Test Ecrit N° 1 en Communications Numériques 2 (CN2)

Deuxième année Master Télécom
Jeudi 10 décembre 2015 à 10H00
Salle H3

Durée de l'épreuve: 45 mns
Calculatrices scientifiques autorisées

SUJET A

Exercice 1 : Le CRC - Code Cyclique de Redondance (6 pts)

Soit le polynôme générateur $G(x)=x^4+x^2+x$.

a. On souhaite transmettre le message suivant :1111011101.
a.1. Donner l'expression de m(x).
a.2. Quel sera le CRC à ajouter ? (utiliser la division binaire)
b. Le message 1111000101010 est reçu. Est-il correct ?
c. Quel est l'utilité de ce code détecteur d'erreur ?

Exercice 2 : Code linéaire en bloc et correction (6 pts)

Soit le code linéaire systématique défini par la matrice de contrôle :

$$P=\begin{pmatrix}1&0&1\\0&1&1\\1&1&0\\1&1&1\end{pmatrix}$$

a. Déterminer *n* et *k*.
b. Donner la désignation complète de ce code.
c. Coder le message suivant : 10011101.
d. Donner la matrice de contrôle *H*.
e. Vérifier si les messages suivants sont corrects : 1111100, 0111000.
f. Donner le nombre de combinaisons pour les erreurs : a) simples, b) doubles.
g. Construire la table d'erreur/syndrome pour les erreurs simples et le cas sans erreurs.
h. Corriger le message 0100101 par la méthode de syndrome.

Exercice 3: Code par répétition (*n*,2) (3 pts)

On considère un code correcteur d'erreur $C(n,k)$ pour lequel $k = 2$ et n est un entier pair tel que $n \geq 6$, et dont les mots de codes « v » sont obtenus à partir des mots d'informations u = (u1, u2) en les répétant ($n/2$- 1) fois. En d'autres termes, le mot de code obtenu à partir de u = (u1, u2) où (u1, u2) appartient à $\{0,1\}^2$ s'écrit :

$$v = (u1, u2, u1, u2, \ldots, u1, u2) \qquad (*)$$

Par exemple, si $n = 8$, le mot de code obtenu à partir de (1, 0) est (1, 0, 1, 0, 1, 0, 1, 0).

a. Donnez une matrice génératrice *G* de ce code $C(n,2)$ (où, pour rappel, n est un entier pair supérieur ou égal à 6).
b. Donnez une matrice de contrôle *H* de ce code.
c. Pour n=6, coder le message 110110.
d. Quel est le nombre maximal q de bits erronés que ce code garantit de pouvoir toujours détecter ?

Rappels : Matrice génératrice et matrice de contrôle d'un code linéaire systématique *C*(*n*, *k*)

k : nombre de bits du message sans codage (longueur du message après codage de source)
n : nombre de bits du message après codage de canal (longueur du mot de code)
$r=n-k$: nombre de bits de contrôle (ou de redondance)

$$G=(I_k \mid P),\ [I_k]=k\times k,\ [P]=k\times r;\ r=n-k$$

$$H=G^T=(P^T \mid I_r),\ [P^T]=r\times k, [I_r]=r\times r;\ r=n-k$$

$$H^T=\begin{pmatrix} P \\ -- \\ I_r \end{pmatrix},\ [P]=k\times r,\ \ [I_r]=r\times r;\ \ r=n-k$$

$$s=R\ H^T$$

QUE DIEU VOUS AIDE

Test Ecrit N° 1 en Communications Numériques 2 (CN2)

Deuxième année Master Télécom
Jeudi 10 décembre 2015 à 10H00
Salle H3

Durée de l'épreuve: 45 mns
Calculatrices scientifiques autorisées

SUJET B

Exercice 1 : Le CRC - Code Cyclique de Redondance (6 pts)

Soit le polynôme générateur $G(x)=x^4+x^2+x$.

a. On souhaite transmettre le message suivant : 1100010101.
a.1. Donner l'expression de m(x).
a.2. Quel sera le CRC à ajouter ? (utiliser la division binaire)
b. Le message 1111000101010 est reçu. Est-il correct ?
c. Quel est l'utilité de ce code détecteur d'erreur ?

