



Projet de traitement du signal : Simulation d'un Modem

WU Julien
MALINGE Romain

Département Sciences du Numérique - Première année
2022-2023

Table des matières

1	Introduction	4
2	Modem de fréquence	4
2.1	Construction du signal modulé en fréquence	4
2.1.1	Génération du signal NRZ	5
2.1.2	Génération du signal modulé en fréquence	7
3	Canal de transmission à bruit additif, blanc et Gaussien	12
4	Démodulation par filtrage	12
4.1	Synthèse du filtre passe-bas	13
4.2	Synthèse du filtre passe-haut	13
4.3	Résultats obtenus avec un ordre des filtres de 61	14
4.4	Modification de l'ordre des filtres	16
4.5	Utilisation des fréquences de la recommandation V21	18
5	Démodulateur de fréquence adapté à la norme V21	19
5.1	Contexte de synchronisation idéale	19
5.1.1	Principe de fonctionnement de ce récepteur	19
5.1.2	Résultats obtenus	21
5.2	Gestion d'une erreur de synchronisation de phase porteuse	22
5.2.1	Impact d'une erreur de phase porteuse sur le modem implanté précédemment	22
5.2.2	Principe de fonctionnement de cette nouvelle version du modem de fréquence	23
5.2.3	Résultats obtenus	25
5.3	Reconstitution d'une image	26
5.4	Présentation de Mr. Escande	28
5.4.1	Biographie	28
5.4.2	Postérité	28
5.4.3	Publications	28
6	Conclusion	29

Table des figures

1	Signal modulé en fréquence	4
2	Signal NRZ en fonction du temps	5
3	DSP de NRZ (théorique et estimée)	6
4	Signal modulé en fréquence x en fonction du temps	7
5	DSP du signal modulé en fréquence x	11
6	Démodulation par filtrage.	12
7	Réponse fréquentielle (à droite) et impulsionnelle (à gauche) du filtre passe-bas et passe-haut de fréquence de coupure F_c	14
8	Signal et DSP en sortie du filtre passe-pas et passe-haut d'ordre 61 et de fréquence de coupure F_c	15
9	DSP du signal modulé et réponse fréquentielle du filtre PB et PH en échelle logarithmique . .	15
10	Signal et DSP en sortie du filtre passe-pas et passe-haut d'ordre 201 et de fréquence de coupure F_c	17
11	Signal et DSP en sortie du filtre passe-pas et passe-haut d'ordre 201 et de fréquence de coupure F_c	17
12	Signal et DSP en sortie du filtre passe-pas et passe-haut d'ordre 61 dans les recommandations V21	18
13	Démodulation FSK. Synchronisation supposée idéale.	19
14	Démodulation FSK - Gestion d'une erreur de phase porteuse.	22
15	Image recollée à partir de 6 images démodulées	27

Remerciement

Un grand merci à Nathalie Thomas pour son cours très complet sur le traitement numérique du signal. Également une attention toute particulière pour notre encadrant de TP Samy Labsir qui nous a bien aidé pendant nos moments de doute.

1 Introduction

L'objectif de ce projet était de mettre en place une chaîne de communication qui fait intervenir un modem. Cette chaîne de communication prend en entrée un flux binaire qui va être changé en signal sinusoïdale avec un modulateur. Du bruit est ensuite ajouté pour simuler les perturbations liées à la transmission du signal. Finalement, le signal sera reconverti en flux binaire avec le démodulateur.

Dans ce projet, nous allons nous intéresser tout d'abord à une démodulation par filtrage qui utilise deux filtres, un passe-bas et un passe-haut, supposés idéal. Cette démodulation suppose un écart assez élevé entre les fréquences prises pour la modulation du signal, ce qui n'est pas le cas pour les recommandations V21. Cela nous pousse alors à envisager une autre méthode de démodulation : un démodulateur FSK qui permet alors de démoduler correctement le signal modulé dans la recommandation V21. Cependant, ce dernier démodulateur ne prend pas en compte les problèmes de désynchronisation de phase porteuse, un dernier démodulateur permet alors de gérer ce problème tout en étant dans la recommandation V21.

Finalement, nous allons pouvoir démoduler les signaux contenant une image et reconstituer l'image à partir de ces 6 morceaux à l'aide du dernier démodulateur.

2 Modem de fréquence

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 1).

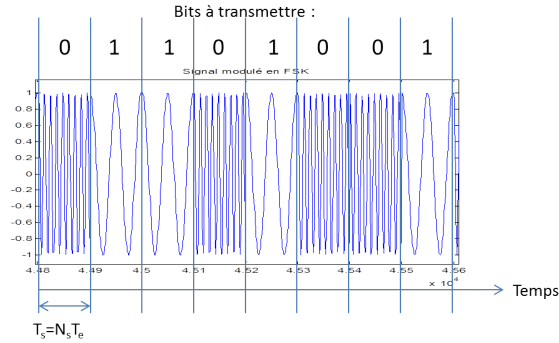


FIGURE 1 – Signal modulé en fréquence

Le 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. Génération du signal NRZ à partir de la suite de bits à transmettre

```
Signal_Binaire = randi(2, 1, N) -1;  
% Avec N le nombre de bits dans le signal binaire  
  
Fe = 48000;           % Frequence d'echantillonnage  
Fs = 300;             % Debit binaire par seconde demande  
Ts = 1 / Fs;          % Periode par niveau  
Te = 1 / Fe;          % Periode d'echantillonnage  
Ns = floor(Ts*Fe);     % Nombre d'echantillons par niveau  
  
NRZ = kron(Signal_Binaire, ones(1, Ns));
```

2. Tracé du signal NRZ en fonction du temps

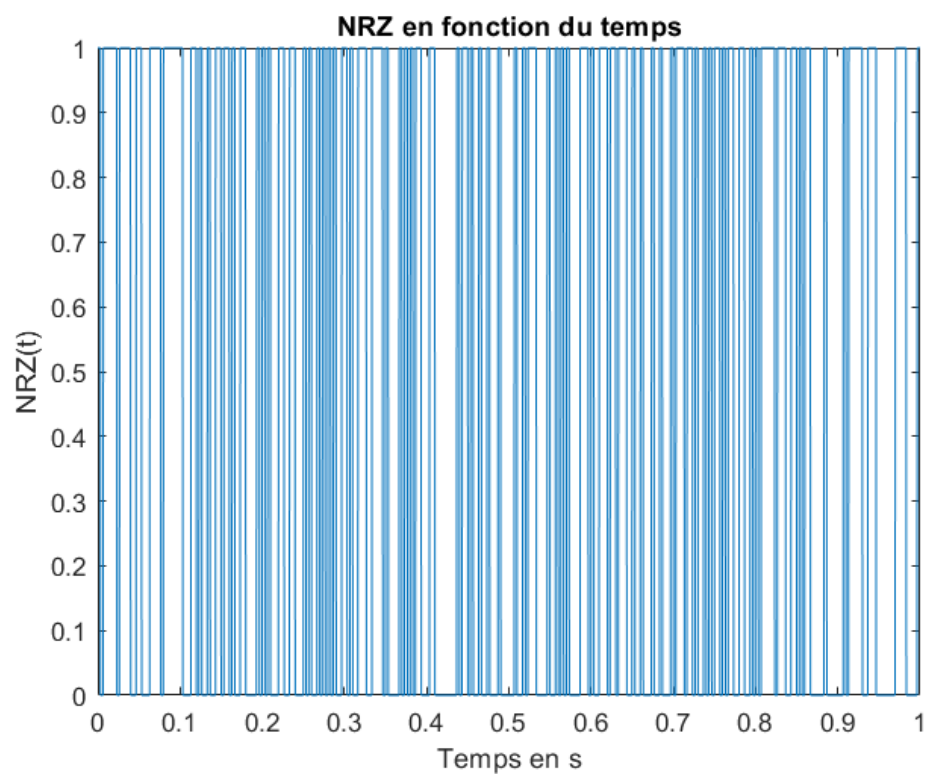


FIGURE 2 – Signal NRZ en fonction du temps

3. Tracé de la DSP de NRZ théorique et estimée en fonction de la fréquence

On utilise la fonction de Matlab `pwelch` pour pouvoir calculer numériquement la densité de puissance du signal NRZ et à l'aide de la formule théorique de la DSP de NRZ donnée, nous pouvons alors tracer la DSP estimée et théorique de NRZ.

```
% Calcul de la DSP de NRZ par la fonction pwelch
DSP_estime = pwelch (NRZ, [], [], [], Fe, 'twosided');
% Nombre de ligne dans DSP_estime
taille = length(DSP_estime);
% Intervalle de frequence
frequence = linspace(0,Fe,size(DSP_estime,1));
% Formule donnee par le sujet
DSP_theorique = (1/4) * Ts * (sinc(pi * Ts * frequence)).^2 + (1/4)*
    dirac(frequence);
```

