

Protocole Ethernet

Couches liaison de données

- 4 -

Module:

Réseaux Informatique

- Téléinformatique -

A.Mazoul

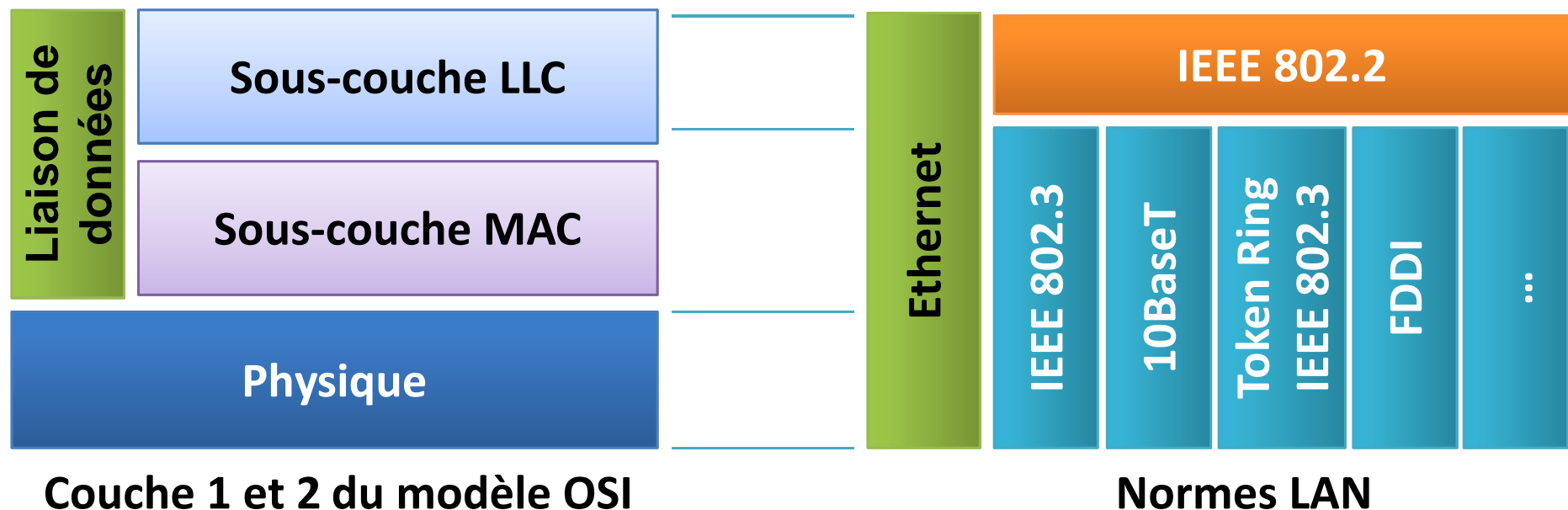
Département Informatique

Mise à jour: 29/10/2013



Normes IEEE & modèle OSI

- L'**IEEE** (*Institute of Electrical and Electronic Engineers*) est un organisme professionnel qui définit les normes touchant les réseaux.
- Les normes de l'**IEEE** (dont **IEEE 802.3** et **IEEE 802.5**) sont actuellement les normes prédominantes et les plus connues dans le monde en matière de LAN.
 - **Sous-couche LLC**: gère les communications entre les dispositifs sur une seule liaison d'un réseau.
 - **Sous-couche MAC**: concerne les protocoles que doit suivre un ordinateur hôte pour accéder au média physique.



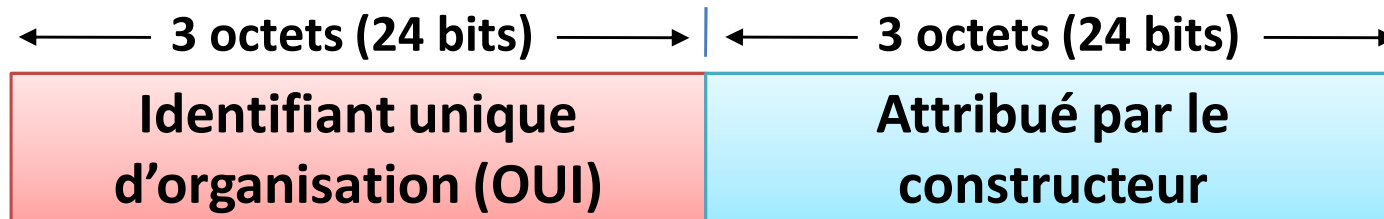
└ Normes IEEE

Normes	Appellation	Débits	Médias utilisés
802.3	Ethernet	10 Mbps	Coaxial/UTP/ Fibre optique
802.3u	Fast Ethernet	100 Mbps	UTP / Fibre optique
802.3z	Gigabit Ethernet	1000 Mbps	Fibre optique
802.3ab	Gigabit Ethernet	1000 Mbps	Câble UTP
802.3ae	10 Gigabit Ethernet	10000 Mbps	Fibre optique

└ Adresse MAC

- **Adresse MAC** (*Media Access Control*): est une adresse physique se trouvant par défaut sur les carte réseaux, et attribuée par le constructeur. Elle permet d'identifier les ordinateurs de façon unique sur le réseau.