Exercice 2 : Code linéaire en bloc et correction (6 pts)

Soit le code linéaire systématique défini par la matrice de contrôle :

$$P=\begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \end{pmatrix}$$

a. Déterminer *n* et *k*.
b. Donner la désignation complète de ce code.
c. Coder le message suivant : 10011101.
d. Donner la matrice de contrôle *H*.
e. Vérifier si les messages suivants sont corrects : 1111100, 0111000.
f. Donner le nombre de combinaisons pour les erreurs : a) simples, b) doubles.
g. Construire la table d'erreur/syndrome pour les erreurs simples et le cas sans erreurs.
h. Corriger le message 0100101 par la méthode de syndrome.

Exercice 3: Code par répétition (n,2) (3 pts)

On considère un code correcteur d'erreur $C(n,k)$ pour lequel $k = 2$ et n est un entier pair tel que $n \geq 6$, et dont les mots de codes « v » sont obtenus à partir des mots d'informations u = (u1, u2) en les répétant (n/2- 1) fois. En d'autres termes, le mot de code obtenu à partir de u = (u1, u2) où (u1, u2) appartient à $\{0,1\}^2$ s'écrit :

$$v = (u1, u2, u1, u2, \ldots, u1, u2) \qquad (*)$$

Par exemple, si $n = 8$, le mot de code obtenu à partir de (1, 0) est (1, 0, 1, 0, 1, 0, 1, 0).

a. Donnez une matrice génératrice G de ce code $C(n,2)$ (où, pour rappel, n est un entier pair supérieur ou égal à 6).
b. Donnez une matrice de contrôle H de ce code.
c. Pour n=6, coder le message 110110.
d. Quel est le nombre maximal q de bits erronés que ce code garantit de pouvoir toujours détecter ?

<u>Rappels : Matrice génératrice et matrice de contrôle d'un code linéaire systématique *C*(*n*, *k*)</u>

k : nombre de bits du message sans codage (longueur du message après codage de source)
n : nombre de bits du message après codage de canal (longueur du mot de code)
$r=n-k$: nombre de bits de contrôle (ou de redondance)

$$G=(\, I_k \mid P), \; [I_k]=k\times k, \; [P]=k\times r; \; r=n-k$$

$$H=G^T=(P^T \mid I_r), \; [P^T]=r\times k, [I_r]=r\times r; \; r=n-k$$

$$H^T = \begin{pmatrix} P \\ -- \\ I_r \end{pmatrix}, \; [P]=k\times r, \;\; [I_r]=r\times r; \;\; r=n-k$$

$$s=R\,H^T$$

<u>QUE DIEU VOUS AIDE</u>

Corrigé type de test écrit n°1 / sujet A

Exercice 1 : Le CRC - Code Cyclique de Redondance (Cyclic Redundancy Code) (6 pts)

Soit le polynôme générateur $G(x)=x^4+x^2+x$.

a. On souhaite transmettre le message suivant :1111011101.

a.1. Donner l'expression de M(x).

La représentation générale sous forme polynomiale des suites de bits à transmettre $M=m_1m_2….m_n$ s'écrit par le polynôme :

$$M(x)=m_n+m_{n-1}x+…+m_1x^{n-1}$$

La suite 1111011101 est donc représentée par le polynôme (n=10) :

$$M(x)=1.x^9+1.x^8+1.x^7+1.x^6+0.x^5+1.x^4+1.x^3+1.x^2+0.x^1+1.x^0=x^9+ x^8+ x^7+ x^6+ x^3+ x^2+ 1$$ **(0.5 pt)**

a.2. Quel sera le CRC à ajouter ? (Utiliser la division binaire)

Le code cyclique de redondance (CRC) est un code linéaire systématique détecteur d'erreurs. Le calcul de bits de redondance constituant le CRC est décrit ci-après.