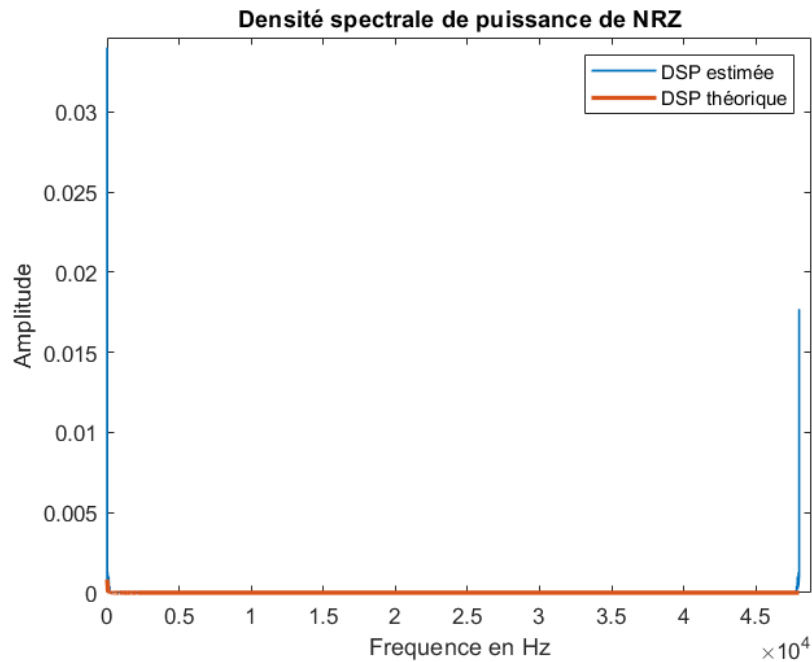


FIGURE 3 – DSP de NRZ (théorique et estimée)

2.1.2 Génération du signal modulé en fréquence

1. Génération du signal modulé à partir de NRZ

```
% Frequence pour le signal sinusoidal des 1 de NRZ
F0 = 6000;
% Frequence pour le signal sinusoidal des 0 de NRZ
F1 = 2000;
% Phase tire aleatoirement pour le signal sinusoidal des 0 de NRZ
psi0 = rand*2*pi;
% Phase tire aleatoirement pour le signal sinusoidal des 1 de NRZ
psi1 = rand*2*pi;
% Par definition dans le paragraphe 2.1
x = (1-NRZ) .* cos(2*pi*F0*temps + psi0) + NRZ .* cos(2*pi*F1*temps +
    psi1);
```

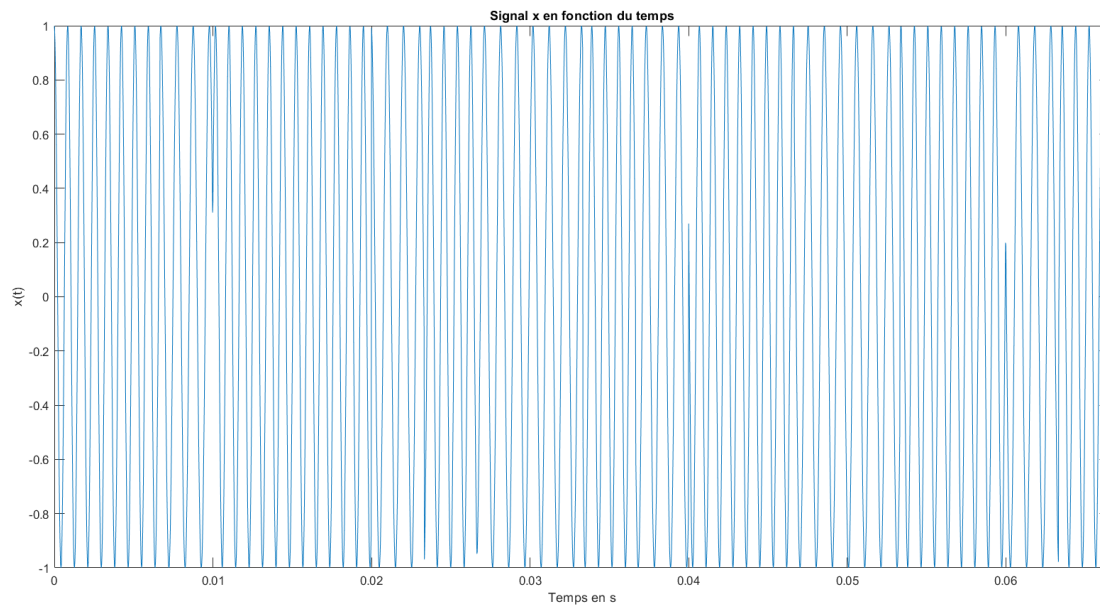


FIGURE 4 – Signal modulé en fréquence x en fonction du temps

2. Calcul de la DSP de $x(t)$

En partant de la définition du coefficient de corrélation, on a :

$$\begin{aligned} R_x(t) &= E\left(x(t) \times x(t - \tau)\right) \\ &= E\left(\left[(1 - NRZ(t)) \times \cos(2\pi F_0 t + \phi_0) + NRZ(t) \times \cos(2\pi F_1 t + \phi_1) \right] \right. \\ &\quad \left. \times \left[(1 - NRZ(t - \tau)) \times \cos(2\pi F_0(t - \tau) + \phi_0) + NRZ(t - \tau) \times \cos(2\pi F_1(t - \tau) + \phi_1) \right] \right) \end{aligned}$$

On pose les écritures suivantes pour plus de lisibilité :

$$\psi_0(t) = \cos(2\pi F_0 t + \phi_0) \quad \text{et} \quad \psi_1(t) = \cos(2\pi F_1 t + \phi_1)$$

$$\begin{aligned} R_x(t) &= E\left(\left[(1 - NRZ(t)) \times \psi_0(t) + NRZ(t) \times \psi_1(t) \right] \right. \\ &\quad \left. \times \left[(1 - NRZ(t - \tau)) \times \psi_0(t - \tau) + NRZ(t - \tau) \times \psi_1(t - \tau) \right] \right) \\ &= E\left((1 - NRZ(t))(1 - NRZ(t - \tau))\psi_0(t)\psi_0(t - \tau) + (1 - NRZ(t))NRZ(t - \tau)\psi_0(t)\psi_1(t - \tau) \right. \\ &\quad \left. + NRZ(t)(1 - NRZ(t - \tau))\psi_1(t)\psi_0(t - \tau) + NRZ(t)NRZ(t - \tau)\psi_1(t)\psi_1(t - \tau) \right) \\ &= E\left((1 - NRZ(t))(1 - NRZ(t - \tau))\psi_0(t)\psi_0(t - \tau) \right) + E\left((1 - NRZ(t))NRZ(t - \tau)\psi_0(t)\psi_1(t - \tau) \right) \\ &\quad + E\left(NRZ(t)(1 - NRZ(t - \tau))\psi_1(t)\psi_0(t - \tau) \right) + E\left(NRZ(t)NRZ(t - \tau)\psi_1(t)\psi_1(t - \tau) \right) \\ &= E\left((1 - NRZ(t))(1 - NRZ(t - \tau)) \right) E\left(\psi_0(t)\psi_0(t - \tau) \right) \\ &\quad + E\left((1 - NRZ(t))NRZ(t - \tau) \right) E\left(\psi_0(t)\psi_1(t - \tau) \right) \\ &\quad + E\left(NRZ(t)(1 - NRZ(t - \tau)) \right) E\left(\psi_1(t)\psi_0(t - \tau) \right) \\ &\quad + E\left(NRZ(t)NRZ(t - \tau) \right) E\left(\psi_1(t)\psi_1(t - \tau) \right) \end{aligned}$$

$$R_x(t) = E(A)E(B) + E(C)E(D) + E(E)E(F) + E(G)E(H)$$

Comme les termes $E(D)$ et $E(F)$ sont des espérances de cosinus, ils sont égales à 0 car l'intégrale d'un cosinus sur sa période est égale à 0. On va maintenant calculer les quatre termes $E(A)$, $E(B)$, $E(G)$ et $E(H)$:

$$\begin{aligned}
E(A) &= E\left((1 - NRZ(t)) \times (1 - NRZ(t - \tau)) \right) \\
&= E\left(1 - NRZ(t - \tau) - NRZ(t) + NRZ(t) \times NRZ(t - \tau) \right) \\
&= E\left(1 - 0,5 - 0,5 + NRZ(t) \times NRZ(t - \tau) \right) \\
&= 0 + E\left(NRZ(t) \times NRZ(t - \tau) \right) \\
&= R_{NRZ}(\tau)
\end{aligned}$$

$$\begin{aligned}
E(B) &= E\left(\psi_0(t) \psi_0(t - \tau) \right) \\
&= \frac{1}{2\pi} \int_0^{2\pi} \cos(2\pi F_0 t + \phi_0) \cos(2\pi F_0(t - \tau) + \phi_0) dt \\
&= \frac{1}{2\pi} \int_0^{2\pi} \frac{1}{2} \left(\cos(2\pi F_0(2t - \tau) + 2\phi_0) + \cos(2\pi F_0 \tau) \right) dt \\
&= \frac{1}{4\pi} \left(\int_0^{2\pi} \cos(2\pi F_0(2t - \tau) + 2\phi_0) dt + 2\pi \times \cos(2\pi F_0 \tau) \right) \\
&= \frac{1}{2} \cos(2\pi F_0 \tau) + 0 \quad (\text{L'intégrale d'un cosinus sur sa période est nulle})
\end{aligned}$$

$$\begin{aligned}
E(G) &= E\left(NRZ(t) \times NRZ(t - \tau) \right) \\
&= R_{NRZ}(\tau)
\end{aligned}$$

$$\begin{aligned}
E(H) &= E(B) \text{ avec } \phi_0 \leftarrow \phi_1 \text{ et } F_0 \leftarrow F_1 \\
&= \frac{1}{2} \cos(2\pi F_0 \tau) \text{ avec } \phi_0 \leftarrow \phi_1 \text{ et } F_0 \leftarrow F_1 \\
&= \frac{1}{2} \cos(2\pi F_1 \tau)
\end{aligned}$$

