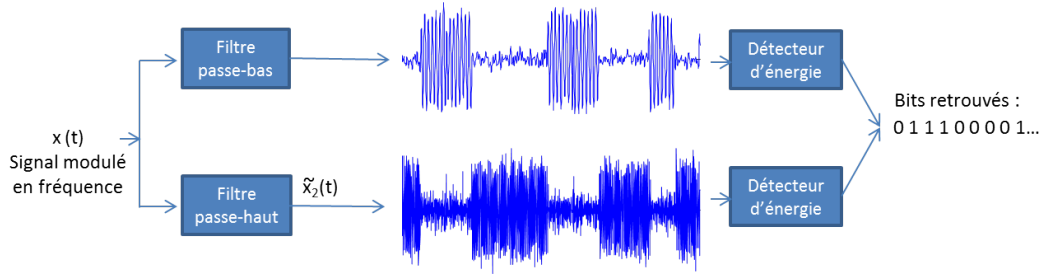


FIGURE 8 – Démodulation par filtrage.

Un filtre passe-bas permet de filtrer les morceaux de cosinus à la fréquence $F_0 = 6000\text{Hz}$, tandis qu'un filtre passe-haut permet de filtrer les morceaux de cosinus à la fréquence $F_1 = 2000\text{Hz}$. Une détection d'énergie réalisée tous les T_s secondes permet de récupérer, à partir des signaux filtrés, les bits 0 et 1 transmis.

2.3.1 Synthèse du filtre passe-bas

Nous avons utilisé le filtre passe-bas réalisé dans le TP2 de formule :

$$h(t) = 2 \frac{f}{F_e} \text{sinc} \left(2\pi k \frac{f}{F_e} t \right)$$

Nous avons décidé d'utiliser une fréquence de coupure $f_c = \frac{f_0 + f_1}{2}$, et d'ordre = 121 ce qui permet d'avoir un filtre précis tout en ayant un temps de calcul raisonnable.

Nous remarquons dans la figure 10, ci-dessous, que le filtre passe-bas sélectionne uniquement les basses fréquences permettant de conserver les fréquences $-f_1$ et f_1 .

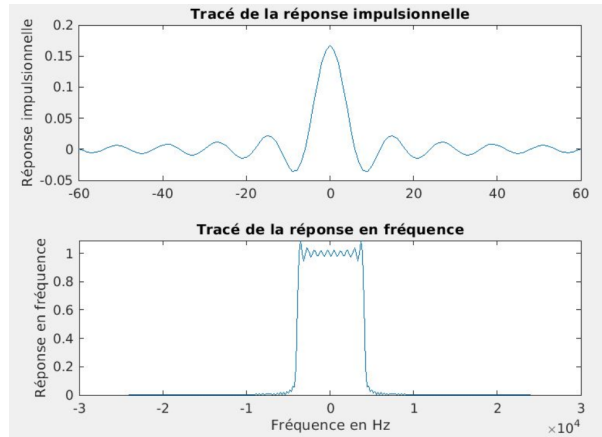


FIGURE 9 – Réponse impulsionnelle et en fréquence du filtre passe-bas.

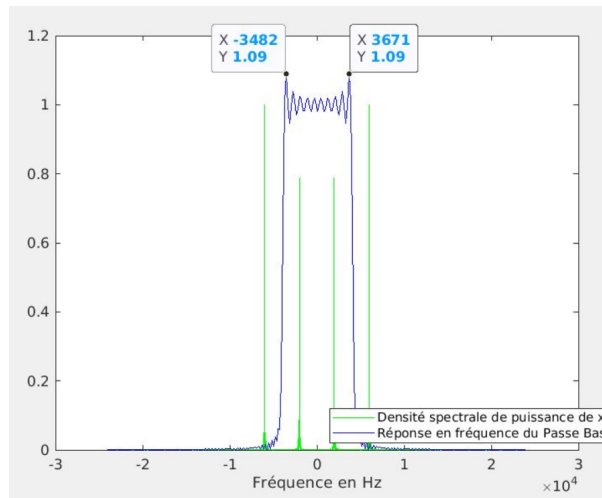


FIGURE 10 – Superposition de la densité spectrale de puissance de $x(t)$ et de la réponse fréquentielle du passe-bas.

