

## Examen de Moyenne Durée

### Correction

**Niveau :** Licence 2, 2018-2019

**Module :** Réseaux de communication, S2.

**Date & heure :** Jeudi, 11-07-2019, 13h30-15h00.

#### Exercice 1 : (5 points)

Soit le réseau : 132.45.0.0/16. Nous cherchons à découper ce réseau en 8 sous réseaux.

1. Combien de bits sont nécessaires pour définir 8 sous réseaux.
2. Donner le masque réseau correspondant.
3. Donner l'adresse réseau de chacun des 8 sous réseaux.
4. Quelle est la plage des adresses utilisables pour le sous réseau 3.
5. Quelle est l'adresse de diffusion du sous réseau 4.

#### Solution exercice 1 : (1 point par question)

1. Nombre de bits : 3 bits car  $8=2^3$
2. Masque :  $255.255.11100000.0 = 255.255.224.0$
3. Adresse réseau de chacun des 8 sous réseaux :

0	132.45.00000000.0=132.45.0.0	1	132.45.00100000.0=132.45.32.0
2	132.45.01000000.0=132.45.64.0	3	132.45.01100000.0=132.45.96.0
4	132.45.10000000.0=132.45.128.0	5	132.45.10100000.0=132.45.160.0
6	132.45.11000000.0=132.45.192.0	7	132.45.11100000.0=132.45.224.0

4. Plage d'adresses pour le sous réseau 3 :

[132.45.01100000.00000001 –

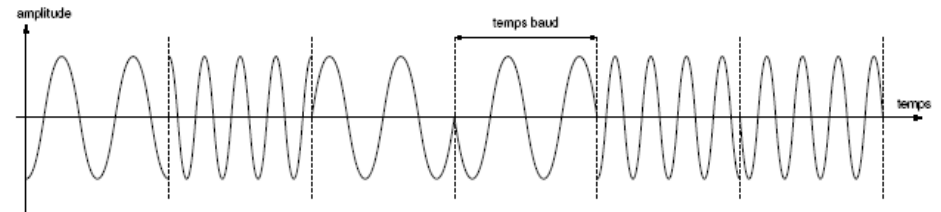
132.45.01111111.11111110]=[132.45.96.1-132.45.127.254]

5. Adresse de diffusion du sous réseau 4 :

132.45.10011111.11111111=132.45.159.255

#### Exercice 2 : (5 points)

Soit le signal suivant :



1. Quel est le type de la modulation utilisée ?
2. Supposons que toutes les combinaisons possibles apparaissent sur le signal précédent, calculer la valence et donner le nombre de bits transmis par baud.
3. Proposer une valeur binaire pour chaque combinaison possible, ensuite déduire l'information transportée par le signal.
4. Sans recourir à d'autre type de modulation, proposer une modulation pour transmettre 5 bits à la fois.
5. Représenter l'information transportée, trouvée dans la question 3, en utilisant le codage NRZ et Manchester.

#### Solution exercice 2 : (1 point par question)

1. Type de modulation : fréquence-phase.
2. Dans ce type de modulation, 2 valeurs de fréquences (fréquence basse  $f_b$ , fréquence haute  $f_h$ ) et 4 valeurs de phases ( $-\pi/2$ ,  $+\pi/2$ , 0 et  $\pi$ ) ont été utilisées, donc le nombre de combinaisons possibles est égal à  $4 \times 2 = 8$ , en conséquence la valence est égale à 8 et le nombre de bits transmis par baud est égal à 3 car  $8=2^3$
3. Valeur binaire proposée pour chaque combinaison

$F_b, -\pi/2$	000	$F_h, -\pi/2$	001
---------------	-----	---------------	-----

Fb, $+\pi/2$	010	Fh, $+\pi/2$	011
Fb, 0	100	Fh, 0	101
Fb, $\pi$	110	Fh, $\pi$	111

Suite de bits transmise par le signal : 000 011 100 110 001 111

Il se peut que d'autres combinaisons sont proposées, au quel cas, la suite ne sera pas la même, donc on les juge sur la démarche plutôt que sur le résultat.

4. Modulation proposée pour transmettre 5 bits à la fois :  
5 bits i.e,  $2^5 = 32$  combinaisons, on propose par exemple : 4 valeurs fréquences et 8 valeurs phases, l'essentiel c'est que le nombre de valeurs fréquence multiplié par le nombre de valeurs phases soit égal à 32.
5. La suite trouvée dans la question 3 : 000 011 100 110 001 111  
NRZ (Non Return to Zero) : le bit 1 est représenté par un état significatif, et le bit 0 par un autre état significatif.  
Manchester : une transition montante au milieu de la période correspond à la valeur 0 et une transition descendante au milieu de la période correspond à la valeur 1.