On finit par trouver le résultat en regroupant ces quatre termes :

$$\begin{aligned}
R_x(t) &= E(A)E(B) + 0 + 0 + E(G)E(H) \\
&= R_{NRZ}(\tau) \times \frac{1}{2} \cos(2\pi F_0 \tau) + R_{NRZ}(\tau) \times \frac{1}{2} \cos(2\pi F_1 \tau) \\
&= \frac{1}{2} R_{NRZ}(\tau) \left(\cos(2\pi F_0 \tau) + \cos(2\pi F_1 \tau) \right)
\end{aligned}$$

$$\begin{aligned}
S_x(f) &= TF(R_x(t)) \\
&= \frac{1}{2} TF(R_{NRZ}(\tau) \left(\cos(2\pi F_0 \tau) + \cos(2\pi F_1 \tau) \right)) \\
&= \frac{1}{2} S_{NRZ}(f) * \frac{1}{2} \left(\delta(f - F_0) + \delta(f + F_0) + \delta(f - F_1) + \delta(f + F_1) \right) \\
&= \frac{1}{4} \left(S_{NRZ}(f - F_0) + S_{NRZ}(f + F_0) + S_{NRZ}(f - F_1) + S_{NRZ}(f + F_1) \right)
\end{aligned}$$

D'où le résultat :

$$S_x(f) = \frac{1}{4} \left(S_{NRZ}(f - F_0) + S_{NRZ}(f + F_0) + S_{NRZ}(f - F_1) + S_{NRZ}(f + F_1) \right)$$

3. Tracé de la DSP du signal modulé en fréquence

De la même manière que pour NRZ, on utilise la fonction `pwelch` de Matlab puis on trace suivant l'axe temps défini précédemment.

```
% Calcul de la DSP de x par la fonction pwelch
DSP_estime_X = pwelch(x, [], [], [], 1/Te, 'twosided');
```

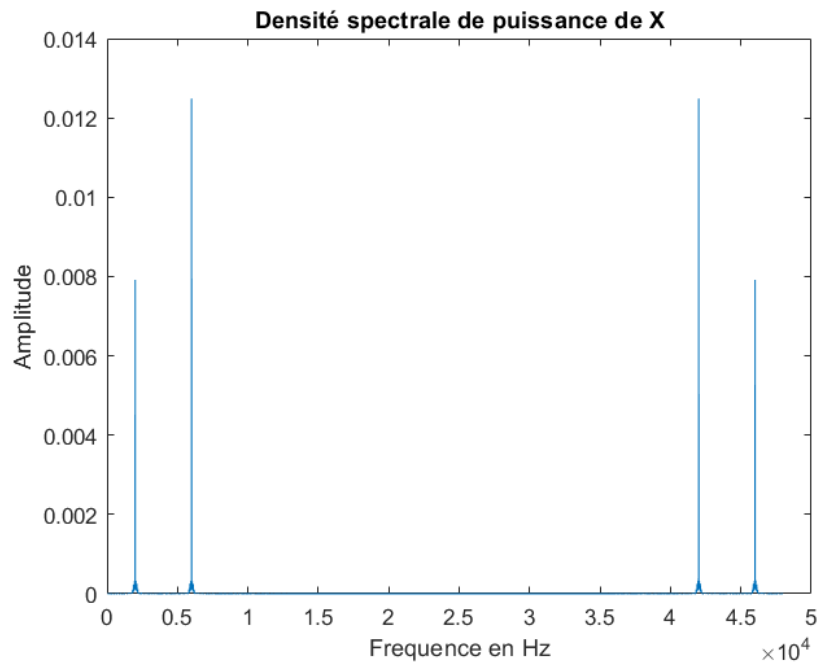


FIGURE 5 – DSP du signal modulé en fréquence x

Sur la Figure 5, nous apercevons bien deux pics en $F_0 = 6000$ Hz et en $F_1 = 2000$ Hz ce qui est normal au vu de la construction du signal $x(t)$

3 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.

Cela va être implémenté dans Matlab par les lignes suivantes :

```
% Sound to Noise Ratio choisi arbitrairement
SNR = 50;
% Puissance du signal
Px = mean(abs(x).^2);
% Puissance du bruit
Pb = Px * 10 ^ (-SNR / 10);
% Generation d'un bruit blanc gaussien
bruit = sqrt(Pb) * randn(1,N*Ns);
% Signal bruité
x_bruitee = x + bruit;
```

4 Démodulation par filtrage

La figure 6 présente le récepteur implanté pour retrouver, à partir du signal modulé en fréquence bruité, le message binaire envoyé.

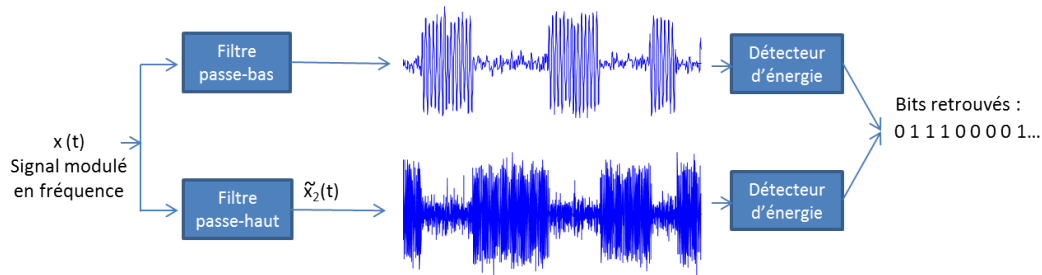


FIGURE 6 – Démodulation par filtrage.

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.

4.1 Synthèse du filtre passe-bas

Afin de synthétiser le filtre passe-bas, nous passons par le filtre passe-bas idéal, c'est à dire une porte centrée en 0 et de largeur $F_c/2$ avec $F_c = 4000\text{Hz}$ choisi de manière arbitraire, dont la réponse impulsionnelle s'obtient facilement par transformation de Fourier inverse. En Matlab, nous allons la synthétiser à l'aide de la fonction filter.

```
% Ordre du filtre
Ord = 61;
% Frequence de coupure
Fc = (F0 + F1) / 2; % egale a 4000 Hz dans notre cas.
% Nombre d'echantillons en retard du a l'ordre du filtre
Nbr_Echant_Retard = (Ord - 1) / 2;
% On utilise un filtre passe-bas ideal de frequence de coupure Fc
b_bas = (2 * Fc/Fe) * sinc(2 * (Fc/Fe) * (-(Ord-1)/2:1:(Ord-1)/2) );
% Decalage du signal module pour ne pas garder les zeros
x_bruitee_decaler = [x_bruitee , zeros(1, Nbr_Echant_Retard)];
% Signal en sortie du filtre passe bas nomme y_bas
y_bas = filter(b_bas, 1, x_bruitee_decaler);
y_bas = y_bas(1, Nbr_Echant_Retard+1:end);
```

4.2 Synthèse du filtre passe-haut

La réponse fréquentielle du filtre passe-haut idéal s'obtient à partir de la réponse fréquentielle du filtre passe-bas idéal selon la relation suivante :

$$H_{IPH}(\tilde{f}) = 1 - H_{IPB}(\tilde{f}) \quad \text{avec } \tilde{f} = \frac{f}{F_e} \text{ la fréquence normalisée}$$

Dans Matlab, nous écrirons le code suivant :

```
% Creation d'un dirac
dirac1 = zeros(1,Ord);
dirac1((Ord-1)/2 + 1) = 1;
% Definition de la reponse impulsionnelle du passe-haut ideal
b_haut = dirac1 - b_bas;
% Creation du filtre passe-haut
y_haut = filter(b_haut, 1, x_bruitee_decaler);
y_haut = y_haut(1, Nbr_Echant_Retard+1:end);
```

4.3 Résultats obtenus avec un ordre des filtres de 61

Dans le cas où l'ordre des filtres est fixé à 61, nous avons pu obtenir les résultats suivant :