2.3.2 Synthèse du filtre passe-haut

La réponse impulsionnelle du filtre passe-haut correspond à l'opposé de celle du filtre passe-bas, tout en rajoutant 1 à la valeur centrale. Sur Matlab, nous avons procédé de la manière suivante afin de la calculer.

```
% Calcul de la réponse impulsionnelle du filtre Passe Haut
Passe_Haut = -Passe_Bas;
Passe_Haut(Ord_div + 1) = 1 - Passe_Bas(Ord_div+ 1);
```

FIGURE 11 – Calcul de la réponse impulsionnelle du passe-haut sous Matlab.

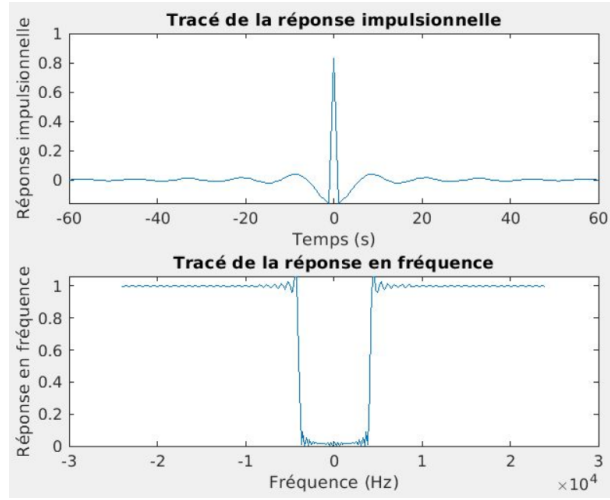


FIGURE 12 – Réponse impulsionnelle et en fréquence du filtre passe-haut.

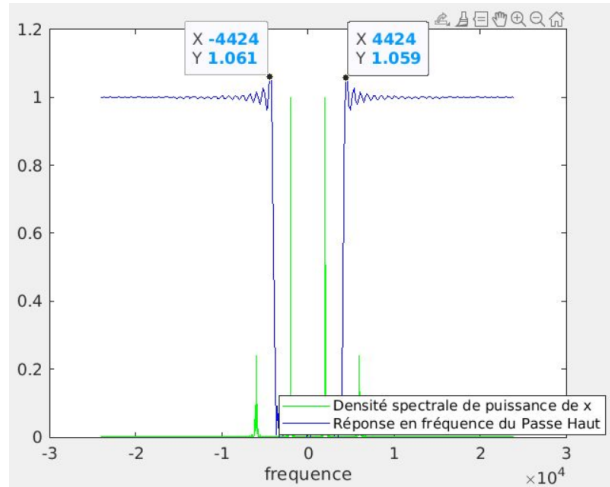


FIGURE 13 – Superposition de la densité spectrale de puissance de $x(t)$ et de la réponse fréquentielle du passe-haut.

Nous remarquons dans la figure 13 que le filtre passe-haut sélectionne uniquement les hautes fréquences permettant de conserver les fréquences $-f_0$ et f_0 .

2.3.3 Filtrage, tracés, commentaires sur les résultats obtenus

On peut remarquer sur la figure 14, qu'après filtrage du signal bruité, on retrouve exactement le même signal émis non bruité, sachant que la puissance du bruit est de $P_{bruit} = 15$ elle-même relativement importante.

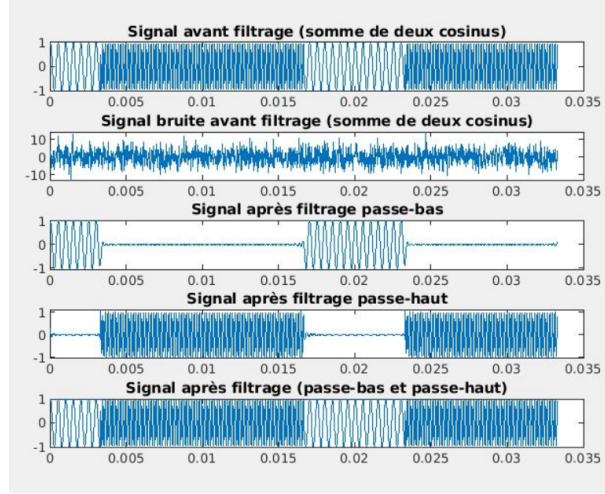


FIGURE 14 – Signal bruité avant et après filtrage pour 10 bits et une puissance de bruit de 15 dB

