HETER - TP3 : Effet des non-idéalités sur la réception d'un signal QPSK

Michel Vasilevski, Nicolas Beilleau et Hassan Aboushady

Avant propos

Ce TP fait l'objet d'un compte-rendu qui devra être chargé sous la forme d'un seul fichier au **format pdf** et au plus tard le **mardi 11 octobre 2011** à l'adresse www-asim.lip6.fr/∼hassan.
Le compte-rendu doit comprendre des explications pour chacune des réponses. Les figures doîvent être mises dans un contexte, expliquées, commentées et avoir une légende.
chemin des fichiers : /users/enseig/trncomun/heter/2010-2011/TP3/
Matlab sous linux : /users/soft/matlab/jan03.v6.5r13/bin/matlab -nodesktop
Matlab sous linux : /users/soft/matlab/july04.v7.0_r14/bin/matlab -nodesktop
Matlab sous linux : /users/soft/matlab/mars09.R2009a/bin/matlab -nodesktop
Compilation du code SystemC AMS et execution : make puis ./run.x
Visualisation des courbes avec matlab ; ouvrir matlab executer le script ./trace/Trace.m
Les seuls fichiers a modifier contiennent un commentaire commençant par "//TODO".
Vous pouvez néanmoins lire les autres fichiers pour assimiler les concepts vus dans le TP2.

On désire maintenant implémenter certaines des non-idéalités qui perturbent la réception du signal et qui peuvent provoquer des erreurs de détection. Elles peuvent être extèrieures aux circuits de réception et d'émission ou bien provenir de l'implémentation de ceux-ci.

1 Non-linéarité et la figure de bruit du LNA

Nous nous plaçons dans le dossier "1_lna".
Les fichiers a modifier sont : "include/lna.h".
Voici (Figure 1) un appercu de l'effet de la non-linearité du LNA.

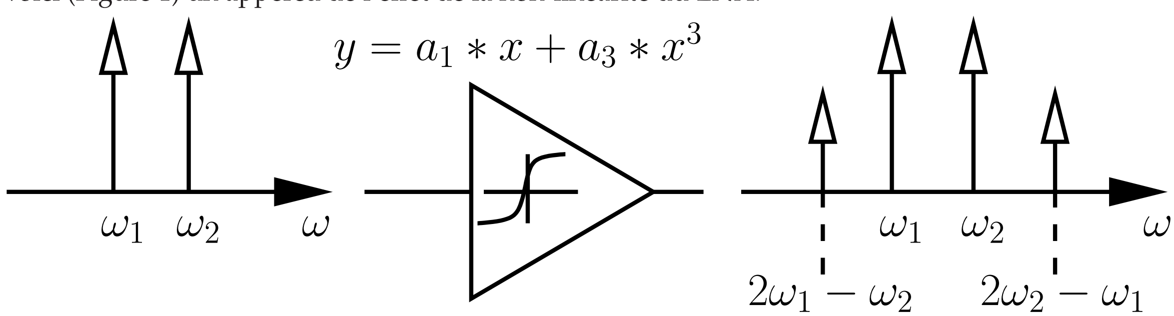

Figure 1: Effet de la non-linéarité du LNA sur deux signaux adjacents.

Question 1 : Faire les modifications nécessaires pour introduire la non-linéarité du LNA à partir du "IIP3".

L'ensemble du bruit ajouté par un composant RF est exprimé sous forme de "Noise Figure" (NF). Il met en relation la modification apportée sur le SNR.
Question 2 : Faire les modifications nécessaires pour introduire le bruit du LNA à partir du "Noise

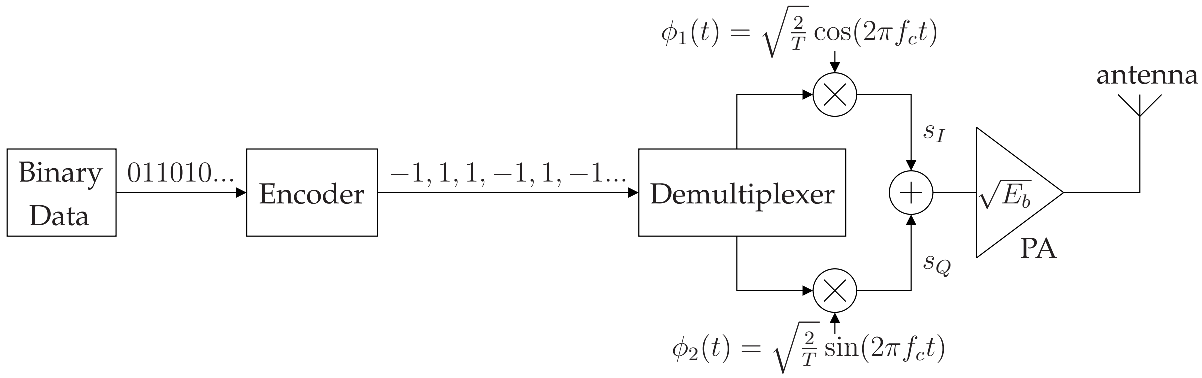

Figure 2: Emetteur d'un signal numérique modulé en QPSK.