1. Pour les réponse fréquentielles et impulsionnelles des filtres

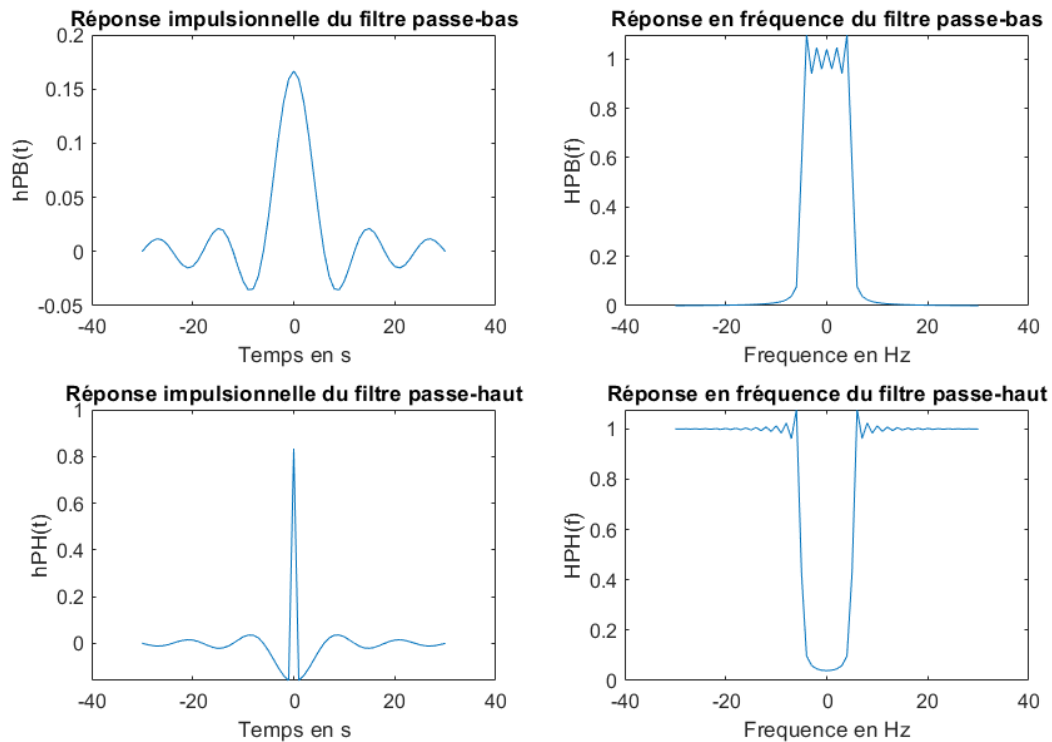


FIGURE 7 – Réponse fréquentielle (à droite) et impulsionnelle (à gauche) du filtre passe-bas et passe-haut de fréquence de coupure F_c

On oublie pas d'utiliser la fonction `fftshift` de Matlab pour pouvoir afficher entre $-\frac{F_c}{2}$ et $+\frac{F_c}{2}$ dans le cas contraire nous aurions un affichage entre 0 et F_c

```
% Affichage de la reponse impulsionnelle du passe-bas
plot((-1/2:1/2),b_bas)
% Affichage de la reponse frequentielle du passe-bas
plot((-1/2:1/2),abs(fftshift (fft(b_bas)) ) )
% Affichage de la reponse impulsionnelle du passe-haut
plot((-1/2:1/2),b_haut)
% Affichage de la reponse frequentielle du passe-haut
plot((-1/2:1/2),abs(fftshift (fft(b_haut)) ) )
```

2. Pour le signal en sortie des filtres et sa DSP correspondante

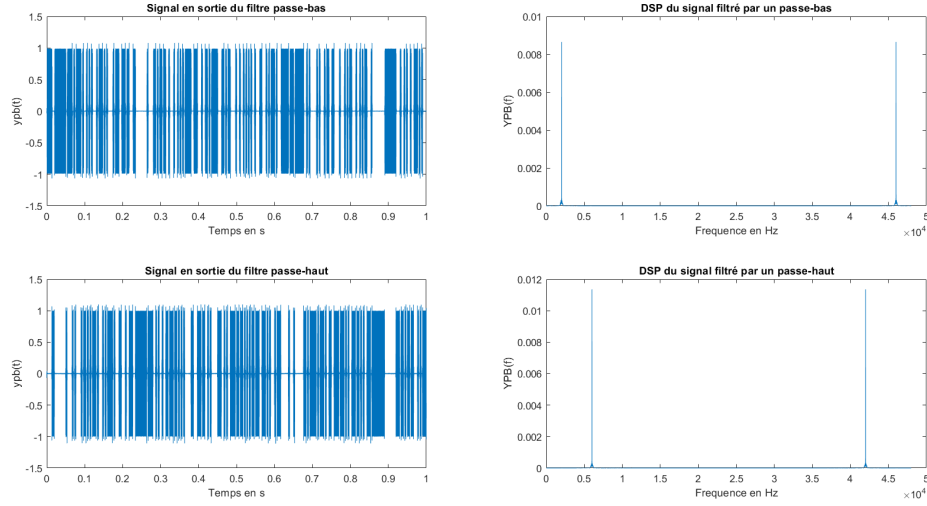


FIGURE 8 – Signal et DSP en sortie du filtre passe-pas et passe-haut d'ordre 61 et de fréquence de coupure F_c

Sur la Figure 8, nous pouvons remarquer que le choix arbitraire de $F_c = 4000Hz$ était pertinent, en effet, le filtre passe-haut permet de conserver uniquement la fréquence $F_0 = 6000Hz$ et le filtre passe-bas permet de conserver uniquement la fréquence $F_1 = 2000Hz$ ce qui était bien l'objectif des deux filtres. Cela est surtout visible sur la densité de puissance, mais il est aussi visible sur le graphique temporel où l'on observe des zéros. Une autre manière d'observer la pertinence des filtres est d'afficher la densité spectrale du signal modulé et la réponse fréquentielle des filtres sur le même graphique en échelle logarithmique (voir Figure 9).

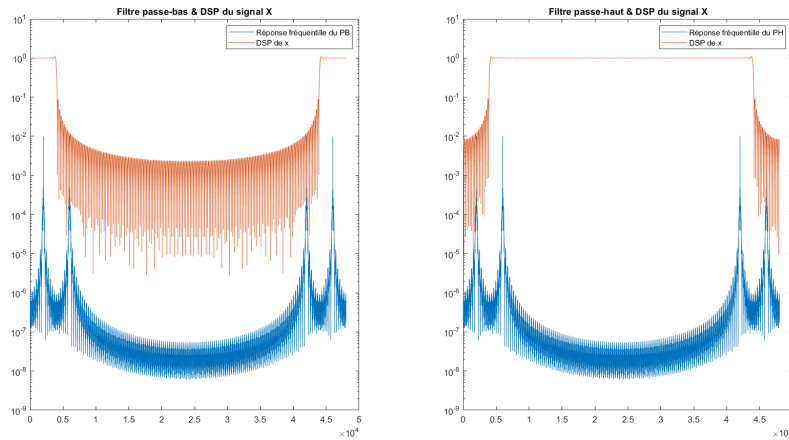


FIGURE 9 – DSP du signal modulé et réponse fréquentielle du filtre PB et PH en échelle logarithmique

Nous remarquons de plus que sans décaler le signal, les 30 premiers points sont proches de 0, cela est dû à l'ordre des filtres.

3. Détection d'énergie

Chaque signal filtré est divisé en périodes temporelles de durée T_s secondes. Et sur chaque tranche du signal, noté x , nous avons N_s échantillons. Le calcul d'énergie sur une période temporelle du signal filtré par le passe-bas suit la règle suivante :

$$\text{Si } \sum_{i=1}^{N_s} x_i^2 > K \text{ alors bit 1 sinon 0}$$

Avec K choisi arbitrairement, ici nous allons choisir de prendre la moyenne de l'énergie sur toutes les périodes temporelles.

En Matlab, nous avons alors :

```
% Calcul de l'energie sur les differentes periodes
somme_energie_bas = sum(reshape(y_bas, Ns, N)'.^2,2);
% Determination de K
K_bas = mean(somme_energie_bas);
% Determination des bits suivant la regle erigee
bits_reconstitues_bas = somme_energie_bas > K_bas;
% Taux d'erreur binaire
Erreur_Filtre_Bas = 1 -
    sum(bits_reconstitues_bas' == Signal_Binaire) / N
```

Nous obtenons bien un taux d'erreur binaire égale à 0 ce qui était l'objectif : retrouver le signal binaire émis. Bien évidemment, si l'on diminue le SNR, c'est-à-dire que le signal devient de plus en plus bruité, le taux d'erreur binaire est affecté directement. En effet, ce dernier augmente et on a donc plus de mal à retrouver le signal binaire initial.

4.4 Modification de l'ordre des filtres

En modifiant l'ordre du filtre à 201, sous Matlab on écrit $Ord = 201$; nous apercevons que l'erreur binaire n'est plus égale à 0. Cela est causé par le retard dû à l'ordre du filtre. En effet, pour un filtre d'ordre n , nous avons un retard de $\frac{n-1}{2}$ échantillons. Pour remédier à ce problème, il suffit d'ajouter autant de zéros que d'échantillons retardés à la fin du signal modulé puis de filtrer ce signal et enfin conserver dans le signal filtré que les échantillons à partir du $\frac{n-1}{2}$ ^{ème}. Pour pouvoir faire cela, nous utilisons ce qui est écrit précédemment en langage Matlab dans le paragraphe 4.1. Après modification, nous retrouvons bien un taux d'erreur binaire égale à 0.

Nous pouvons constater sur la Figure 11 que le filtre sélectionne bien les bonnes fréquences F_0 et F_1 , mais le taux d'erreur binaire n'est pas nul après lancement du programme Matlab. Cela est bien causé par le retard dû à l'ordre des filtres.