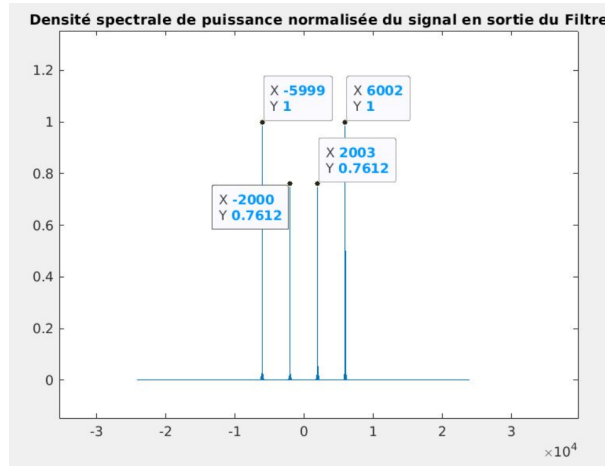


FIGURE 15 – Densité spectrale de puissance normalisée de $x(t)$ après filtrage.

Nous retrouvons sur la figure 15 les 4 diracs en f_0 , $-f_0$, f_1 et $-f_1$, présents initialement dans la figure 7. Cependant, pour mieux voir les sinus cardinaux, il suffit de zoomer près de l'un des diracs.

2.3.4 Détection d'énergie

Nous avons créé une matrice dont les lignes correspondent aux bits. Il y a donc N_{bit} lignes et N_s colonnes. Nous sommes ensuite sur les colonnes le carré de chaque valeur. Nous regardons ensuite pour chaque valeur du vecteur si elle est supérieure à un seuil. Nous avons choisi comme seuil $\frac{\min(\text{Energie}) + \max(\text{Energie})}{2}$.

Si la valeur est inférieure au seuil, alors c'est un 0. Sinon, c'est un 1.

Voici notre solution Matlab pour implanter le détecteur d'énergie :

```

Matrice=reshape(ybas,Ns,Nbbit);
Energie=sum(Matrice.^2);
YenerBass=find((min(Energie)+max(Energie))/2<Energie);
Energie = zeros(1, Nbbit);
Energie(YenerBass)=1;
Taux_erreur_Filtre=length(find(Energie~=r))/Nbbit

```

FIGURE 16 – Solution Matlab pour la détection d'énergie.

Après avoir reconstitué les 6 images fournies avec notre code matlab, nous avons obtenu l'image de la figure 17. Qui est ce monsieur ? On ne sait pas... Par contre où se trouve l'image ? Elle se trouve à coté du garage à vélo.



FIGURE 17 – Image reconstituée.

2.4 Application de la recommandation V21

La recommandation V21 conseille une fréquence f_0 de 1180 Hz et une fréquence f_1 de 980 Hz. A ces fréquences, la fréquence de coupure est très proche de f_0 et de f_1 . Cela demanderait donc un ordre très important pour avoir des résultats probants mais nécessiterait en contre-partie beaucoup trop de temps de calcul et de mémoire. Lorsqu'on essaie avec l'ordre qu'on avait choisi de 121, nous obtenons un taux d'erreur de 73% pour 10000 bits et pour $SNR_{db} = 50db$.

3 Modem de fréquence V21 - Démodulateur FSK

3.1 Démodulateur FSK - Contexte de synchronisation idéale

La figure 18 présente le récepteur modifié implanté afin de retrouver, à partir du signal modulé en fréquence suivant la recommandation V21, le message binaire envoyé.

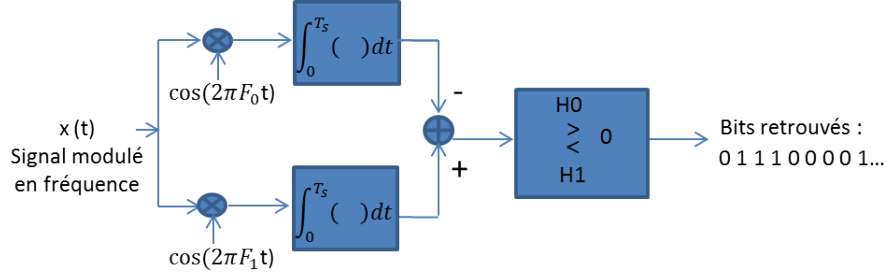


FIGURE 18 – Démodulation FSK. Synchronisation supposée idéale.

Déterminons les 2 cas possibles.