### Exercice 3 : QCM (1.5 points)

1. Un câble croisé est utilisé ? (0.25pt)
  - a. Pour connecter un hôte au routeur via le port console.
  - b. Pour connecter un hôte à un commutateur.
  - c. Pour connecter un hôte à un autre.
  - d. Pour connecter un commutateur à un routeur.
2. Un câble droit est utilisé ? (0.25pt)
  - a. Pour connecter un routeur via le port console.
  - b. Pour connecter un commutateur à un autre.
  - c. Pour connecter un hôte à un commutateur.
  - d. Pour connecter un routeur à un autre.

3. Quel type de câble est requis pour connecter un port console de routeur à un port COM de PC ? (Question éliminé)
  - a. Un câble droit
  - b. Un câble croisé
  - c. Un câble à paires torsadées.
  - d. Un câble coaxial
  - e. Un câble DB-9
4. Dans le modèle OSI, le dialogue entre couches homologues est défini par : (0.25pt)
  - a. Les services
  - b. Les interfaces
  - c. Les primitives
  - d. Aucune de ces réponses
5. La commande Ping utilise comme adresse ? (0.25pt)
  - a. L'adresse IP d'origine.
  - b. L'adresse MAC d'origine.
  - c. L'adresse IP de destination.
  - d. L'adresse MAC de destination.
  - e. Aucune de ses réponses
6. Que désigne-t-on par "bande passante" ? (0.25pt)
  - a. La quantité de données maximale transmissible par unité de temps
  - b. La quantité de données maximale transmise par unité de temps
  - c. Le nombre de bits transmis sur une ligne de transmission
  - d. Le nombre de bits transmis par bauds
  - e. Aucune de ces réponses
7. Dans la technique de commutation de trame « store and forward » le commutateur : (0.25pt)
  - a. Mémoire la trame entière.
  - b. Analyse la trame avant sa réception entière.
  - c. Commence à analyser la trame avant sa réception entière.
  - d. S'assure que la trame n'est pas corrompue.
  - e. Transmet la trame avant sa réception entière sans vérification.

#### Exercice 4 : (8.5 points)

##### Partie 1:

Pour des mesures de sécurité, les étudiants du département MI de l'Université d'Alger 1 ont développé leurs propres couches protocolaires. Dans cet exercice, nous allons étudier le protocole LDL2MI qui est le protocole utilisé au niveau de la couche 2. Le protocole LDL2MI suit les étapes suivantes avant de transmettre un message M1 :

**Etape 1:** Le protocole applique la méthode (VRC/LRC) sur le message M1. Pour chaque bloc de donnée composé de 3 bits, il rajoute un bit de parité paire (LRC). Ensuite, à la fin il rajoute les bits de parité verticale (VRC) pour obtenir un message M2.

**Etape 2:** Le protocole applique sur le message M2 la méthode de la division polynomiale binaire, le bloc de contrôle (CRC), pour obtenir un message M3 = M2R où R est le reste de division sur le polynôme générateur  $G(x) = x^3 + x^2 + 1$ .

**Etape 3:** Le protocole applique sur le message M3 une méthode de découpage en trame en utilisant un fanion spécial au protocole. Ce dernier rajoute au début et à la fin le fanion 01110, et dans le message M3, après chaque suite 11 il rajoute un 0. Le message obtenu à la fin (M4) est transmis à la couche physique.

1. Soit le message M1 = 101110, en appliquant le protocole LDL2MI donner les messages M2, M3 et M4. (1.5pt)

**Solution :**

101	0
110	0
011	0

**M2 =** 101011000110  
101011000110000

```
1101
----
11111000110000
1101
----
0101000110000
1101
----
11100110000
1101
----
0110110000
1101
----
00010000
1101
----
1010
1101
----
```

0111

**M3=** 101011000110111

**M4=** 0111010101100001100110101110

2. Expliquer les 3 étapes exécutées par le récepteur pour récupérer le message M1.

**Solution : (0.75pt)**

*Le protocole LDL2MI suit les étapes suivantes pour récupérer le message d'origine :*

**Etape 1:** Le protocole applique sur le message M4 reçu une méthode de récupération de trame. Ce dernier supprime du début et de la fin du message le fanion 01110, et après chaque suite 11 il supprime un 0. Le message obtenu à la fin est le message (M3).

**Etape 2:** Le protocole applique sur le message M3 la méthode de la division polynomiale binaire sur le polynôme générateur  $G(x) = x^3 + x^2 + 1$ . Le reste de la division détermine si le message a des erreurs ou pas. Si le reste est égal à zéro donc le message ne contient pas d'erreur. Ensuite le message M2 est récupéré en supprimant les bits de contrôle (les trois derniers bits) du message M3.

