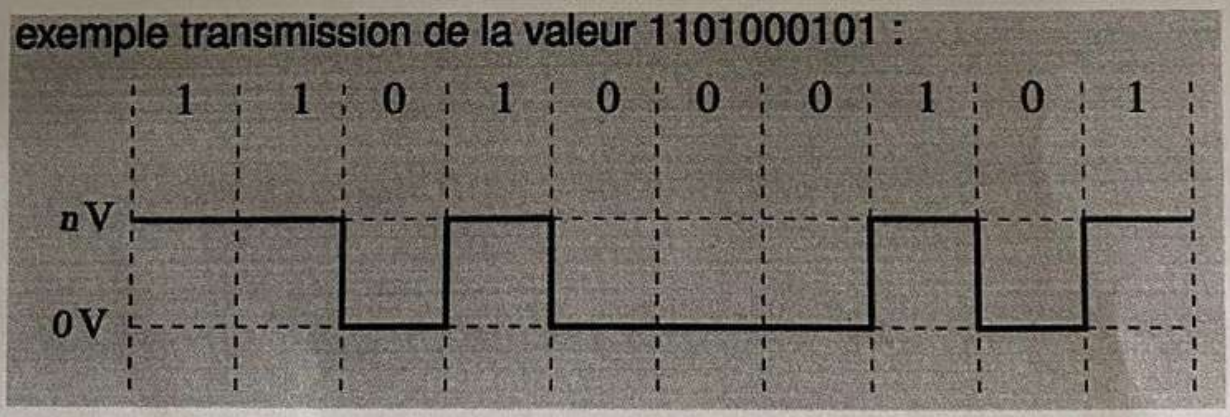
Travaux Dirigés

Thème : Codage des signaux et contrôle d'intégrité
Module : Principes et Fondements des Réseaux
Automne 2024
Filières : GNDC et Génie Informatique
Département : Génie Electrique et Informatique

Partie I- CODAGE

Introduction

réseaux locaux ⇒ distance relativement modérée entre deux ordinateurs, ainsi le signal émis sur un câble électrique reste relativement peu affaibli.
Transmission en **bande de base** : les données binaires codées par un signal numérique sont transmises directement sur le câble. Le codage le plus simple consiste à faire correspondre au bit 1 un signal électrique de tension *n* volts et au bit 0 un signal de tension nulle.



Problèmes posés par le codage trop simple :

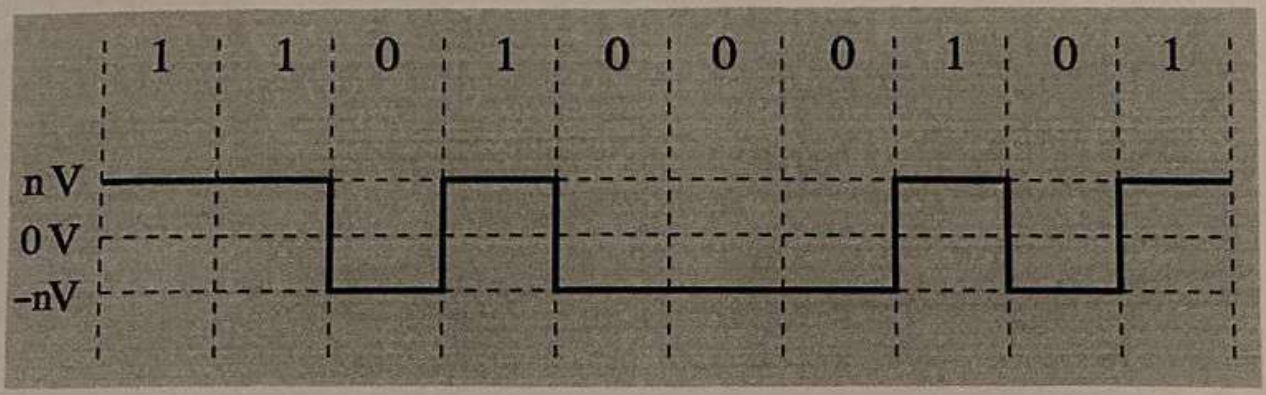
- une tension nulle correspond à l'envoi d'un 0 binaire mais peut aussi correspondre à l'absence d'envoi de données. Si une suite binaire comprend plusieurs 0 ou 1 binaires consécutifs, il faut que l'émetteur et le récepteur soient parfaitement synchronisés pour que le décodage se fasse correctement.