Soient $(i, j) \in \{0, 1\}$

Si $i \neq j$

$$\begin{aligned}
 \int_0^{T_s} \cos(2\pi f_i t + \phi_i) \cos(2\pi f_j t + \phi_j) dt &= \frac{1}{2} \times \int_0^{T_s} (\cos(2\pi t (f_i + f_j) + \phi_j + \phi_i) + \cos(2\pi t (f_i - f_j) + \phi_i - \phi_j)) dt \\
 &= \frac{1}{4\pi} \times \left(\frac{\sin(2\pi T_s (f_i + f_j) + \phi_j + \phi_i)}{f_i + f_j} - \frac{\sin(\phi_j + \phi_i)}{f_i + f_j} \right. \\
 &\quad \left. + \frac{\sin(2\pi T_s (f_i - f_j) + \phi_i - \phi_j)}{f_i - f_j} - \frac{\sin(\phi_i - \phi_j)}{f_i - f_j} \right)
 \end{aligned} \tag{5}$$

Si $i = j$,

$$\int_0^{T_s} \cos^2(2\pi f_i t + \phi_i) dt = \frac{1}{2} \times \frac{\sin(4\pi f_i T_s + 2\phi_i)}{4\pi f_i} + \frac{T_s}{2} \tag{6}$$

Rappelons l'expression de $x(t)$:

$$x(t) = (1 - NRZ(t)) \times \cos(2\pi F_0 t + \phi_0) + NRZ(t) \times \cos(2\pi F_1 t + \phi_1)$$

Si on enlève les NRZ pour simplifier on obtient :

$$x(t) = \cos(2\pi f_0 t + \phi_0) + \cos(2\pi f_1 t + \phi_1)$$

Lorsqu'on multiplie par $\cos(2\pi f_0 t + \phi_0)$ et qu'on intègre on obtient d'après (5) et (6) :

$$\begin{aligned}
 \int_0^{T_s} x(t) \cos(2\pi f_0 t + \phi_0) dt &= \frac{1}{2} \times \frac{\sin(4\pi f_0 T_s + 2\phi_0)}{4\pi f_0} + \frac{T_s}{2} + \frac{1}{4\pi} \times \left(\frac{\sin(2\pi T_s (f_0 + f_1) + \phi_1 + \phi_0)}{f_0 + f_1} - \frac{\sin(\phi_0 + \phi_1)}{f_0 + f_1} \right. \\
 &\quad \left. + \frac{\sin(2\pi T_s (f_0 - f_1) + \phi_0 - \phi_1)}{f_0 - f_1} - \frac{\sin(\phi_0 - \phi_1)}{f_0 - f_1} \right)
 \end{aligned} \tag{7}$$

De même, lorsqu'on multiplie par $\cos(2\pi f_1 t + \phi_1)$ et qu'on intègre on obtient d'après (5) et (6) :

$$\int_0^{T_s} x(t) \cos(2\pi f_1 t + \phi_1) dt = \frac{1}{2} \times \frac{\sin(4\pi f_1 T_s + 2\phi_1)}{4\pi f_1} + \frac{T_s}{2} + \frac{1}{4\pi} \times \left(\frac{\sin(2\pi T_s(f_0 + f_1) + \phi_1 + \phi_0)}{f_0 + f_1} - \frac{\sin(\phi_0 + \phi_1)}{f_0 + f_1} + \frac{\sin(2\pi T_s(f_0 - f_1) + \phi_0 - \phi_1)}{f_0 - f_1} - \frac{\sin(\phi_0 - \phi_1)}{f_0 - f_1} \right) \quad (8)$$

Lorsqu'on fait (8) - (7), on obtient :

$$(8) - (7) = \frac{1}{8\pi} \times \left(\frac{\sin(4\pi f_1 T_s + 2\phi_1)}{f_1} - \frac{\sin(4\pi f_0 T_s + 2\phi_0)}{f_0} \right) \quad (9)$$

Malheureusement, nous n'avons pas réussi à trouver le lien entre le signe de l'expression (9) et le choix des bits à retrouver.

Voici notre implantation Matlab de ce démodulateur FSK :

```
xhaut=xbruite.*cos(2*pi*F0*temps+phase0);
xbas=xbruite.*cos(2*pi*F1*temps+phase1);

Matrice_int_haut=reshape(xhaut,Ns,Nbbit);
Vect_int_haut=sum(Matrice_int_haut);

Matrice_int_bas=reshape(xbas,Ns,Nbbit);
Vect_int_bas=sum(Matrice_int_bas);

Vect_diff= Vect_int_haut-Vect_int_bas;

Indice_1=find(0>Vect_diff);
Energie_demodu = zeros(1, Nbbit);
Energie_demodu(Indice_1)=1;
```

FIGURE 19 – Solution Matlab pour le démodulateur FSK en synchronisation idéale.