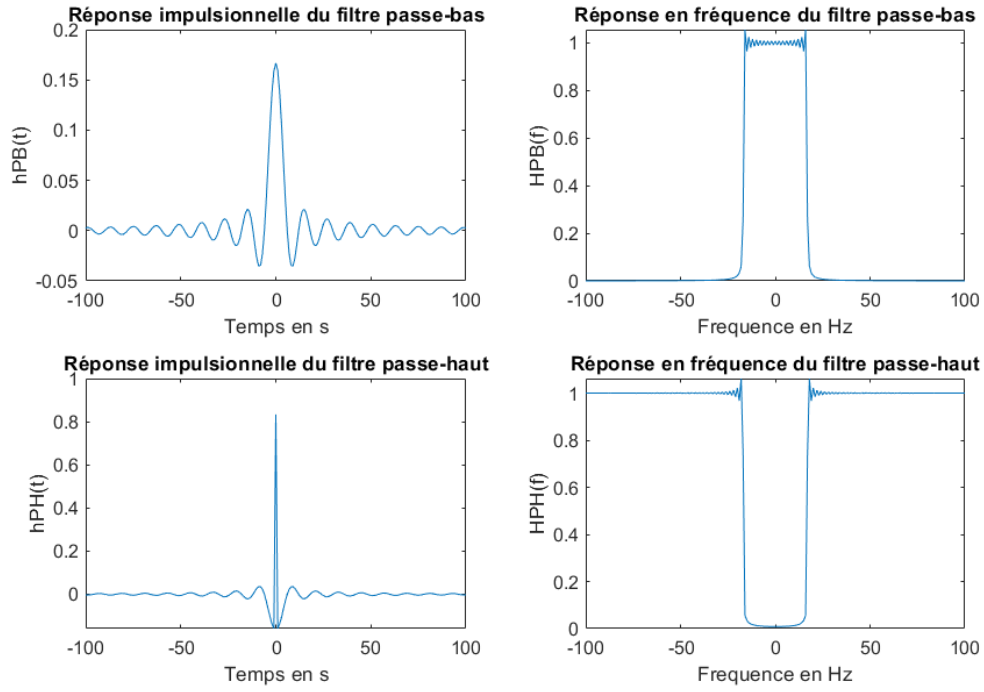


FIGURE 10 – Signal et DSP en sortie du filtre passe-pas et passe-haut d'ordre 201 et de fréquence de coupure F_c

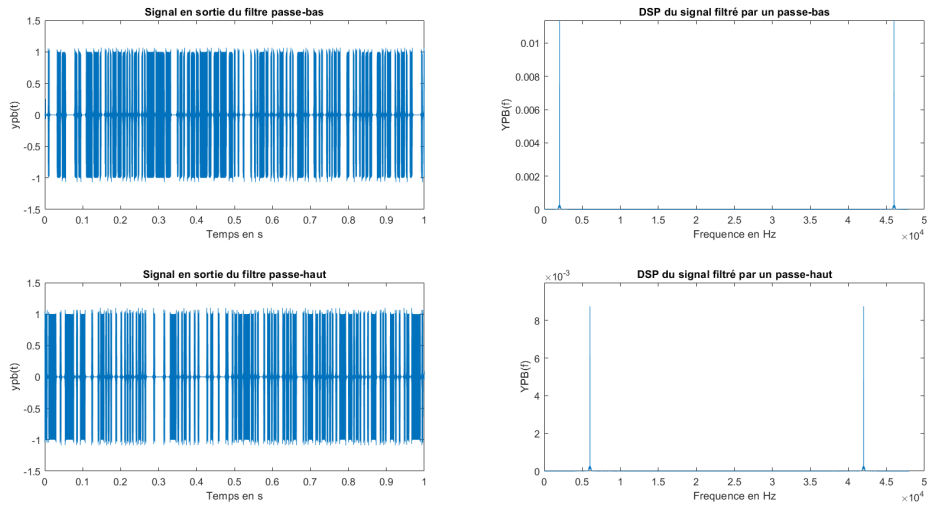


FIGURE 11 – Signal et DSP en sortie du filtre passe-pas et passe-haut d'ordre 201 et de fréquence de coupure F_c

4.5 Utilisation des fréquences de la recommandation V21

Dans la recommandation V21, les fréquences F_0 et F_1 sont modifiées, ils valent désormais :

$$F_0 = 1180 \text{ Hz et } F_1 = 980 \text{ Hz}$$

Nous allons choisir une fréquence de coupure pour le filtre passe-bas égale à $F_c = 1080 \text{ Hz}$ et un débit binaire égale à 300 bits/s. De plus, nous allons nous intéresser à un signal sans bruit (ou en pratique avec un SNR très élevé). On retrouve un taux d'erreur binaire non nul, ce qui montre les limites de la démodulation par filtrage. En effet, pour si l'on code les bits 0 et 1 avec des fréquences très proches, il est difficile voir impossible de pouvoir créer un filtre assez sélectif pour pouvoir sélectionner les 1 ou les 0.

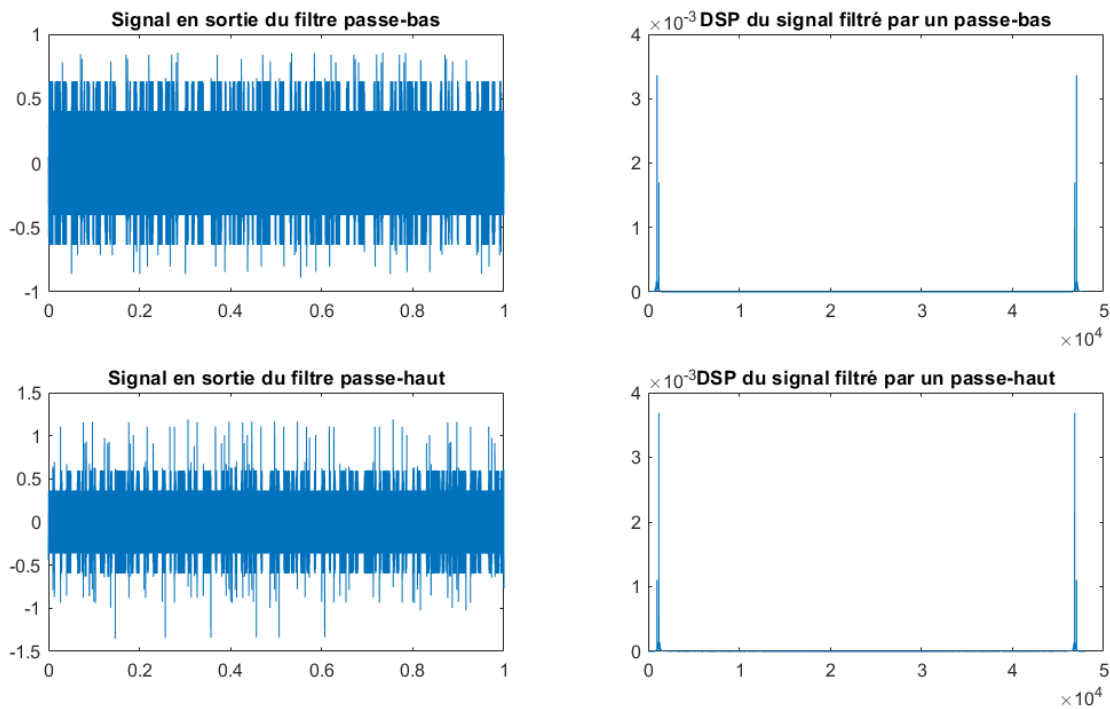


FIGURE 12 – Signal et DSP en sortie du filtre passe-pas et passe-haut d'ordre 61 dans les recommandations V21

Nous pouvons remarquer dans la Figure 12 que le filtre passe-bas et passe haut n'arrivent pas à sélectionner les fréquences de la recommandation V21. En effet, il y a sur la DSP du signal filtré (dans les deux cas que ce soit filtre passe-bas ou passe-haut) deux pics et non un unique pic en F_0 ou en F_1 . Pour remédier à ce problème, nous allons voir dans la suite un nouveau moyen de démoduler un signal.

5 Démodulateur de fréquence adapté à la norme V21

5.1 Contexte de synchronisation idéale

La figure 13 présente le récepteur implanté afin de retrouver, dans un contexte de synchronisation idéale, le message binaire envoyé à partir du signal modulé en fréquence suivant la recommandation V21.

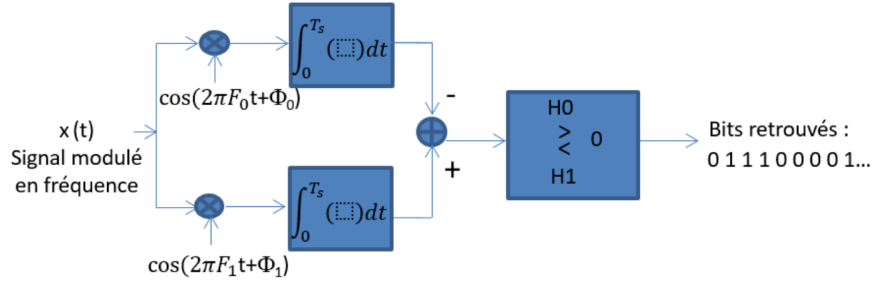


FIGURE 13 – Démodulation FSK. Synchronisation supposée idéale.