⇒ cela peut conduire le récepteur à ne pas reconnaître les données reçues.

Pour éliminer ces problèmes, plusieurs codes plus évolués ont été élaborés :

- le NRZ pour sa simplicité de conception,
- le code de Manchester pour sa mise en œuvre dans les réseaux Ethernet,
- le code de Manchester différentiel, le code de Miller, etc.

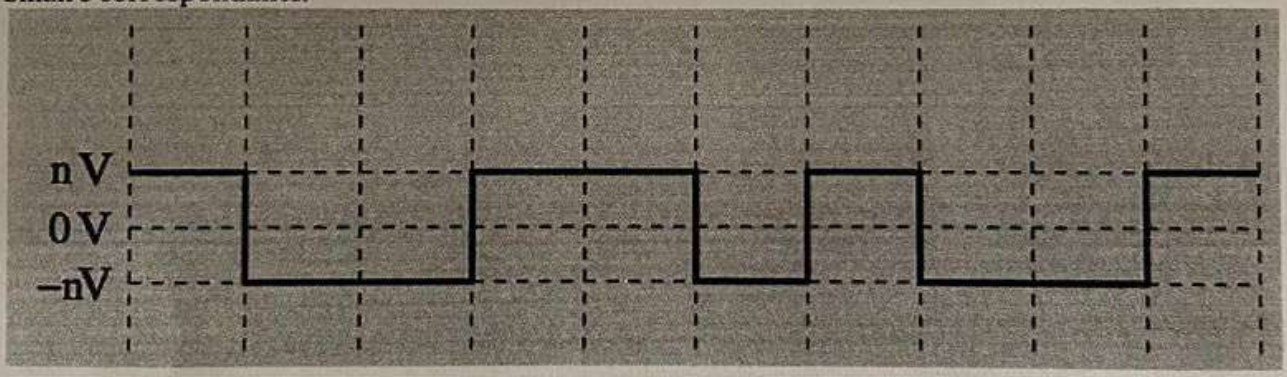
Code NRZ



Le code NRZI (No *Return to Zero Inverted)* est similaire au code NRZ mais les tensions associées aux valeurs binaires sont inversées : 1 est codé par une tension négative et 0 par une tension positive.

Exercice 1 : donner le signal correspondant à l'envoi du message 1100110101 avec le code NRZ.

Exercice 2 : Soit le signal suivant (codé avec NRZ), reçu sur un cable électrique, retrouver la trame binaire correspondante.