Emission d'un CRC : On ajoute à droite du message à émettre « M » un code de contrôle tel que le polynôme correspondant au message plus le code de contrôle soit divisible par le polynôme générateur choisi.

Le calcul du CRC à la transmission inclut les étapes suivantes:

1) Choix de polynôme générateur, noté G(x), d'ordre *m*.
2) Transformation de G(x) en un mot binaire comme nous l'avons fait avec le message « M ».
3) Ajout de *m* zéros au message binaire à transmettre. Le mot résultant « M_e » contient alors ($n+m$) bits.
4) À cet étape, si nous choisissons la méthode de la division euclidienne, on transforme M_e en forme polynomiale puis on effectue une division euclidienne de $M_e(x)$ sur G(x) dans laquelle on ne tient pas compte du quotient. Le reste de division représente le CRC à ajouter au mot avant transmission.

Sinon, la division binaire est plus simple et donc plus recommandée : Elle consiste à ajouter itérativement au mot binaire M_e en base 2 ('ou exclusif *xor*'), le mot correspondant au polynôme générateur de telle sorte que si le bit de poids fort du reste intermédiaire est '0' on décale de reste intermédiaire. Cette opération de décalage-sommation est poursuivie jusqu'à ce que le mot obtenu soit inférieur au polynôme générateur. Ce mot correspond au CRC tandis que le quotient est sans intérêt.

Dans notre cas :

$G(x)= x^4+x^2+x$ ce qui correspond au mot binaire de polynôme générateur : 10110.

M=1111011101 et donc M_e=1111011101**0000** (ajout de 4 zéros correspondant à l'ordre de polynôme générateur m=4).

La division binaire est effectuée comme suit :

1 1 1 1 0 1 1 1 0 1 0 0 0 0

1 0 1 1 0

0 1 0 0 0 1 1 1 0 1 0 0 0 0

1 0 1 1 0

0 0 1 1 1 1 1 0 1 0 0 0 0

1 0 1 1 0

0 1 0 0 1 0 1 0 0 0 0

1 0 1 1 0

0 0 1 0 0 1 0 0 0 0

1 0 1 1 0

0 0 1 0 0 0 0 0

1 0 1 1 0

0 0 **1 1 0 0**

(1.5 pts)

Le CRC est par conséquent **1100 (0.5 pt)**. Le mot de code à transmettre sur le canal est 1111011101**1100 (0.5 pt)**.

b. Le message 1111000101010 est reçu. Est-il correct ?

Réception d'un CRC et détection d'erreurs: Le mot reçu contient le message de la source plus le CRC de *m* bits. Il doit être divisible par le polynôme générateur. La vérification de transmission sans erreurs se fait par division euclidienne en base 2 ou par division binaire de mot reçu et du polynôme générateur qui doit résulter en un reste qui est nul.

1 1 1 1 0 0 0 1 0 1 0 1 0 1

1 0 1 1 0

0 1 0 0 0 0 0 1 0 1 0 1 0 1

1 0 1 1 0

0 0 1 1 0 0 1 0 1 0 1 0 1

1 0 1 1 0

0 1 1 1 1 0 1 0 1 0 1

1 0 1 1 0

0 1 0 0 0 1 0 1 0 1

1 0 1 1 0

0 0 1 1 1 0 1 0 1

1 0 1 1 0

0 1 0 1 1 0 1

1 0 1 1 0

0 0 0 0 0 1

(1.5 pts)

Le reste de la division n'est pas nul ; il existe alors des erreurs dans le mot reçu **(0.5 pt)**.

c. Quel est l'utilité de ce code détecteur d'erreur ?