Le processus de l'encapsulation de la couche **Liaison de données** permet l'ajout des adresses d'origine et de destination dans la **Trame**.



```
C:\>getmac /v
```

Nom de la conne	Carte réseau	Adresse physique
Connexion au ré	Realtek PCIe FE	1C-C1-DE-A3-3E-A2
Connexion résea	Realtek RTL8191	1C-65-9D-33-99-51
-4598-B13A-7401	4D1D3118>	

Structure des trames Ethernet & IEEE 802.3

IEEE 802.3						
7	1	6	6	2	de 46 à 1500	4
Préambule	Délimiteur de début de trame	Adresse de destination	Adresse source	Type de longueur	En-tête et données 802.2	Séquence de contrôle de trame

Ethernet					
8	6	6	2	de 46 à 1500	4
Préambule	Adresse de destination	Adresse source	Type	Données	Séquence de contrôle de trame

Champs des trames Ethernet IEEE 802.3

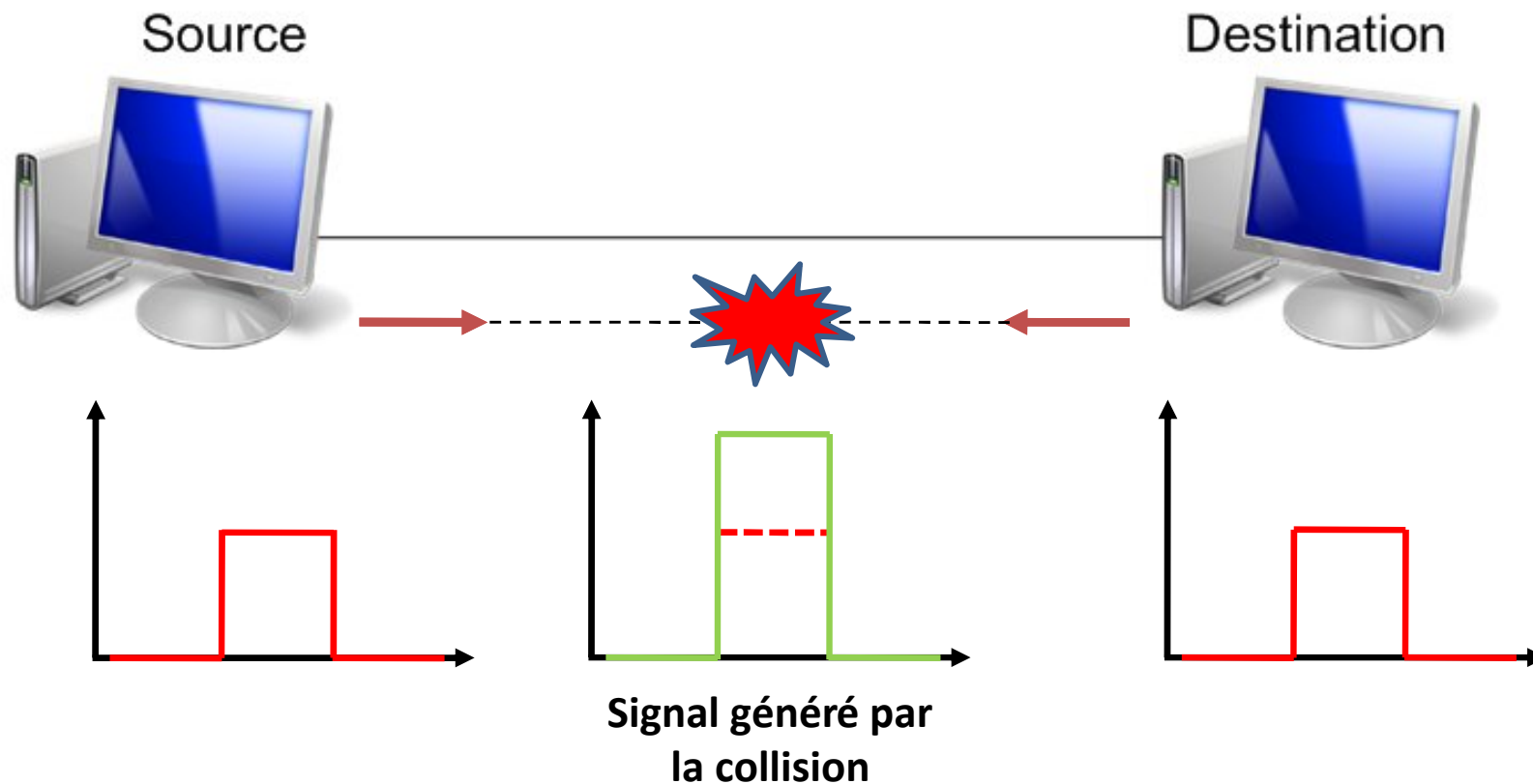
Octets	Description
• 7	Préambule
• 1	Délimiteur de début de trame (SFD)
• 6	Adresse MAC de destination
• 6	Adresse MAC source
• 2	Champ Longueur/Type (si la longueur est inférieure à 0600 en hexadécimal, sinon Type de protocole)
• de 46 à 1500	Données* (si inférieures à 46 octets, des données de remplissage doivent être ajoutées à la fin)
• 4	Séquence de contrôle de trame (Somme de contrôle CRC)