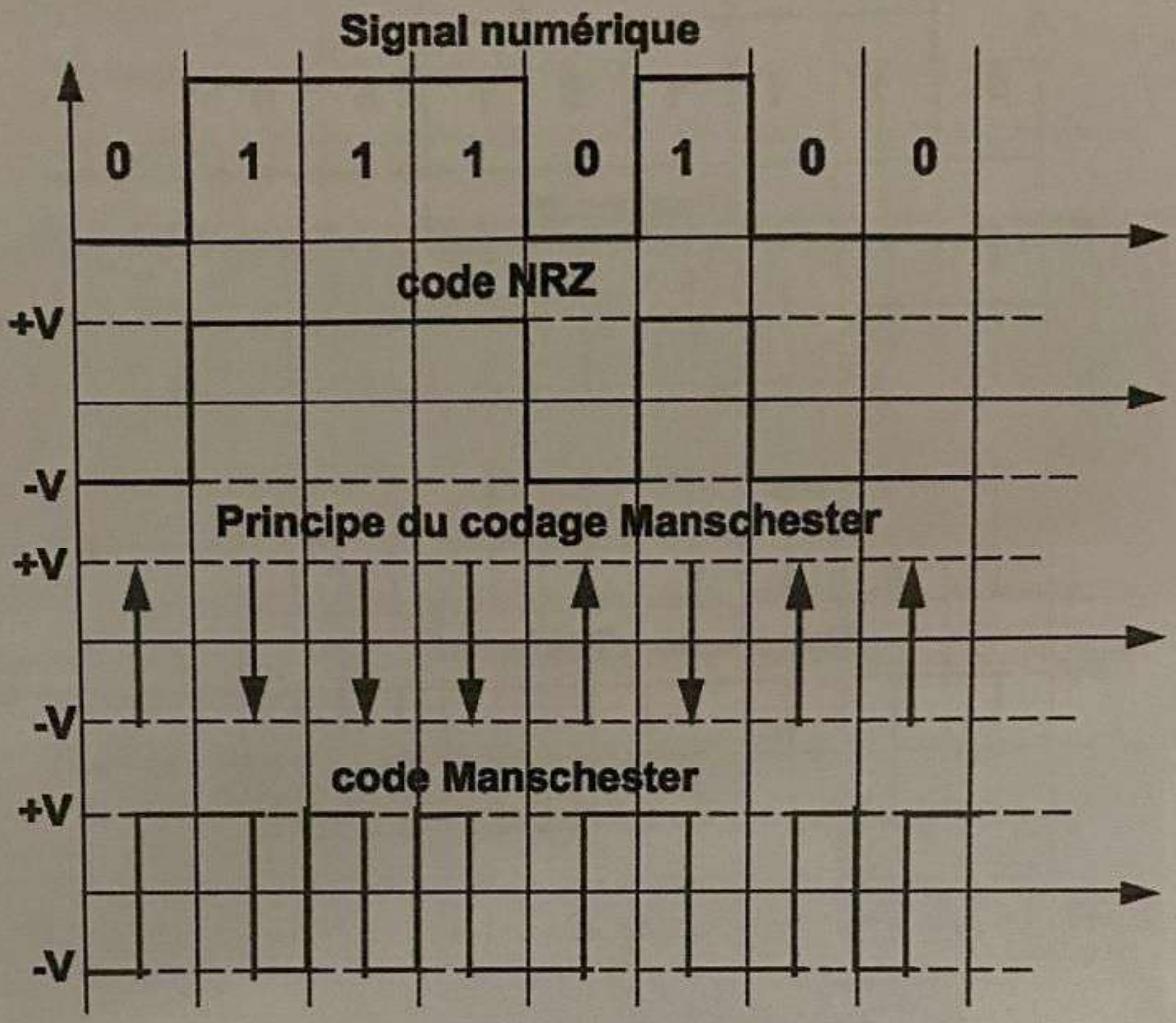


Exercice 3 : Donner la trame binaire si le code utilisé avait été NRZI.