Ce code est très utilisé car il est connu par sa performance très élevée de détection d'erreurs dû essentiellement à sa méthode particulière de calcul de bits de redondance en dépendance du polynôme générateur. Nous le savons d'ailleurs de la théorie de l'information que le codage est plus performant lorsqu'il est plus complexe. On retrouve ce code très souvent dans les réseaux locaux ainsi que les réseaux de télécommunications.

Par exemple, le CRC-16 ($G(x)=x^{16}+x^{15}+x^{2}+1$) et CRC-CCITT ($G(x)=x^{16}+x^{12}+x^{5}+1$) utilisés en téléphonie sans fils détectent des erreurs simples et doubles, des erreurs impaires, plusieurs bits successifs erronés, dits salves, de 16 bits ou moins, 99.997% des salves de 17 bits et jusqu'à 99.998% des salves de 18 bits ou plus.

La détection performante des erreurs sans pouvoir les détecter permet de réinterroger l'émetteur afin de retransmettre de nouveau le message jusqu'à être reçu sans erreurs **(1 pt)**.

Exercice 2 : Code linéaire en bloc et correction (7.5 pts)

Soit le code linéaire systématique défini par la matrice de contrôle :

$$P=\begin{pmatrix}1&0&1\\0&1&1\\1&1&0\\1&1&1\end{pmatrix}$$

a. Déterminer n et k.

La matrice génératrice d'un code linéaire en bloc (version systématique) a la forme suivante :

$$G=(\,I_k\mid P),\ [I_k]=k\times k,\ [P]=k\times r;\ r=n\text{-}k$$

où pour rappel :
k : nombre de bits du message sans codage (longueur du message après codage de source)
n : nombre de bits du message après codage de canal (longueur du mot de code)
$r=n-k$: nombre de bits de contrôle (ou de redondance)

Dans notre cas:

$$[P]=k\times r=4\times 3$$

Ce qui donne :

$$k=4,\ r=3 \text{ et donc } n=r+k=7 \qquad \textbf{(0.5 pt)}$$

b. Donner la désignation complète de ce code.

Ce code est un code linéaire systématique de longueur n=7 dont 4 bits de message à coder et 3 bits de contrôle. La forme de P est connue en théorie de codage et elle correspond au code de Hamming dans sa version systématique **(0.5 pt)**. La distance minimale de ce code est donc d=3. La désignation complète de ce code s'écrit alors $C(7,4,3)$ **(0.5 pt)**.

c. Coder le message suivant : 10011101.

Le codage du message est effectué en deux étapes :

1) Découpage du message en paquets de k bits,
2) Multiplication de chaque paquet par la matrice génératrice donnant ainsi le mot de code : $v=mG$, m étant le message de source de longueur $k=4$ dans notre cas.

$$G=(I_k \mid P)=G=(I_4 \mid P)=\begin{pmatrix}1&0&0&0&1&0&1\\0&1&0&0&0&1&1\\0&0&1&0&1&1&0\\0&0&0&1&1&1&1\end{pmatrix} \quad \textbf{(0.25 pt)}$$

$$m_1=[1\ 0\ 0\ 1] \rightarrow v_1=m_1G=[1\ 0\ 0\ 1]\begin{pmatrix}1&0&0&0&1&0&1\\0&1&0&0&0&1&1\\0&0&1&0&1&1&0\\0&0&0&1&1&1&1\end{pmatrix}=[1\ 0\ 0\ 1\ 0\ 1\ 0]$$

$$m_2=[1\ 1\ 0\ 1] \rightarrow v_2=m_2G=[1\ 1\ 0\ 1]\begin{pmatrix}1&0&0&0&1&0&1\\0&1&0&0&0&1&1\\0&0&1&0&1&1&0\\0&0&0&1&1&1&1\end{pmatrix}=[1\ 1\ 0\ 1\ 0\ 0\ 1]$$

Le mot à transmettre sur le canal est donc 10010101101001. **(0.25 pt)**

d. Donner la matrice de contrôle H.