5.1.1 Principe de fonctionnement de ce récepteur

Comme le signal modulé est une somme de deux cosinus (de fréquence F_0 et F_1), quand il arrive dans le démodulateur il y a deux cas possibles avec deux sous-cas ce qui fait quatre possibilités :

Rappel : $\psi_0(t) = \cos(2\pi F_0 t + \phi_0)$ et $\psi_1(t) = \cos(2\pi F_0 t + \phi_1)$

A) $\psi_0(t)$ et $\psi_0(t)$ B) $\psi_0(t)$ et $\psi_1(t)$ B') $\psi_1(t)$ et $\psi_0(t)$ C) $\psi_1(t)$ et $\psi_1(t)$

On constate immédiatement que les cas mixtes (B et B') sont identiques ce qui, après avoir ajouté l'intégrale, donne les trois calculs suivants :

$$\begin{aligned}
 A &= \int_0^{T_s} \cos(2\pi F_0 t + \phi_0) \cos(2\pi F_0 t + \phi_0) dt \\
 &= \frac{1}{2} \left(\int_0^{T_s} \cos(4\pi F_0 t + 2\phi_0) dt + \int_0^{T_s} \cos(0) dt \right) \\
 &= \frac{1}{2} \left(\left[\frac{\sin(4\pi F_0 t + 2\phi_0)}{4\pi F_0} \right]_0^{T_s} + T_s \right) \\
 &= \frac{1}{2} \left(\frac{\sin(4\pi F_0 T_s + 2\phi_0)}{4\pi F_0} - \frac{\sin(2\phi_0)}{4\pi F_0} + T_s \right)
 \end{aligned}$$

Comment la fonction sinus est-elle dans $[-1; 1]$ l'ordre de grandeur des deux fractions dans la parenthèse valent 10^{-4} , de plus l'ordre de grandeur de T_s est de 10^{-2} . On peut donc négliger ces deux termes car $10^{-4} \ll 10^{-2}$. Ce qui donne :

$$\begin{aligned}
 A &\approx \frac{1}{2} (0 + 0 + T_s) \\
 &\approx \frac{T_s}{2}
 \end{aligned}$$

$$\begin{aligned}
C &= \int_0^{T_s} \cos(2\pi F_1 t + \phi_1) \cos(2\pi F_1 t + \phi_1) dt \\
&= A \text{ avec } \phi_0 \leftarrow \phi_1 \text{ et } F_0 \leftarrow F_1 \\
&\approx \frac{T_s}{2} \text{ avec } \phi_0 \leftarrow \phi_1 \text{ et } F_0 \leftarrow F_1 \\
&\approx \frac{T_s}{2}
\end{aligned}$$

$$\begin{aligned}
B &= \int_0^{T_s} \cos(2\pi F_0 t + \phi_0) \cos(2\pi F_1 t + \phi_1) dt \\
&= \frac{1}{2} \left(\int_0^{T_s} \cos(2\pi(F_0 + F_1)t + (\phi_0 + \phi_1)) dt + \int_0^{T_s} \cos(2\pi(F_0 - F_1)t + (\phi_0 - \phi_1)) dt \right) \\
&= \frac{1}{2} \left(\left[\frac{\sin(2\pi(F_0 + F_1)t + (\phi_0 + \phi_1))}{2\pi(F_0 + F_1)} \right]_0^{T_s} + \left[\frac{\sin(2\pi(F_0 - F_1)t + (\phi_0 - \phi_1))}{2\pi(F_0 - F_1)} \right]_0^{T_s} \right) \\
&= \frac{1}{2} \left(\frac{\sin(2\pi(F_0 + F_1)t + (\phi_0 + \phi_1))}{2\pi(F_0 + F_1)} - \frac{\sin(\phi_0 + \phi_1)}{2\pi(F_0 + F_1)} + \frac{\sin(2\pi(F_0 - F_1)t + (\phi_0 - \phi_1))}{2\pi(F_0 - F_1)} - \frac{\sin(\phi_0 - \phi_1)}{2\pi(F_0 - F_1)} \right)
\end{aligned}$$

De la même manière que pour A les termes avec $4\pi(F_0 + F_1)$ au dénominateur sont négligeable devant T_s et donc devant A et B.

De plus, l'ordre de grande grandeur de $4\pi(F_0 - F_1)$ est de 10^3 car $F_0 - F_1 = 200$, ce qui fait un ordre de grandeur de 10^{-3} pour les termes avec $4\pi(F_0 - F_1)$ au dénominateur. Ils sont donc également négligeable devant T_s et donc devant A et B

Ce qui donne :

$$\begin{aligned}
B &\approx \frac{1}{2} (0 + 0 + 0 + 0) \quad \text{devant A et C} \\
&\approx 0 \quad \text{devant A et C}
\end{aligned}$$

Pour savoir si le bit qui rentre dans le démodulateur regarde le signe de la différence de deux des trois intégrales possibles (A, B et C), on note cette différence H . Les deux cas possible sont donc les suivants :
Cas H0 : un bits 0 arrive dans le démodulateur :

$$H_0 = B - A \approx 0 - \frac{T_s}{2} = -\frac{T_s}{2} < 0$$

Cas H1 : un bits 1 arrive dans le démodulateur

$$H_1 = C - B \approx \frac{T_s}{2} - 0 = \frac{T_s}{2} > 0$$

De cette manière le démodulateur parvient bien à reconstituer le signal binaire initiale.

5.1.2 Résultats obtenus

Afin d'implémenter cette méthode de démodulation, nous avons besoin de calculer les intégrales :

$$\int_0^{T_s} x(t) \cos(2\pi F_0 t + \phi_0) dt \text{ et } \int_0^{T_s} x(t) \cos(2\pi F_1 t + \phi_1) dt$$

Pour cela, en Matlab, nous écrivons (en utilisant la méthode des rectangles pour calculer numériquement ces intégrales) :

```
% On definit d'abord le terme a integrer pour plus de lisibilite
x_cos_0 = x_bruitee .* cos(2 * pi * F0 * temps + psi0);
x_cos_1 = x_bruitee .* cos(2 * pi * F1 * temps + psi1);

% On somme les echantillons par paquets de Ns echantillons
int_x_0 = sum(reshape(x_cos_0,Ns,N)) * 1/Ns;
int_x_1 = sum(reshape(x_cos_1,Ns,N)) * 1/Ns;
```

Puis on calcul la différence entre ces intégrales et on reconstruit selon la règle établie :

```
% On calcul la difference des integrales
diff_int = int_x_1 - int_x_0;
% On reconstruit le signal binaire
bit_reconstruit_V21 = diff_int > 0;
```

Finalement, on obtient le taux d'erreur binaire avec le calcul simple :

```
% Calcul du taux d'erreur binaire
Erreur_FSK_1 = 1 - sum(bit_reconstruit_V21 == Signal_Binaire) / N
```

Cette formule provient simplement de la formule :

$$\text{Taux d'erreur binaire} = 1 - \frac{\text{Nombre de bits retrouvés}}{\text{Nombre de bits envoyés}}$$

Et nous retrouvons bien un taux d'erreur binaire égale à 0, ce qui était bien l'objectif de ce nouveau démodulateur.

5.2 Gestion d'une erreur de synchronisation de phase porteuse

Le problème de la synchronisation entre l'émetteur et le récepteur est un problème important lorsque l'on réalise une transmission. Les deux doivent être parfaitement synchronisés en temps et en fréquence pour que le démodulateur implanté précédemment fonctionne, ce qui en pratique n'est bien entendu pas possible. Afin que le modem puisse continuer à fonctionner en présence d'une erreur de phase porteuse, celui-ci doit être modifié. La figure 14 présente un démodulateur permettant de s'affranchir de problèmes de synchronisation de phase entre les oscillateurs d'émission et de réception.

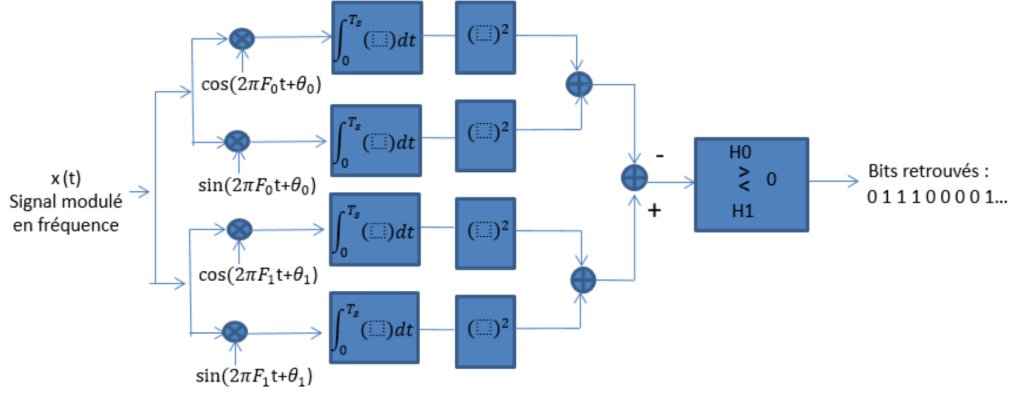


FIGURE 14 – Démodulation FSK - Gestion d'une erreur de phase porteuse.