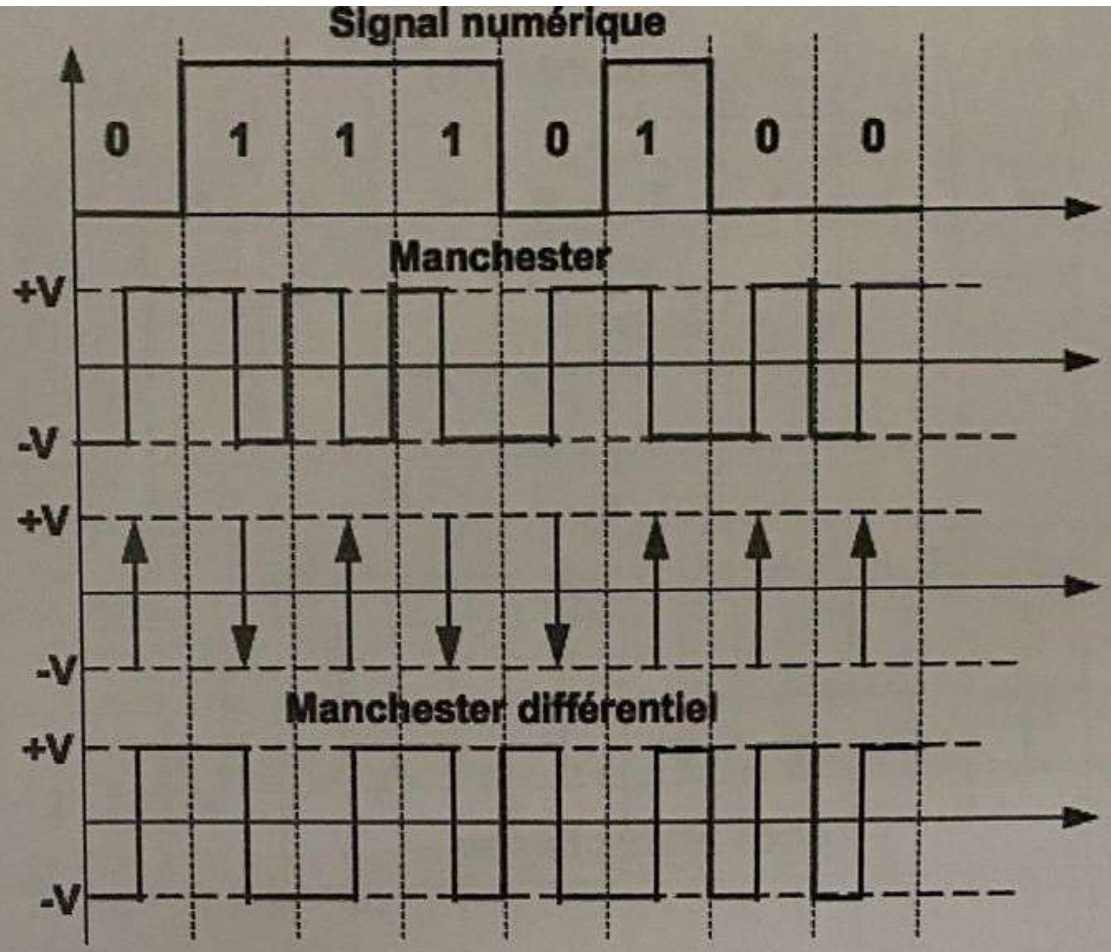
Code Manchester

Il est aussi appelé le code biphase.
Il propose une solution au problème de détection des longues chaînes de 0 ou 1.
Il s'agit d'un code basé sur les variations du signal : ce n'est plus la tension qui est importante mais la différence de signal.
1 est codé par un passage de la tension n à *-n* et 0 par le passage en sens inverse.

Avec une transition au milieu de chaque temps bit, le codage Manchester remédie à l'absence d'information de synchronisation. La transition est croissante pour les 0, décroissante pour les 1. Le sens des transitions est significatif, ce qui pose des problèmes en cas d'inversion des fils de liaison. Multipliant les transitions, le codage Manchester a un spectre très large, il est utilisé dans les réseaux locaux de type Ethernet sur câble coaxial. La bande passante du support y est importante et l'inversion de fils impossible.

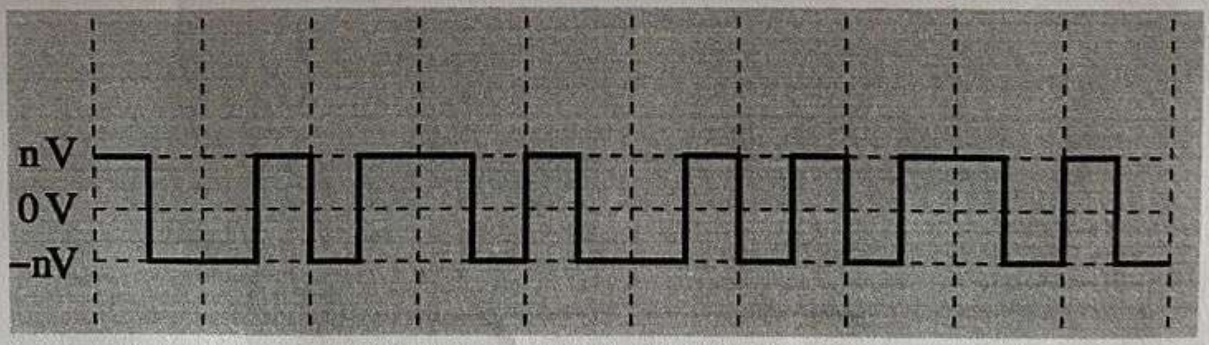


Le codage Manchester différentiel résout le problème d'inversion des conducteurs.
Chaque transition, au milieu du temps bit, est codée par rapport à la précédente. Si le bit à coder vaut 0, la transition est de même sens que la précédente ($\Delta\varphi = 0$) ; si le bit est à 1, on inverse le sens de la transition par rapport à celui de la précédente ($\Delta\varphi = \pi$). Ce codage résout la plupart des problèmes posés, mais son spectre est relativement large. Il est utilisé dans les réseaux locaux de type Token Ring.



Exercice 4 : donner le signal correspondant à l'envoi du message 1100110101 avec le code de manchester.