La matrice de contrôle d'un code linéaire en bloc (version systématique) a la forme suivante :

$$H=G^T=(P^T \mid I_r),\ [P^T]=r\times k, [I_r]=r\times r;\ r=n-k$$

ce qui donne:

$$H=\begin{pmatrix}1&0&1&1&1&0&0\\0&1&1&1&0&1&0\\1&1&0&1&0&0&1\end{pmatrix} \quad \textbf{(0.25 pt)}$$

e. Vérifier si les messages suivants sont corrects : 1111100, 0111000.

Un mot de code R est reçu correctement si :

$$R\,H^T=0$$

avec

$$H^T=\begin{pmatrix}P\\ \text{----}\\ I_r\end{pmatrix},\ [I_r]=r\times r;\ r=n-k$$

d'où

$$H^T=\begin{pmatrix}1&0&1\\0&1&1\\1&1&0\\1&1&1\\1&0&0\\0&1&0\\0&0&1\end{pmatrix} \quad \textbf{(0.25 pt)}$$

Pour R_1=1111100 on trouve

$$R_1\, H^T=011$$ **(0.25 pt)**

Il existe des erreurs dans R_1 **(0.25 pt)**.

Pour R_2=0111000 on trouve

$$R_2\, H^T=010$$ **(0.25 pt)**

Il existe des erreurs dans R_2 **(0.25 pt)**.

f. Donner le nombre de combinaisons pour les erreurs : a) simples, b) doubles.

Nous savons d'après la théorie de codage que chaque ligne de H^T correspond à une erreur simple. Donc le nombre des erreurs simples est n **(0.25 pt)**.

Le syndrome s défini par :

$$s=R\, H^T, \quad R= r_0 r_1 \ldots r_{n-1}$$

est de taille r. Le nombre de combinaisons possibles du syndrome est donc 2^r dont un état de zéros partout relatif à une transmission sans erreurs.

Le nombre de combinaisons doubles est donc :

$$2^r-n-1$$ **(0.25 pt)**

Dans notre cas, le nombre des erreurs simples est n=7. Le nombre des erreurs doubles est donc

$$2^3-7-1=0$$ **(0.25 pt)**

Ce qui est attendu car le code de Hamming est de distance d=3 et donc ne peut corriger qu'une seule erreur :

$$t=\left\lfloor \frac{d-1}{2} \right\rfloor = \lfloor 1 \rfloor = 1$$ **(0.25 pt)**

où $\lfloor . \rfloor$ est le symbole de la partie entière.

g. Construire la table d'erreur/syndrome pour les erreurs simples et le cas sans erreurs.

La table d'erreur/syndrome est constituée en se basant sur les règles suivantes :

1) Chaque ligne de H^T correspond à une erreur simple,

2) La i ème ligne correspond à l'erreur en bit d'indice i-1. Ceci correspond à une erreur en bit r_{i-1} donc à une valeur de 1 dans le vecteur d'erreur e à la position i-1 où $e=e_0 e_1 \ldots e_{n-1}$ relié au syndrome par la relation :

$$s = RH^T = (v+e)H^T = vH^T+eH^T = eH^{\mathrm{T}}$$

puisque vH^T=0 du fait que v est un mot de code valide.

(2 pts)

Syndrome	**Erreur**	**Observation**
000	0000000	Pas d'erreurs
001	0000001	Erreur simple en septième bit r_6 (ceci correspond à la 7ème ligne de H^T)
010	0000010	Erreur simple en sixième bit r_5 (ceci correspond à la 6ème ligne de H^T)
011	0100000	Erreur simple en deuxième bit r_1 (ceci correspond à la 2ème ligne de H^T)
100	0000100	Erreur simple en quatrième bit r_4 (ceci correspond à la 5ème ligne de H^T)
101	1000000	Erreur simple en premier bit r_0 (ceci correspond à la 1ère ligne de H^T)
110	0010000	Erreur simple en troisième bit r_2 (ceci correspond à la 3ème ligne de H^T)
111	0001000	Erreur simple en quatrième bit r_3 (ceci correspond à la 4ème ligne de H^T)

