Travaux pratiques concernant les codes cycliques - Première partie

23 avril 2010

1 Génération de code cycliques

Soit le code cyclique $(7,4)$ dont le polynôme générateur est

$$g(X) = X^3 + X + 1.$$

1. Prouver d'abord théoriquement que le code est cyclique en vérifiant que les deux conditions données dans le cours sont bien remplies.
2. Ecrire une routine qui génère tous les mots-code à l'aide du polynôme générateur. Afficher dans un tableau la correspondance entre mots d'information et mots-code. Pour faire les multiplications on utilisera la routine Matlab $gfconv$. Parexemple, si l'on veut multiplier $I(X) = 1 + X^2 + X^3$ par $X^3 + X + 1$, on effectuera,

   $$C = gfconv([1,0,1,1],[1,1,0,1])$$

   qui donne $C = [1,1,1,1,1,1,1]$, soit $C(X) = 1 + X + X^2 + X^3 + X^4 + X^5 + X^6$. On prendra garde que les poids forts dans Matlab sont à droite.
3. Vérifiez "manuellement" sur quelques codes que les mots-code appartiennent bien à un code cyclique.
4. Ecrire maintenant une routine qui rende le code systématique. Afficher dans un nouveau tableau la correspondance entre mots d'information et mots-code. Pour faire les divisions on utilisera pour simplifier la routine

gfdeconv de Matlab. Par exemple, si l'on veut diviser $(1 + X^5 + X^6)$ par $(X^3 + X + 1)$, on effectuera

$$Q = gfdeconv([1, 0, 0, 0, 0, 1, 1], [1, 1, 0, 1])$$

qui donne $Q = [0, 1, 1, 1]$ soir $Q(X) = X + X^2 + X^3$. On prendra garde que les poids forts dans Matlab sont à droite. Lorsque l'on voudra calculer les restes, on utilisera pour les multiplications la routine Matlab *gfconv* et pour les additions la routine *gfadd*.

5. Compléter votre routine en ajoutant un calcul de la distance minimale entre les mots-code, la capacité de détection et la capacité de correction.
6. Générer les mêmes mots-code à l'aide d'une matrice génératrice systématique.

2 Equivalence entre le domaine temporel et le domaine polynomial

Lorsque l'on reçoit des mots-code de n bits, on considère que le bit reçu au temps 0 est le bit de poids fort ($i.e$ associé à la puissance X^{n-1}) et le bit reçu au temps $n-1$ est le bit de poids faible ($i.e$ associé à la puissance 0). On rappelle que chaque fois que l'on considère un mot-code de n bits sous la forme

$$[c_{n-1},\ c_{n-2},\ c_{n-3}, \ldots c_2,\ c_1,\ c_0]$$

on peut tout aussi bien l'associer à un polynôme de la forme

$$c_{n-1}X^{n-1} + c_{n-2}X^{n-2} + c_{n-3}X^{n-3} + \ldots + c_2X^2 + c_1X + c_0$$

1. Pour simplifier, on suppose que c_{n-1} et c_0 son nuls. Montrer que si des bits du mot-code sont retardés d'un coup d'horloge cela revient à multiplier le mot-code polynômial initial par X^{-1}. Montrer que si l'on était capable d'anticiper d'un coup d'horloge, cela reviendrait à multiplier le mot-code polynômial initial par X.
2. En déduire que la bascule D qui symbolise un retard temporel d'un coup d'horloge (T secondes entre deux bits consécutifs), peut être représentée par un circuit polynômial qui est une simple multiplication par X^{-1}.

3 Circuit de division élémentaire, théorie

Lorsque l'on veut implémenter un code systématique, on est obligé de faire des divisions polynômiales. Ceci peut sembler très compliqué. En fait, les calculs se font par des circuits très simples. C'est le but de cette section que de le montrer.