Figure".

2 Canal de transmission

Nous nous plaçons dans le dossier "2_channel".
Les fichiers a modifier sont : "include/channel_AWGN.h" et "main.cpp".

Le problème principal de la transmission radio est le passage par l'air. Le signal n'étant plus guidé il part dans différentes directions et son énergie diminue suivant la distance parcourue et suivant les obstacles qu'il rencontre avant d'atteindre le récepteur. Un autre problème est la présence de signaux extérieurs perturbateurs.
Un modèle simple du canal de transmission est un bruit blanc gaussien additionné au signal transmis (**AWGN** : Additive White Gaussian Noise). On dispose de la fonction **randn()** qui renvoit un nombre suivant une loi gaussienne (ou normale). La fonction de densité de probabilité d'une loi gaussienne est :

$$f(x) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{1}$$

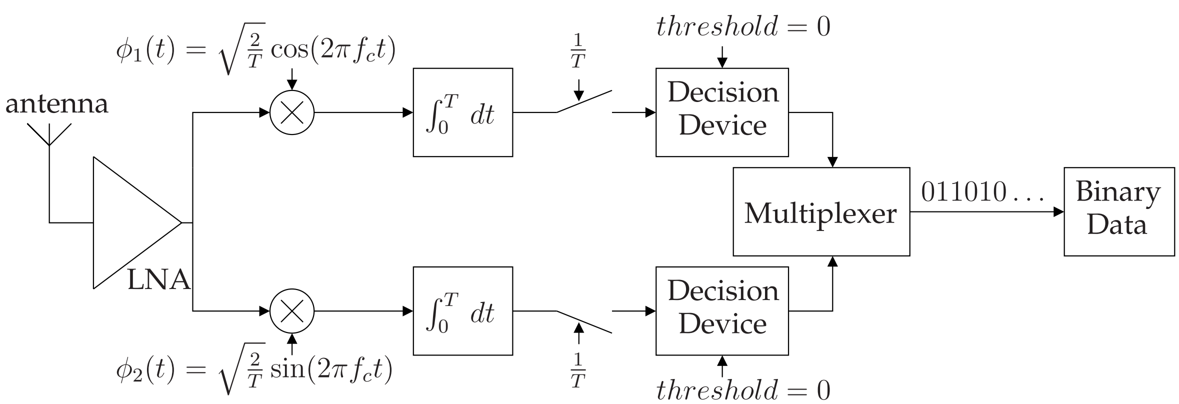

Figure 3: Récepteur idéal d'un signal QPSK cohérent.

La fonction de densité de probabilité de la fonction randn est définie par :

$$f_r(x_r) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x_r^2}{2}} \tag{2}$$

On a $\mu = 0$ et $\sigma = 1$. On peut passer d'une distribution définie par l'équation (1) à une distribution définie par l'équation (2) en posant :

$$x = \mu + \sigma\, x_r \ \text{ avec } \ x_r = randn \tag{3}$$

Question 3 : En considérant un bruit blanc comme défini par la figure 4 déterminez une expression de l'écart type σ en fonction de N_0 et f_s.
(**NB :** La puissance d'un signal aléatoire est σ^2 et on la calcule également en intégrant la densité spectrale de puissance sur sa bande).

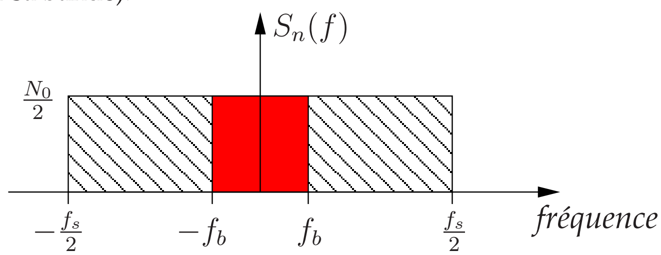

Figure 4: Densité Spectrale de Puissance du bruit blanc.

Question 4 : Déduire une valeur de σ pour un N_0 = -8dB.
Question 5 : Implémenter le bruit de canal avec le σ trouvé dans le récepteur SystemC AMS et affichez la constellation.
Question 6 : Comparez les signaux obtenus avec un canal bruité à ceux obtenus avec un canal idéal. Expliquer les différences et les problèmes engendrés par le bruit au niveau de la détection.

Pour valider le récepteur, c'est à dire pour savoir si le bruit ajouté nuit ou pas au fonctionnement de notre récepteur, on doit calculer le **B**it **E**rror **R**ate (BER). Il exprime, comme son nom l'indique, le nombre de bits erronés par rapport au nombre de bits total. On peut le calculer de plusieurs façons et la plus simple mais la moins rapide est de simuler le récepteur et de comparer les bits émis aux bits reçus.
Néanmoins dans le cas d'un signal modulé en QPSK corrompu par un AWGN on peut trouver une expression mathématique pour le BER (cf. annexe A et cours).
Question 7 : Modifier le "main.cpp" pour effectuer une boucle sur la simulation en faisant varier l'energie du "Power Amplifier" pour des valeur de -8dB à 0dB. Par ce moyen , nous faisons varier Eb/N0 à travers Eb. **Question 8 :** Tracez la courbe de BER à partir de simulations et comparez avec la courbe théorique.