h. Corriger le message 0100101 par la méthode de syndrome.
Le syndrome correspondant à ce mot reçu est :

$$s = [0\ 1\ 0\ 0\ 1\ 0\ 1]\ H^T = [1\ 1\ 0]$$

La correction de l'erreur s'effectue en ajoutant le vecteur d'erreur correspondant au syndrome 110 au mot reçu. Á partir de la table d'erreur/syndrome ci-dessus on trouve le mot corrigé v^* comme suit:

$$v^* = R+e^* = [0\ 1\ 0\ 0\ 1\ 0\ 1] + [0\ 0\ 1\ 0\ 0\ 0\ 0] = [0\ 1\ 1\ 0\ 1\ 0\ 1] \quad \textbf{(1 pt)}$$

On vérifie que :

$$v^*\ H^T=0$$

Exercice 3: Code par répétition (n,2) (1.5 pts)

On considère un code correcteur d'erreur $C(n,k)$ pour lequel $k = 2$ et n est un entier pair tel que $n \geq 6$, et dont les mots de codes « v » sont obtenus à partir des mots d'informations u = (u1, u2) en les répétant ($n/2$- 1) fois. En d'autres termes, le mot de code obtenu à partir de u = (u1, u2) où (u1, u2) appartient à $\{0,1\}^2$ s'écrit :

$$v = (u1, u2, u1, u2, \ldots, u1, u2) \qquad (*)$$

Par exemple, si $n = 8$, le mot de code obtenu à partir de (1, 0) est (1, 0, 1, 0, 1, 0, 1, 0).

a. Donnez une matrice génératrice G de ce code $C(n,2)$ (où, pour rappel, n est un entier pair supérieur ou égal à 6).

$$G=\begin{pmatrix} 1\ 0 & \underbrace{\begin{matrix} 1\ 0 & 1\ 0 & 1\ 0\ \ldots & 1\ 0 \\ 0\ 1 & 0\ 1 & 0\ 1\ \ldots & 0\ 1 \end{matrix}}_{(n/2\ -1)\text{ fois}} \\ 0\ 1 & \end{pmatrix}$$

(0.5 pt)

La matrice G n'est rien que une répétition de (n/2) fois la matrice d'identité I_2. Ceci produira un mot de code de taille n **(0.25 pt)**.

b. Donnez une matrice de contrôle H de ce code $C(n,2)$.

$$H=G^T=(h_1\ h_2)\ ;\ h_1=(1\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ \ldots\ 1\ 0)^T\ ;\ h_2=(0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ \ldots\ 0\ 1)^T. \quad \textbf{(0.25 pt)}$$

c. Pour n=6, coder le message 110110.

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

(0.25 pt)

Le mot de code à transmettre est : v=111111010101101010 **(0.25 pt)**

d. Quel est le nombre maximal q de bits erronés que ce code garantit de pouvoir toujours détecter ?
Il est clair que la distance de Hamming est égale à n/2 (qui correspond aux messages 01 et 10). Puisqu'il s'agit d'un code linéaire, le nombre maximal q de bits erronés que ce code garantit de pouvoir toujours détecter est égale à d-1=n/2 -1. Le nombre maximal de bits erronés que ce code peut corriger est égale à t=partie réelle((d-1)/2)=partie réelle((n/2-1)/2).

Corrigé type de test écrit n°1 / sujet B

Exercice 1 : Le CRC - Code Cyclique de Redondance (Cyclic Redundancy Code) (6 pts)

Soit le polynôme générateur $G(x)=x^4+x^2+x$.

a. On souhaite transmettre le message suivant : 1100010101.

a.1. Donner l'expression de m(x).