5.2.1 Impact d'une erreur de phase porteuse sur le modem implanté précédemment

Si on ajoute une erreur de phase (ψ_0 change pour θ_0) le calcul de A (et donc de C) est impacté :

$$\begin{aligned}
 A &= \int_0^{T_s} \cos(2\pi F_0 t + \phi_0) \cos(2\pi F_0 t + \theta_0) dt \\
 &= \frac{1}{2} \left(\int_0^{T_s} \cos(4\pi F_0 t + (\phi_0 + \theta_0)) dt + \int_0^{T_s} \cos(\phi_0 - \theta_0) dt \right) \\
 &= \frac{1}{2} \left(\frac{\sin(4\pi F_0 t + (\phi_0 + \theta_0))}{4\pi F_0} - \frac{\sin(\phi_0 + \theta_0)}{4\pi F_0} + T_s \cos(\phi_0 - \theta_0) \right) \\
 &\approx \frac{T_s}{2} \cos(\phi_0 - \theta_0) \\
 &\approx \frac{T_s}{2} k \text{ avec } k \in [-1; 1]
 \end{aligned}$$

Comme le démodulateur fonctionne uniquement si A est égale à $\frac{T_s}{2}$. Avec l'erreur de phase, A peut être nul ou négatif car il dépend de k qui est aléatoire. Ce démodulateur ne fonctionne donc pas s'il y a une erreur de phase.

5.2.2 Principe de fonctionnement de cette nouvelle version du modem de fréquence

Dans cette nouvelle version du démodulateur, il y a deux cas possibles avec quatre sous-cas ce qui fait huit possibilités :

Rappel : $\psi_0(t) = \cos(2\pi F_0 t + \phi_0)$ et $\psi_1(t) = \cos(2\pi F_0 t + \phi_1)$

A) $\psi_0(t) \cos(2\pi F_0 t + \theta_0)$ B) $\psi_0(t) \sin(2\pi F_0 t + \theta_0)$ C) $\psi_0(t) \cos(2\pi F_1 t + \theta_1)$ D) $\psi_0(t) \sin(2\pi F_1 t + \theta_1)$
 E) $\psi_1(t) \cos(2\pi F_0 t + \theta_0)$ F) $\psi_1(t) \sin(2\pi F_0 t + \theta_0)$ G) $\psi_1(t) \cos(2\pi F_1 t + \theta_1)$ H) $\psi_1(t) \sin(2\pi F_1 t + \theta_1)$

En réutilisant les calculs précédents et ceux fait au fur et à mesure on trouve A, B, C, D, E, F, G et H qui sont les intégrales de 0 à T_s des termes précédents :

$$A \approx \frac{T_s}{2} \cos(\phi_0 - \theta_0)$$

$$\begin{aligned} B &= \int_0^{T_s} \cos(2\pi F_0 t + \phi_0) \sin(2\pi F_0 t + \theta_0) dt \\ &= \frac{1}{2} \left(\int_0^{T_s} \sin(4\pi F_0 t + (\phi_0 + \theta_0)) dt - \int_0^{T_s} \sin(\phi_0 - \theta_0) dt \right) \\ &= \frac{1}{2} \left(-\frac{\cos(4\pi F_0 t + (\phi_0 + \theta_0))}{4\pi F_0} + \frac{\cos(\phi_0 + \theta_0)}{4\pi F_0} - T_s \sin(\phi_0 - \theta_0) \right) \\ &\approx \frac{T_s}{2} \sin(\phi_0 - \theta_0) \end{aligned}$$

$$\begin{aligned} C &= \int_0^{T_s} \cos(2\pi F_0 t + \phi_0) \cos(2\pi F_1 t + \theta_1) dt \\ &= \frac{1}{2} \left(\frac{\sin(2\pi(F_0 + F_1)t + (\phi_0 + \theta_1))}{2\pi(F_0 + F_1)} - \frac{\sin(\phi_0 + \theta_1)}{2\pi(F_0 + F_1)} + \frac{\sin(2\pi(F_0 - F_1)t + (\phi_0 - \theta_1))}{2\pi(F_0 - F_1)} - \frac{\sin(\phi_0 - \theta_1)}{2\pi(F_0 - F_1)} \right) \\ &\approx 0 \end{aligned}$$

$$\begin{aligned} D &= \int_0^{T_s} \cos(2\pi F_0 t + \phi_0) \sin(2\pi F_1 t + \theta_1) dt \\ &= \frac{1}{2} \left(-\frac{\cos(2\pi(F_0 + F_1)t + (\phi_0 + \theta_1))}{2\pi(F_0 + F_1)} + \frac{\cos(\phi_0 + \theta_1)}{2\pi(F_0 + F_1)} - \frac{\cos(2\pi(F_0 - F_1)t + (\phi_0 - \theta_1))}{2\pi(F_0 - F_1)} + \frac{\cos(\phi_0 - \theta_1)}{2\pi(F_0 - F_1)} \right) \\ &\approx 0 \end{aligned}$$

On remarque que tous les termes qui ont des fréquences différentes (C, D, E et F) ont une intégrale négligeable, car il ne font pas apparaître de termes en T_s .

$$\begin{aligned} E &= \int_0^{T_s} \cos(2\pi F_1 t + \phi_1) \cos(2\pi F_0 t + \theta_0) dt \\ &\approx 0 \end{aligned}$$

$$\begin{aligned} F &= \int_0^{T_s} \cos(2\pi F_1 t + \phi_1) \sin(2\pi F_0 t + \theta_0) dt \\ &\approx 0 \end{aligned}$$

$$\begin{aligned}
G &= \int_0^{T_s} \cos(2\pi F_1 t + \phi_1) \cos(2\pi F_1 t + \theta_1) dt \\
&= \frac{1}{2} \left(\frac{\sin(4\pi F_1 t + (\phi_1 + \theta_1))}{4\pi F_1} - \frac{\sin(\phi_1 + \theta_1)}{4\pi F_1} + T_s \cos(\phi_0 - \theta_1) \right) \\
&\approx \frac{T_s}{2} \cos(\phi_1 - \theta_1) \\
H &= \int_0^{T_s} \cos(2\pi F_1 t + \phi_1) \sin(2\pi F_1 t + \theta_1) dt \\
&= \frac{1}{2} \left(-\frac{\cos(4\pi F_1 t + (\phi_1 + \theta_1))}{4\pi F_1} + \frac{\cos(\phi_1 + \theta_1)}{4\pi F_1} + T_s \sin(\phi_0 - \theta_1) \right) \\
&\approx \frac{T_s}{2} \sin(\phi_1 - \theta_1)
\end{aligned}$$

Le fonctionnement de ce démodulateur est similaire au précédent mais avec une modification sur le calcul de la différence H . Les deux cas possible sont donc les suivants :

Cas H0 : un bits 0 arrive dans le démodulateur :

$$\begin{aligned}
H_0 &= (C^2 + D^2) - (A^2 + B^2) \\
&\approx (0 + 0) - \left(\frac{T_s^2}{4} \cos^2(\phi_0 - \theta_0) + \frac{T_s^2}{4} \sin^2(\phi_0 - \theta_0) \right) \\
&\approx -\frac{T_s^2}{4} \left(\cos^2(\phi_0 - \theta_0) + \sin^2(\phi_0 - \theta_0) \right)
\end{aligned}$$

$$H_0 \approx -\frac{T_s^2}{4} < 0$$

Remarque : La différence de phase et éliminer grâce à la mise au carrées, cela illustre concrètement comment le nouveau modulateur règle le problème de déphasage!

Cas H1 : un bits 1 arrive dans le démodulateur :

$$\begin{aligned}
H_1 &= (G^2 + H^2) - (E^2 + F^2) \\
&\approx \left(\frac{T_s^2}{4} \cos^2(\phi_1 - \theta_1) + \frac{T_s^2}{4} \sin^2(\phi_1 - \theta_1) \right) - (0 + 0) \\
&\approx \frac{T_s^2}{4} \left(\cos^2(\phi_1 - \theta_1) + \sin^2(\phi_1 - \theta_1) \right)
\end{aligned}$$

$$H_1 \approx \frac{T_s^2}{4} > 0$$

De cette manière le démodulateur parvient bien à reconstituer le signal binaire initiale même si il y à une erreur de phase porteuse.