**Etape 3 :** Le protocole applique la méthode (VRC/LRC) sur le message M2. Pour chaque bloc de donnée composé de 3 bits, il vérifie le bit de parité paire (LRC). Ensuite, il vérifie les bits de parité verticale (VRC). Si les bits sont corrects le message M1 est récupéré en supprimant les bits de parités. Si une erreur est détectée par le protocole CRC dans le message M3, ce dernier sera corrigé par le protocole VRC/LRC si cette erreur se trouve sur un seul bit. Autrement, l'erreur ne pourra pas être localisée par le protocole.

3. Soient les deux messages ci-dessous reçus par un récepteur. Un des messages est erroné et le deuxième est correct.

Message 1 : M4 = 01110110110010110010110101110

Message 2 : M4 = 01110100011011001011011001110

- Quel est le message correct et le message erroné.
- Peut-on détecter l'endroit exact de l'erreur ? Si oui corriger là.
- Dans quel cas il est impossible de détecter l'endroit de l'erreur.
- Donner le message M1 correspondant au message (1 et 2) si c'est possible.

**Solution :**

Message 1 : M4 = 01110110110010110010110101110

Message 1 : M3 = 111101011010111 (0.25pt)

```

111101011010111
 1101
----
01001011010111
 1101
----
100011010111
 1101
----
10111010111
 1101
----
1101010111
 1101
----
000010111
 1101
----
 1101

```

```

1101
----

```

0000 (0.5 pt)

Le reste de la division == 0 donc le message est correct. (0.25 pt)

Message 1 : M2 = 1111 0101 1010

111	1
010	1
101	0

(0.25 pt)

Message 1 : M1 = 111010 (0.25pt)

---

Message 2 : M4 = 01110100011011001011011001110

Message 2 : M3 = 100011110101111 (0.25 pt)

```

100011110101111
 1101
----
10111110101111
 1101
----
1101110101111
 1101
----
000110101111
 1101
----
00001111
 1101
----

```

0010

(0.5 pt)

le reste de la division != 0 donc le message est erroné. (0.25 pt)

Message 2 : M2 = 100011110101

On essaye de détecter l'endroit exact de l'erreur en utilisant le protocole (VRC/LRC)

100	0
111	1
010	1

(0.25 pt)

On a trouvé qu'il existe une erreur au niveau du bit numéro 3 de la donnée.

(0.25pt)

**Hypothèse :** la donnée correcte devrait être  $M3=101011110101111$  (0.25pt)

On essaye de faire la division encore une fois pour voir si le message est correct :

```

101011110101111
  1101
  ----
    11111110101111
      1101
      ----
        0101110101111
          1101
          ----
            11010101111
              1101
              ----
                0000101111
                  1101
                  ----
                    11011
                      1101
                      ----
                        0001

```

0001  
(0.5pt)

Le résultat de la division est différent de zéro donc malgré la correction effectuée par le protocole (VRC/LRC) il existe encore des erreurs dans le message envoyé qui ne pourront pas être corrigées, et donc on ne connaît pas le message M1. (0.5pt)

- a. Quel est le message correct et le message erroné.

M1 : correcte M2 : Erroné

- b. Peut-on détecter l'endroit exact de l'erreur ? Si oui corriger là.

Oui il est possible grâce au protocole (VRC/LRC) mais seulement quand nous avons un seul bit erroné. Dans le message 2 il existe plus d'un bit et donc nous avons détecté un et nous l'avons corrigé mais il reste d'autres bits à corriger que nous n'avons pas pu détecter. (0.5 pt)

- c. Dans quel cas il est impossible de détecter l'endroit de l'erreur. (0.5 pt)

Il est impossible de détecter l'endroit de l'erreur quand nous avons par exemple 4 bits qui se trouvent sur deux lignes et deux colonnes superposées comme le montre cet exemple :

Message

101	0
111	1
010	1

envoyé :

Message reçu :

110	0
100	1
010	1

- d. Donner le message M1 correspondant au message (1 et 2) si c'est possible.

Message 1 :  $M1 = 111010$

Message 2 :  $M1 =$  on ne peut pas le savoir.

## Partie 2

Le département de biologie n'était pas d'accord avec le protocole proposé par le département MI et ils ont décidé de travailler avec le code de Hamming à parité impaire.

1. Soit le message  $M1 = 101110$ , quel est le message transmis ?

(0.5pt)

1	0	0	1	1	1	0	0	0	1
10	9	8	7	6	5	4	3	2	1

2. Montrer s'il y a une erreur dans  $M2 : 1101101$  ? Localiser l'erreur si elle existe. (0.5pt)

Recalcule des bit de contrôle.				1		0	1
	1	1	0	1	1	0	1
	7	6	5	4	3	2	1

Les bits de contrôles sont corrects et donc y'a pas d'erreur dans le message reçu.

NB : La calculatrice est interdite. Merci de justifier vos réponses.

Bon courage