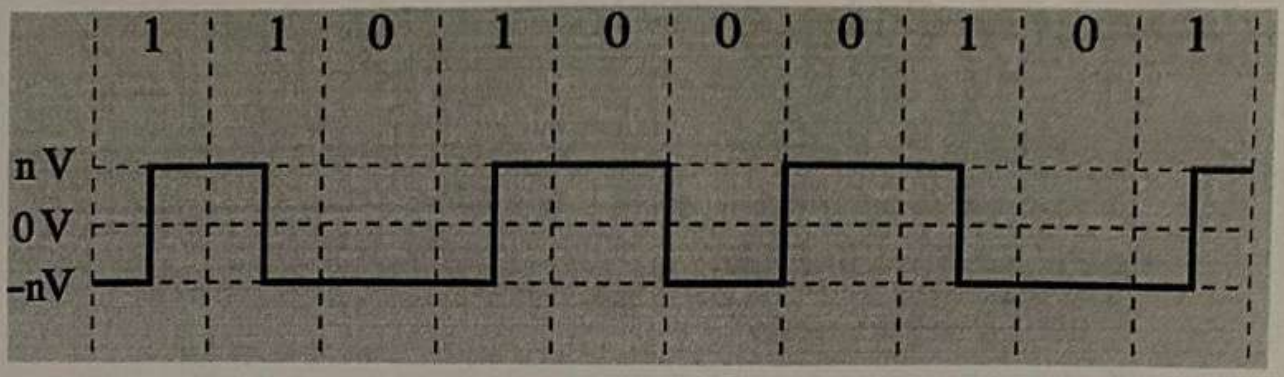
Exercice 5 : Soit le signal suivant (codé avec Manchester), reçu sur un cable électrique, retrouver la trame binaire correspondante.



Exercice 6 : Reprendre les exercices 1 et 5 avec le code de manchester différentiel.

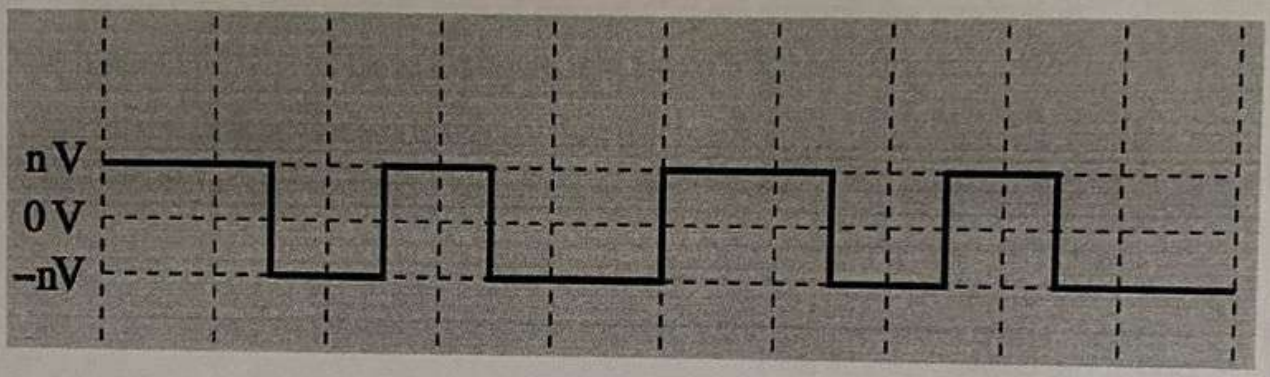
Code MILLER

Le bit 1 est codé par une transition en milieu de temps horloge et le bit 0 par une absence de transition.
Les longues suites de 0 posant toujours le problème de la synchronisation, si un bit 0 est suivi d'un autre 0 une transition est rajoutée à la fin du temps horloge.



Exercice 7 : donner le signal correspondant à l'envoi du message 1100110101 avec le code de Miller.

Exercice 8 : Soit le signal suivant (codé avec Miller), reçu sur un câble électrique, retrouver la trame binaire correspondante.



Partie II- CONTROLE D'INTEGRITE

Description du principe

Émission d'un mot :

- On choisit un polynôme générateur puis on le transforme en un mot binaire.
- Exemple : avec le polynôme générateur $x^4 + x^2 + x$, on obtient 10110.
- On ajoute m zéros au mot binaire à transmettre où m est le degré du polynôme générateur.
- Exemple : on souhaite transmettre le mot 11100111 en utilisant le polynôme générateur $x^4 + x^2 + x$, on obtient alors 111001110000.
- On va ajouter itérativement à ce mot, le mot correspondant au polynôme générateur jusqu'à ce que le mot obtenu soit inférieur au polynôme générateur. Ce mot obtenu correspond au CRC à ajouter au mot avant de l'émettre.
- On effectue donc une division euclidienne dans laquelle on ne tient pas compte du quotient.