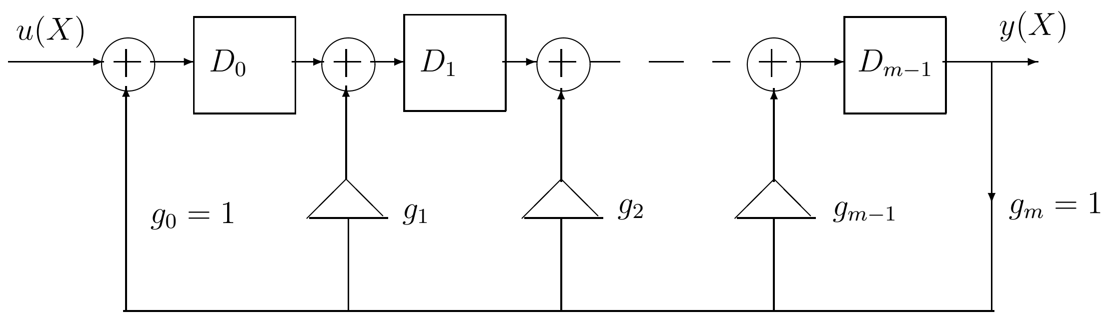

FIGURE 1 – Circuit à m bascules rebouclées.

1. Soit le circuit de la Figure 1 dans lequel les g_i pour $i = 1 \ldots m-1$ sont des 0 ou des 1 et les triangles représentent des multiplicateurs. En utilisant les réponses aux questions précédentes, montrer que le cicuit de la Figure 1 a un équivalent polynômial sous forme du circuit de la Figure 2.
2. Montrer théoriquement que le circuit de la Figure 2 divise $U(X)$ par

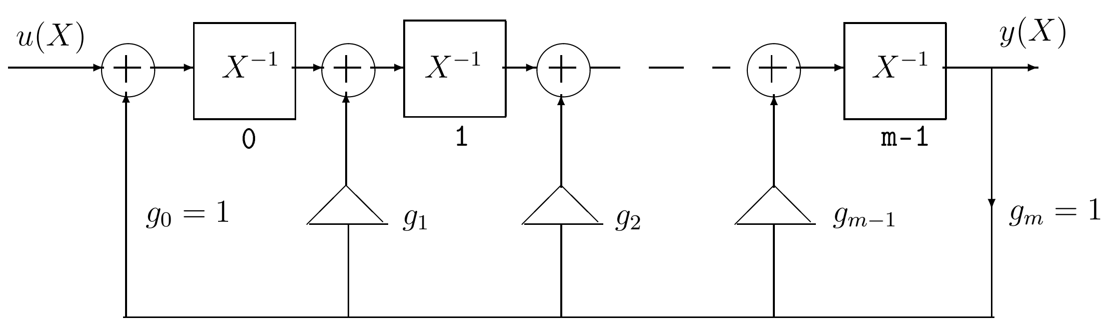

FIGURE 2 – Equivalent "polynômial" du circuit précédent.

$g(X) = 1 + g_1X + g_2X^2 + \dots g_{m-1}X^{m-1} + X^m$, c'est-à-dire que

$$y(X) = \frac{u(X)}{1 + g_1X + g_2X^2 + \ldots + g_{m-1}X^{(m-1)} + X^m}$$

4 Circuit de division élémentaire et encodage systématique, pratique

Considérer un polynôme générateur d'un code de Golay (23, 12) de la forme

$$g(X) = X^{11} + X^{10} + X^6 + X^5 + X^4 + X^2 + 1$$

1. Dessiner le circuit d'encodage systématique dans le domaine temporel.
2. Choisir au hasard un $u(X)$ de degré 22 et programmer une routine qui simule le circuit temporel (sans utiliser de division polynômiale). On veillera à écrire proprement les équations d'état du système de façon à ce que la programmation soit la plus simple possible.
3. Vérifier sur quelques cas que vos calculs sont bien corrects en utilisant les restes par divisions polynômiales (utiliser les routines $gfdeconv$, $gfconv$, $gfadd$).

5 Circuit de décodage systématique, pratique

1. Imaginer un principe de calcul de syndrome.
2. Sur combien de bits le syndrome se calcule-t-il ?
3. Dessiner le circuit de décodage en vous basant sur le circuit d'encodage.
4. En reprenant le code de Golay, faire un décodage d'une trame de longeur 23 bits. Tester le décodage avec des trames qui sont des mots-codes et des trames qui sont entachées d'erreur (simuler 1 erreur par exemple).
5. Vérifier sur quelques cas que les calculs produits par votre circuit de décodage sont bien corrects en utilisant les restes par divisions polynômiales (utiliser les routines $gfdeconv$, $gfconv$, $gfadd$).