Fabien DONIUS, Nicolas GRILL, Chérine KAMEL, Selim MILED - Ing1 Gr4

ANALYSE MATHEMATIQUE
GOLAY (24,12,8)

Projet CCE

On ne corrige pas les correcteurs

Les codes correcteurs d'erreur

ECOLE CENTRALE D'ELECTRONIQUE

GROUPE ECE

I. **Génération des matrices :**

Le code de Golay, comme le code de Hamming est un code matriciel généré à partir d'un polynôme générateur.

Ce code a néanmoins quelques particularités par rapport aux codes de Hamming sur lesquels il est basé. Tout d'abord, il s'agit d'un code correcteur dit cyclique, c'est-à-dire que toute permutation circulaire d'un mot de code reste un mot de code (un mot de code correspond à des données encodées par la matrice génératrice). Les codes cycliques ont aussi une particularité concernant leur polynôme générateur : ils divisent le polynôme $z^n + 1$ (ou $z^n - 1$).

Le polynôme générateur du code correcteur binaire étendu de Golay (23,12,7) est :

$$H(z) = z^{11} + z^{10} + z^6 + z^5 + z^4 + z^2 + 1$$

En appliquant les divisions euclidiennes (pour connaître dans les détails ce type de génération, voir le principe de fonctionnement du code de Hamming).

Pour obtenir le code de Golay (24,12,8) à partir du code de Golay (23,12,7), il suffit de rajouter un bit de parité à la fin de la matrice génératrice obtenue. Pour obtenir ce bit de parité, il suffit de faire la somme de tous les bits de la ligne considérée. Si cette somme est égale à 0[2], alors le bit de parité ajouté sera 1. Si cette somme est égale à 1[2], ce bit sera égal à 0.

Nous obtenons ainsi la matrice génératrice suivante :

$$G := \begin{pmatrix}
1&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&1&1&1&1&1&1\\
0&1&0&0&0&0&0&0&0&0&0&0&1&1&1&0&1&1&1&0&0&0&1\\
0&0&1&0&0&0&0&0&0&0&0&0&1&1&0&1&1&1&0&0&0&1&0\\
0&0&0&1&0&0&0&0&0&0&0&0&1&0&1&1&1&0&0&0&1&0&1\\
0&0&0&0&1&0&0&0&0&0&0&0&1&1&1&1&0&0&0&1&0&1&1\\
0&0&0&0&0&1&0&0&0&0&0&0&1&1&1&0&0&0&1&0&1&1&0\\
0&0&0&0&0&0&1&0&0&0&0&0&1&1&0&0&0&1&0&1&1&0&1\\
0&0&0&0&0&0&0&1&0&0&0&0&1&0&0&0&1&0&1&1&0&1&1\\
0&0&0&0&0&0&0&0&1&0&0&0&1&0&0&1&0&1&1&0&1&1&1\\
0&0&0&0&0&0&0&0&0&1&0&0&1&0&1&0&1&1&0&1&1&1&0\\
0&0&0&0&0&0&0&0&0&0&1&0&1&1&0&1&1&0&1&1&1&0&0\\
0&0&0&0&0&0&0&0&0&0&0&1&1&0&1&1&0&1&1&1&0&0&0
\end{pmatrix}$$

Bien sûr, comme pour le code de Hamming, nous devons transposer cette matrice afin de pouvoir l'exploiter. Nous obtenons ainsi :

$$G^T = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \end{pmatrix}$$

La matrice de parité est :

$$H := \begin{pmatrix} 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

II. Correction des erreurs :

Comme pour le code de Hamming, le vecteur représentant les données à transmettre est encodé avec la matrice génératrice selon cette équation :

Soit D le vecteur de données à transmettre et D_e les données encodées. Nous avons alors la relation :

$$D_e = G^T \cdot D$$

On pose les données à transmettre suivantes :

$$D := \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \end{pmatrix}$$

Les données encodées sont données par le produit $G^T \cdot D$

Nous obtenons alors :

$$De := \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1 \end{pmatrix}$$

III. Capacité de correction d'erreur du code de Golay (24,12,8)

Le troisième paramètre du code nous indique la distance de Hamming maximale entre deux mots de code, à savoir 8 bits. La capacité de correction du code est donc : $\frac{8-1}{2} = 3{,}5$, or 3,5 n'étant pas un nombre entier, et le nombre d'erreur étant toujours entier, le code de Golay(24,12,8) peut donc corriger trois erreurs, consécutives ou non.

IV. Cas idéal : aucune erreur lors de la transmission :

Bien que le code de Golay soit basé sur le code de Hamming, la manière de détecter les erreurs diffère. Nous pouvons utiliser deux méthodes pour détecter la présence d'erreurs.

Nous faisons le produit $H \cdot De$ et nous devons trouver le vecteur nul, ce qui signifie qu'il n'y a eu aucune erreur lors de la transmission. Seulement cette méthode n'est valable que pour une seule erreur. Le code de Golay étant capable de corriger plus d'erreurs, cette méthode ne présente aucun avantage.

Une autre méthode consiste à séparer le message en deux à l'arrivée.

Nous voyons que le vecteur de données De est clairement séparé en deux parties. Les premiers 12 bits correspondent aux données à transmettre. Les 12 derniers bits aux données de parité introduites par la matrice génératrice.

Soit p le vecteur de données de parité.

Dans le cas précédent, ce vecteur est :

$$p := \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1 \end{pmatrix}$$

En faisant le produit $G^T \cdot P$, nous obtenons le vecteur suivant :

$$z := \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \end{pmatrix}$$

En inversant toutes les coordonnées, nous pouvons remarquer que les 12 derniers bits de ce vecteur correspondent aux 12 premiers bits des données effectivement reçues. La transmission s'est donc effectuée sans erreur.

V. **Cas non idéal : présence d'erreurs**

Nous devons d'abord savoir combien d'erreur sont apparues lors de la transmission.

Nous supposons que 2 erreurs sont apparues lors de la transmission (au bit 4 et au bit 9) :

Nous obtenons les données suivantes :

$$Dr := \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1 \end{pmatrix}$$

Nous coupons ce vecteur en deux parties : l'une contenant les 12 bits de données, l'autre contenant les 12 bits de parité. Sachant que les erreurs sont apparues uniquement dans la partie contenant les données, nous pouvons faire le produit de la matrice génératrice et des données de parité pour trouver les erreurs et les corriger. Si des erreurs sont apparues uniquement dans les données de parité, nous effectuons le produit de la matrice génératrice et des 12 premiers bits représentant les données pour trouver les erreurs dans les données de parité.

Maintenant, nous allons supposer qu'une erreur est apparue dans chaque partie du message, c'est-à-dire une dans la partie contenant les données, une autre dans les données de parité.

Nous prenons la partie contenant les données du message reçu, puis nous l'encodons avec la matrice génératrice. Étant donné que des erreurs sont apparues, nous obtiendrons quelque chose de très différent du message reçu.

La méthode de correction consiste à intervertir les bits de la partie contenant les données un par un et à encoder le résultat avec la matrice génératrice. Quand nous trouverons une différence inférieure à deux bits entre les données reçue et les données précédemment encodées, nous avons trouvé l'erreur, et nous l'avons corrigée. Il ne reste plus qu'à décoder.