3 Non-idéalités des composants

Nous nous plaçons dans le dossier "3_txrx".
Les fichiers a modifier sont : "include/receiver/lo.h" et "main.cpp".

3.1 Bruit thermique

Le bruit thermique (dû à l'agitation thermique des électrons) est présent dans tous les transistors et toutes les résistances d'un circuit. On peut le modéliser sous certaines approximations par un bruit blanc gaussien. On a donc la même courbe de BER que celle obtenue pour le bruit du canal.

3.2 Mauvais appariement des gains des chemins I et Q ("Quadrature Gain Mismatch")

Les signaux qui arrivent à l'antenne du récepteur sont extrêment faibles. Les différents filtrages peuvent également diminuer l'amplitude du signal, on doit donc utiliser des gains pour amplifier le signal. Le problème est l'appariement des gains des 2 chemins I et Q.
Question 9 : Implémentez le "Quadrature Gain Mismatch" dans votre récepteur avec une difference de 0,1 entre les 2 chemins. Affichez la constellation.
Question 10 : Faites varier la différence entre les gains et affichez la constellation. Commentez.

3.3 Mauvais appariement des phases des chemins I et Q ("Quadrature Phase Mismatch")

Les signaux ϕ_1 et ϕ_2 utilisés par les mélangeurs doivent être en quadrature mais les imperfections des circuits qui génèrent ces signaux peuvent causer une quadrature imparfaite. On traduit ce problème par un mauvais appariement entre les phases des 2 signaux.
Question 11 : Implémentez le "Quadrature Phase Mismatch" dans votre récepteur avec une déviation de 10 degrés par rapport à la quadrature parfaite. Affichez la constellation.
Question 12 : Faites varier la déviation par rapport à la quadrature parfaite et affichez la constellation. Commentez.

3.4 Décalage de fréquence ("Frequency offset")

Pour générer les signaux $\phi_i(t)$ on utilise des circuits qui ont besoin de références en fréquence (oscillateurs). Ces références ont besoin d'un certain temps pour stabiliser leur signal. On traduit ce phénomène par une déviation Δf de la fréquence. L'ordre de grandeur de Δf dépend du circuit utilisé pour faire la référence et de la référence à atteindre mais l'ordre de grandeur est donné par la relation suivante :

$$\left[\frac{1}{60};\frac{10}{6}\right] = \frac{\Delta f}{f_c}$$

Question 13 : Implémentez le "Frequency offset" dans votre récepteur avec un décalage de 1.10^{-3}. Affichez la constellation.
Question 14 : Faites varier le décalage de fréquence et affichez la constellation. Commentez.

A Calcul du BER pour un bruit blanc gaussien dans un récepteur QPSK

Nous allons considérer dans un premier temps un seul chemin (I ou Q).
La probabilité d'avoir un 0 détecté à la sortie du récepteur lorsqu'un 0 a été émis dépend de la puissance du bruit qui a été ajouté au signal. Considèrons la fonction de densité du bruit blanc gaussien :

$$f(x) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{4}$$

On peut définir, en posant $\sigma = \sqrt{\frac{N_0}{2}}$ et $\mu = -\sqrt{E_b}$, la fonction de densité de probabilité pour un 0 ($-\sqrt{E_b}$) transmis :

$$f(x|0) = \frac{1}{\sqrt{\pi\,N_0}} e^{-\frac{(x+\sqrt{E_b})^2}{N_0^2}} \tag{5}$$

Dans notre cas l'erreur est la détection d'un 1 à la place du 0. Donc pour connaître la probabilité d'erreur on détermine la probabilité que x soit positif :

$$P_{e_0} = \int_0^\infty f(x|0)\,dx = \frac{1}{\sqrt{\pi\,N_0}} \int_0^\infty e^{-\frac{(x+\sqrt{E_b})^2}{N_0^2}}\,dx \tag{6}$$

En posant :

$$z = \frac{x+\sqrt{E_b}}{\sqrt{N_0}}$$

On fait un changement de variable pour avoir à partir de l'équation (7) la forme suivante :

$$\begin{aligned} P_{e_0} &= \frac{1}{\sqrt{\pi}} \int_{\sqrt{\frac{E_b}{N_0}}}^\infty e^{-z^2} dz \\ &= \frac{1}{2} erfc(\frac{\sqrt{E_b}}{N_0}) \end{aligned} \tag{7}$$

La fonction erfc(x) est la fonction d'erreur complémentaire **qui existe dans Matlab**.
On a donc défini la probabilité de l'erreur "1 à la place de 0" sur un des deux chemins du récepteur.
La probabilité de l'erreur inverse est la même dûe à la symétrie des régions de décision par rapport à l'origine.
On en déduit donc l'expression du BER ou la probabilité de se tromper sur un bit, c'est à dire sur un des 2 chemins :

$$BER = \frac{1}{2} erfc(\frac{\sqrt{E_b}}{N_0}) \tag{8}$$