Ici, comme avec la détection d'énergie, nous créons une matrice de Nbbits lignes et Ns colonnes. En discret, l'intégrale est équivalent à une somme; on fait donc la somme sur les colonnes pour chaque ligne.

Ensuite, on crée un vecteur Vect_diff qui est la différence des deux vecteurs créés par l'étape précédente. Pour chaque valeur de ce vecteur, si c'est positif alors le bit correspondant est 0 sinon le bit correspondant est 1.

Pour une puissance de bruit $SNR_{db} = 50$, pour 1000 bits générés aléatoirement et pour des fréquences imposées par la norme V21, nous retrouvons bien un taux d'erreur binaire nul.

3.2 Démodulateur FSK avec gestion d'une erreur de synchronisation de phase porteuse

Lorsqu'on change la phase à la réception, le démodulateur précédent ne fonctionne plus.

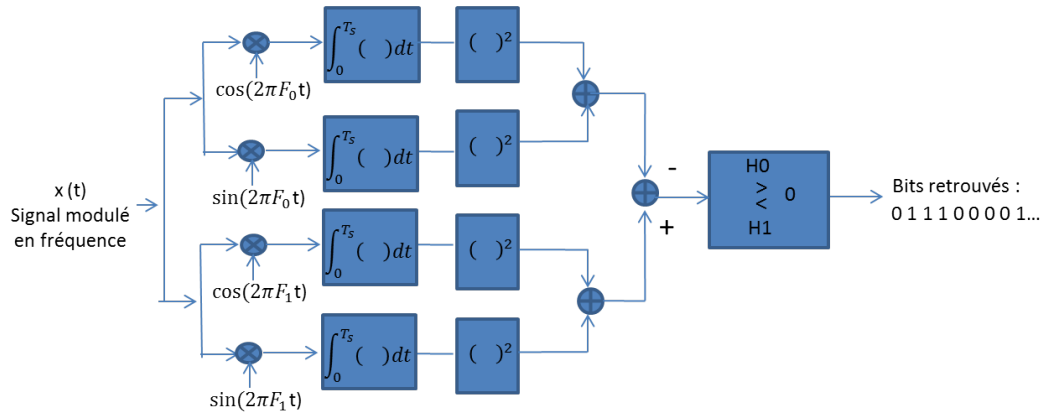


FIGURE 20 – Démodulation FSK - Gestion d'une erreur de phase porteuse.

Pour l'implantation Matlab de ce démodulateur FSK avec gestion de l'erreur de phase porteuse, nous avons procédé comme précédemment en rajoutant 2 vecteurs résultant du produit du signal $x(t)$ et du sinus. Puis contrairement à précédemment, nous avons mis au carré les vecteurs obtenus après la sommation.

Après avoir effectué toutes ces manipulations, nous retrouvons bien un taux d'erreur binaire de 0 même après avoir rajouté une erreur de phase sur les sinus et cosinus.

4 Conclusion

Grâce à ce magnifique projet, nous avons pu découvrir nos premiers algorithmes pour envoyer et traiter un signal composé de bits. Le but final était de déchiffrer six photos codées initialement. Nous avons réussi à atteindre cet objectif.

Pour cela, nous avons d'abord dû moduler le signal initial pour le transformer en un signal modulé prêt à la transmission. Nous avons ensuite bruité ce signal afin de rendre le signal plus réaliste.

Nous avons par la suite utilisé 3 manières de traiter le signal. La première était d'utiliser soit un filtre passe-haut, soit un filtre passe-bas puis d'utiliser un détecteur d'énergie pour déterminer la valeur du bit. La deuxième méthode était d'utiliser un démodulateur FSK, dans le contexte de synchronisation idéale. La troisième méthode consistait à utiliser un démodulateur FSK avec gestion d'une erreur de synchronisation de phase porteuse.

Ce projet nous a permis de mettre en application les notions vues en cours et en TP, et d'avoir un aperçu concret du traitement de signal dans la vie professionnelle.

5 Références

TP1 : Signaux et Spectres
TP2 : Filtrage numérique

Traitement du Signal Numérique, Nathalie Thomas
Traitement du Signal, Jean-Yves Tournet