La représentation générale sous forme polynomiale des suites de bits à transmettre $M=m_1m_2\ldots m_n$ s'écrit par le polynôme :

$$M(x)=m_n+m_{n-1}x+\ldots+m_1x^{n-1}$$

La suite 1100010101 est donc représentée par le polynôme (n=10) :

$$M(x)=1.x^9+1.x^8+0.x^7+0.x^6+0.x^5+1.x^4+0.x^3+1.x^2+0.x^1+1.x^0=x^9+x^8+x^4+x^2+1$$ **(0.5 pt)**

a.2. Quel sera le CRC à ajouter ? (Utiliser la division binaire)

Dans notre cas :

$G(x)=x^4+x^2+x$ ce qui correspond au mot binaire de polynôme générateur : 10110.

M=1100010101 et donc M_e=1100010101**0000** (ajout de 4 zéros correspondant à l'ordre de polynôme générateur m=4).

La division binaire est effectuée comme suit :

```
1 1 0 0 0 1 0 1 0 1 0 0 0 0
1 0 1 1 0
----------------------------
0 1 1 1 0 1 0 1 0 1 0 0 0 0
  1 0 1 1 0
----------------------------
  0 1 0 1 1 0 1 0 1 0 0 0 0
    1 0 1 1 0
----------------------------
    0 0 0 0 0 1 0 1 0 0 0 0
              1 0 1 1 0
----------------------------
              0 0 0 1 0 0 0
```

(1.5 pts)

Le CRC est par conséquent **1000 (0.5 pt)**. Le mot de code à transmettre sur le canal est 1100010101**1000 (0.5 pt)**.

d. Le message 11110001010101 est reçu. Est-il correct ?

La vérification de transmission sans erreurs se fait par division euclidienne en base 2 ou par division binaire de mot reçu et du polynôme générateur qui doit résulter en un reste qui est nul.

```
1 1 1 1 0 0 0 1 0 1 0 1 0 1
1 0 1 1 0
_____________________________
0 1 0 0 0 0 0 1 0 1 0 1 0 1
  1 0 1 1 0
_____________________________
  0 0 1 1 0 0 1 0 1 0 1 0 1
      1 0 1 1 0
_____________________________
      0 1 1 1 1 0 1 0 1 0 1
        1 0 1 1 0
_____________________________
        0 1 0 0 0 1 0 1 0 1
          1 0 1 1 0
_____________________________
          0 0 1 1 1 0 1 0 1
              1 0 1 1 0
_____________________________
              0 1 0 1 1 0 1
                1 0 1 1 0
_____________________________
                0 0 0 0 0 1
```

(1.5 pts)

Le reste de la division n'est pas nul ; il existe alors des erreurs dans le mot reçu **(0.5 pt)**.

e. Quel est l'utilité de ce code détecteur d'erreur ?
Même réponse que celle du sujet A.

Exercice 2 : Code linéaire en bloc et correction **(7.5 pts)**

Soit le code linéaire systématique défini par la matrice de contrôle :

$$P=\begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \end{pmatrix}$$

a. Déterminer n et k.
La matrice génératrice d'un code linéaire en bloc (version systématique) a la forme suivante :

$$G=(\,I_k \mid P),\ [I_k]=k\times k,\ [P]=k\times r;\ r=n\text{-}k$$

Dans notre cas:

$$[P]=k\times r=4\times 3$$

Ce qui donne :

$$k=4,\ r=3 \text{ et donc } n=r+k=7$$

b. Donner la désignation complète de ce code.'
Même réponse qu'en sujet A.

c. Coder le message suivant : 10011101.