5.2.3 Résultats obtenus

Nous allons procéder de la même manière que dans le cas où il n'y avait pas de problème de synchronisation de phase porteuse, c'est à dire que nous allons calculer les intégrales suivantes :

$$\int_0^{T_s} x(t) \cos(2\pi F_0 t + \theta_0) dt \text{ et } \int_0^{T_s} x(t) \sin(2\pi F_0 t + \theta_0) dt$$
$$\int_0^{T_s} x(t) \cos(2\pi F_1 t + \theta_1) dt \text{ et } \int_0^{T_s} x(t) \sin(2\pi F_1 t + \theta_1) dt$$

Pour cela, en Matlab, nous écrivons (en utilisant la méthode des rectangles pour calculer numériquement ces intégrales) :

```
theta_0 = rand*2*pi; % Desynchronisation de phase avec la porteuse
% On definit d'abord les termes a sommer pour plus de lisibilite
x_cos_0 = x_bruitee .* cos(2 * pi * F0 * temps + theta_0);
x_sin_0 = x_bruitee .* sin(2 * pi * F0 * temps + theta_0);
% Calcul de la somme des integrales du haut
sum_x_0 = 1/Ns*sum(reshape(x_cos_0,Ns,N)).^2
          + 1/Ns*sum(reshape(x_sin_0,Ns,N)).^2;

theta_1 = rand*2*pi; % Desynchronisation de phase avec la porteuse
% On definit d'abord les termes a sommer pour plus de lisibilite
x_cos_1 = x_bruitee .* cos(2 * pi * F1 * temps + theta_1);
x_sin_1 = x_bruitee .* sin(2 * pi * F1 * temps + theta_1);
% Calcul de la somme des integrales du bas
sum_x_1 = 1/Ns * sum(reshape(x_cos_1,Ns,N)).^2
          + 1/Ns * sum(reshape(x_sin_1,Ns,N)).^2;
```

Puis on calcul la différence entre ces intégrales et on reconstruit selon la règle établie :

```
% On calcul la difference des sommes d'integrales
diff_sum = sum_x_1' - sum_x_0';
% On reconstruit le signal binaire
bit_reconstruit_V21 = diff_int > 0;
```

Finalement, on calcule le taux d'erreur binaire :

```
% Taux d'erreur binaire
Erreur_FSK_2 = 1 - sum(bit_reconstruit_V21' == Signal_Binaire,2)/N
```

À l'aide ce nouveau modulateur, nous avons pu surpasser le problème de synchronisation de phase porteuse. En effet, nous avons pu obtenir un taux d'erreur binaire nul dans les recommandations V21.

5.3 Reconstitution d'une image

L'un des objectifs de ce projet était de démoduler des signaux contenant une image. Il y en avait 6 et il fallait retrouver une photo prise dans l'ENSEEIH. Pour démoduler les images, nous allons utiliser le dernier démodulateur avec problème de synchronisation de phase porteuse. Par exemple pour la première image, nous écrivons en Matlab :

```
load fichier1.mat                % On importe l'image 1

N = length(signal);              % On recupere la taille du signal de l'image
% On redefinit l'intervalle de temps par rapport a la taille de l'image
temps_image = 0:Te:(N-1)*Te;
% On definit d'abord les termes a sommer pour plus de lisibilite
x_cos_0 = signal .* cos(2 * pi * F0 * temps_image + theta_0);
x_sin_0 = signal .* sin(2 * pi * F0 * temps_image + theta_0);
sum_x_cos_0 = 1/Ns * sum( reshape(x_cos_0,Ns,[])).^2;
sum_x_sin_0 = 1/Ns * sum( reshape(x_sin_0,Ns,[])).^2;

% Calcul de la somme des integrales hauts
sum_x_0 = sum_x_cos_0 + sum_x_sin_0;

% On definit d'abord les termes a sommer pour plus de lisibilite
x_cos_1 = signal .* cos(2 * pi * F1 * temps_image + theta_1);
x_sin_1 = signal .* sin(2 * pi * F1 * temps_image + theta_1);
sum_x_cos_1 = 1/Ns * sum( reshape(x_cos_1,Ns,[])).^2;
sum_x_sin_1 = 1/Ns * sum( reshape(x_sin_1,Ns,[])).^2;

% Calcul de la somme des integrales hauts
sum_x_1 = sum_x_cos_1 + sum_x_sin_1;

% On calcul la difference des sommes d'integrales
diff_sum = sum_x_1 - sum_x_0;
% On reconstruit le signal binaire
bit_reconstruit_image1 = diff_sum > 0;
% On affiche l'image sur la figure 1
figure(1)
% Fonction pour reconstruire l'image a partir du signal
image1 = reconstitution_image(bit_reconstruit_image1);
image(image1)
```

En réitérant sur les 5 autres images et en plaçant les images à la bonne place on trouve l'image suivante :

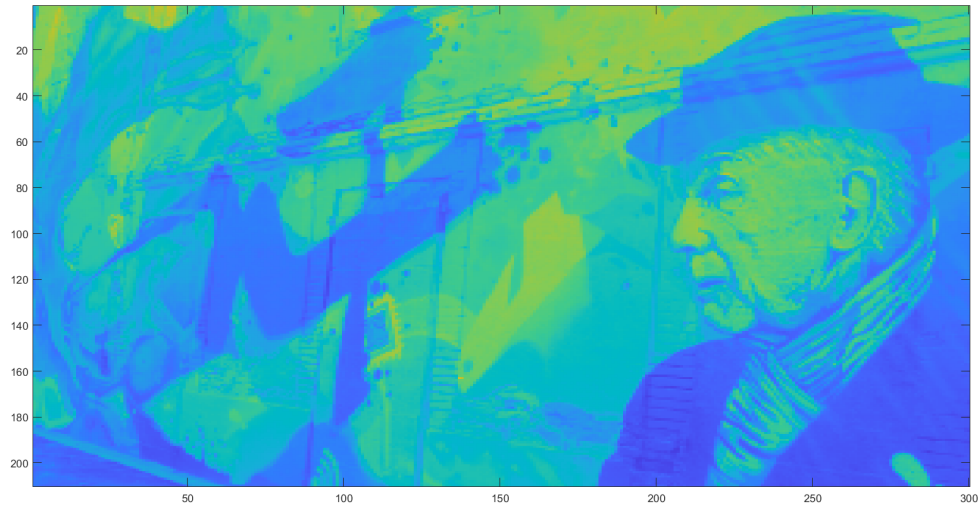


FIGURE 15 – Image recollée à partir de 6 images démodulées

Il s'agit d'une fresque située dans le parc à vélo de l'ENSEEIH ! On y voit le très célèbre Léopold Escande.

5.4 Présentation de Mr. Escande

Léopold Escande, né le 1^{er} juin 1902 à Toulouse et mort le 1^{er} décembre 1980 dans cette même ville, est un physicien français, membre de l'Académie des sciences.

5.4.1 Biographie

Léopold Charles Marie Jean-Baptiste Escande naît le 1^{er} juin 19021 au domicile familial, n° 25 rue d'Alsace-Lorraine à Toulouse. Il est élève du lycée de garçons de la ville (actuel lycée Pierre-de-Fermat), puis entre à l'Institut d'électrotechnique et de mécanique appliquée créé en 1907 par Charles Camichel, et dont il sort diplômé en 1922. Il devient alors le collaborateur de Camichel pour ses recherches en hydraulique. Il soutient sa thèse en 1929, devant un jury présidé par le prix Nobel Paul Sabatier. Il devient maître de conférences de mécanique des fluides en 1930, professeur en 1937. En 1941, il succède à Camichel à la direction de l'Institut d'électrotechnique et de mécanique appliquée qui devient en 1947 l'École nationale supérieure d'électrotechnique et d'hydraulique. Il est élu en 1953 membre correspondant de la section de Mécanique de l'Académie des sciences, puis membre non résident en 1954 (il est alors le plus jeune membre de l'Académie). En 1955, Henri Longchambon le nomme secrétaire général du Conseil supérieur de la recherche qui vient d'être créé. Cette instance devient en 1958 Délégation générale à la recherche scientifique et technique, dont il est nommé Conseiller scientifique.

Il a participé à l'étude de plus de deux cents installations hydrauliques, notamment celles des barrages de Castillon, Génissiat, Assouan, Paulo Afonso (en) et du Lac Vlasina. Léopold Escande a publié 17 ouvrages et 502 articles (dont 284 notes à l'Académie des sciences). Il meurt le 13 septembre 1980 à Toulouse et est inhumé au cimetière de Portet-sur-Garonne.

5.4.2 Postérité

Son nom a été donné à un boulevard de Toulouse et au prix de l'Institut national polytechnique de Toulouse qui récompense des travaux de thèses.

5.4.3 Publications

- Barrages, I, Calcul des barrages poids à profil triangulaire, II, Pratique du calcul des barrages poids à profil triangulaire, 1937
- Recherches sur l'écoulement de l'eau entre les piles de ponts, 1939
- Étude des veines de courant, 1940
- Hydraulique générale, 3 vol., 1941-1943
- Recherches théoriques et expérimentales sur les oscillations de l'eau dans les chambres d'équilibre, 1943
- Dispositif simple de récupération partielle de l'énergie cinétique à l'extrémité aval d'une conduite, 1944
- Étude des coups de bélier dans le cas d'une usine dont le canal de fuite est en charge, 1945
- Étude théorique et expérimentale de la perte de charge de l'eau à la traversée d'une grille, 1946
- Compléments d'hydraulique, 2 vol., 1947, 1951
- Méthodes nouvelles par le calcul des chambres d'équilibre, 1949
- Nouveaux compléments d'hydraulique, 1953

6 Conclusion

Ce projet de traitement numérique du signal a vraiment été très enrichissant sous plusieurs aspects : l'implémentation de signaux numérique, l'utilisation du Matlab et le travail de groupe. De plus, le projet nous a beaucoup appris sur les méthodes de démodulation, avec de nombreuses fonctionnalités Matlab que nous avons découvert.