Exemple : Soit 1000001110000100 l'information à transmettre. Alors le polynôme correspondant est :

$$P(x) = x^{15} + x^9 + x^8 + x^7 + x^2$$

Soit le polynôme de contrôle de degrès 12 suivant :

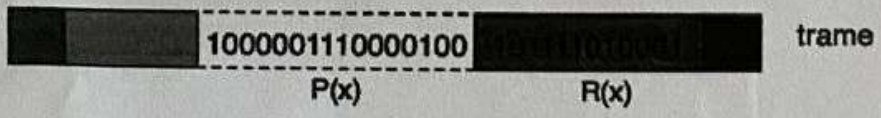
$$Q(x) = x^{12} + x^{11} + x^3 + x^2 + x + 1$$

La division de $x^{12} \times P(x)$ par $Q(x)$ donne le polynôme reste :

$$R(x) = (x^{12} \times P(x)) \mod Q(x) = x^{11} + x^9 + x^8 + x^7 + x^6 + x^4 + 1$$

L'information redondante à ajouter en fin de trame est donc : 101111010001.

La trame envoyée contient donc : 1000001110000100 101111010001.



❑ Exemples de codes polynômiaux :

(i) L'avis V41 du CCITT conseille l'utilisation de codes polynômiaux (de longueurs n = 260, 500, 980 ou 3860 bits) avec le polynôme générateur $G(x) = x^{16} + x^{12} + x^5 + 1$.

(ii) Le polynôme CRC-16 est utilisé par le protocole HDLC :

. $G(x) = x^{16} + x^{15} + x^2 + 1$.

(iii) Le polynôme suivant est utilisé par Ethernet :

. $G(x) = x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^8+x^7+x^5+x^4+x^2+1$.

■ Exercices :

On utilisera le polynôme générateur $x^4 + x^2 + x$.

1. On souhaite transmettre le message suivant :1111011101, quel sera le CRC à ajouter ?
2. Même question avec le mot 1100010101.
3. Je viens de recevoir les messages suivants : 1111000101010, 11000101010110, sont-ils corrects ?

Protocoles de correction des erreurs Correction par Hamming

Soit un mot à transmettre de 7 bits (code ASCII de la lettre a par exemple)

m =

1	1	0	0	0	0	1

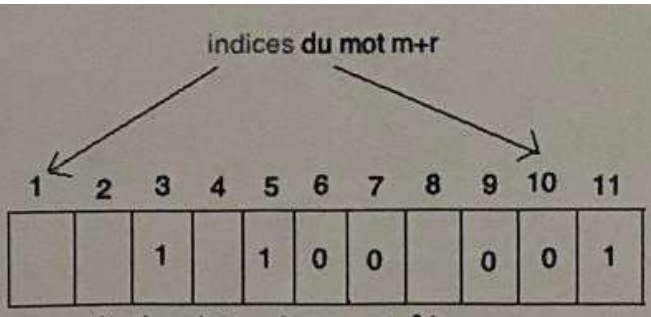
Étape 1 : déterminer r tel que $m + r + 1 \leq 2^r$

Dans notre cas r = 4

Étape 2 : on doit donc ajouter 4 bits de contrôle au mot m

Étape 3 : les bits de contrôle correspondent aux positions 2^i (i = 0,1,2,3) du nouveau mot à transmettre constitué de m+r bits

Il s'agit du mot :



Étape 4 : il reste à remplir les bits de contrôle,
(ayant dans ce cas les indices $1=2^0$, $2=2^1$, $4=2^2$ et $8 = 2^3$)

Étape 5 :

- Le bit d'indice 1 contrôle les bits : 1, 3, 5, 7, 9 et 11 ce qui correspond à la parité
- Le bit d'indice 2 contrôle les bits : 2, 6, 7, 10 et 11 ce qui correspond à la parité 1
- Le bit d'indice 4 contrôle les bits : 4, 5, 6 et 7 ce qui correspond à la parité 1
- Le bit d'indice 8 contrôle les bits : 8, 9, 10 et 11 ce qui correspond à la parité 1

Le mot à transmettre est :

1	2	3	4	5	6	7	8	9	10	11
1	1	1	1	1	0	0	1	0	0	1

À la réception on recalcule les bits de contrôle
Si le résultat est égale à zéro, il n'y a pas d'erreur
Sinon, la position de l'erreur est donnée par le mot constitué des bits de contrôle

Exemple : si on reçoit le mot : 11011001001,
détecter et corriger l'erreur s'elle existe !