Le codage du message est effectué en deux étapes :

1) Découpage du message en paquets de k bits,
2) Multiplication de chaque paquet par la matrice génératrice donnant ainsi le mot de code : $v=mG$, m étant le message de source de longueur $k=4$ dans notre cas.

$$G=(I_k \mid P)=G=(I_4 \mid P)=\begin{pmatrix} 1&0&0&0&1&1&1\\ 0&1&0&0&1&0&1\\ 0&0&1&0&0&1&1\\ 0&0&0&1&1&1&0 \end{pmatrix}$$

$$m_1=[1\ 0\ 0\ 1] \rightarrow v_1=m_1G=[1\ 0\ 0\ 1]\begin{pmatrix} 1&0&0&0&1&1&1\\ 0&1&0&0&1&0&1\\ 0&0&1&0&0&1&1\\ 0&0&0&1&1&1&0 \end{pmatrix}=[1\ 0\ 0\ 1\ 0\ 0\ 1]$$

$$m_2=[1\ 1\ 0\ 1] \rightarrow v_2=m_2G=[1\ 1\ 0\ 1]\begin{pmatrix} 1&0&0&0&1&1&1\\ 0&1&0&0&1&0&1\\ 0&0&1&0&0&1&1\\ 0&0&0&1&1&1&0 \end{pmatrix}=[1\ 1\ 0\ 1\ 1\ 0\ 0]$$

Le mot à transmettre sur le canal est donc 10010011101100.

d. Donner la matrice de contrôle H.

La matrice de contrôle d'un code linéaire en bloc (version systématique) a la forme suivante :

$$H=G^T=(P^T \mid I_r),\ [P^T]=r\times k, [I_r]=r\times r;\ r=n-k$$

ce qui donne:

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

e. Vérifier si les messages suivants sont corrects : 1111100, 0111000.

Un mot de code R est reçu correctement si :

$$R\,H^T=0$$

avec

$$H^T=\begin{pmatrix} P\\ \hdashline I_r \end{pmatrix},\ [I_r]=r\times r;\ \ r=n-k$$

d'où

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

Pour R_1=1111100 on trouve

$$R_1\, H^T=011$$

Il existe des erreurs dans R_1.

Pour R_2=0111000 on trouve

$$R_2\, H^T=000$$

Il n'existe pas d'erreurs dans R_2.

f. Donner le nombre de combinaisons pour les erreurs : a) simples, b) doubles.
Même réponse qu'en sujet A.

g. Construire la table d'erreur/syndrome pour les erreurs simples et le cas sans erreurs.
Le tableau de syndrome est constitué en se basant sur les règles suivantes :
1) Chaque ligne de H^T correspond à une erreur simple,
2) La i ème ligne correspond à l'erreur en bit d'indice i-1. Ceci correspond à une erreur en bit r_{i-1} donc à une valeur de 1 dans le vecteur d'erreur e à la position i-1 où $e=e_0e_1\ldots.e_{n-1}$ relié au syndrome par la relation :

$$s = RH^T = (v+e)H^T = vH^T+eH^T = eH^{\mathrm{T}}$$

puisque vH^T=0 du fait que v est un mot de code valide.

Syndrome	**Erreur**	**Observation**
000	0000000	Pas d'erreurs
001	0000001	Erreur simple en septième bit r_6 (ceci correspond à la 7ème ligne de H^T)
010	0000010	Erreur simple en sixième bit r_5 (ceci correspond à la 6ème ligne de H^T)
011	0010000	Erreur simple en troisième bit r_2 (ceci correspond à la 3ème ligne de H^T)
100	0000100	Erreur simple en cinquième bit r_4 (ceci correspond à la 5ème ligne de H^T)
101	0100000	Erreur simple en deuxième bit r_1 (ceci correspond à la 2ème ligne de H^T)
110	0001000	Erreur simple en premier bit r_3 (ceci correspond à la 4ème ligne de H^T)
111	1000000	Erreur simple en premier bit r_0 (ceci correspond à la 1ère ligne de H^T)

h. Corriger le message 0100101 par la méthode de syndrome.

Le syndrome correspondant à ce mot reçu est :

$$s = \ [0\ 1\ 0\ 0\ 1\ 0\ 1]\ H^T = [0\ 0\ 0]$$

Le syndrome est nul et donc le code ne peut ni détecter ni corriger l'erreur. Il s'agit d'une erreur multiple.