Champ type détermine le protocole de la couche supérieure:

- 0x0806 = ARP
- 0x8035 = RARP
- 0x0800 = IPv4
- 0x86DD = IPv6

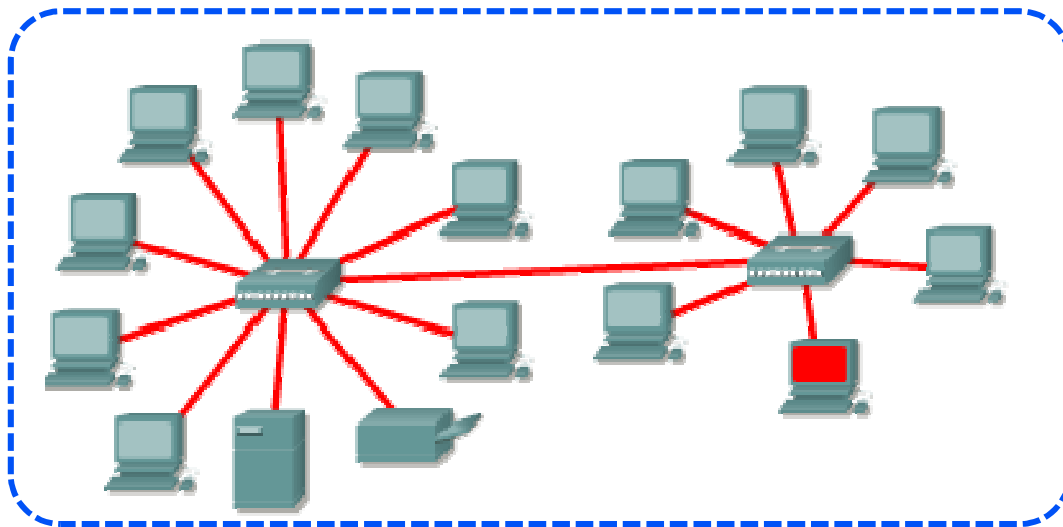
Les signaux dans une collision

- **La collision:** Produite lorsque deux ordinateurs émettent simultanément des signaux sur le même média. *Deux signaux ne peuvent pas circuler simultanément sur le même média.*

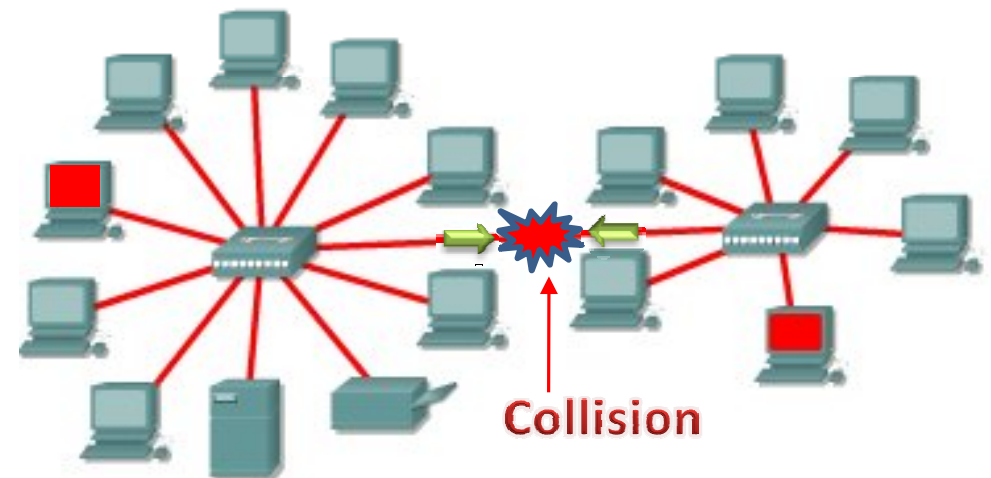


└─ Domaine de collision

- **Domaine de collision**: est une zone logique d'un réseau informatique dans laquelle tous les équipements réseau reçoivent la même trame. Ainsi, les paquets de données peuvent entrer en collision entre eux.



Domaine de Collision



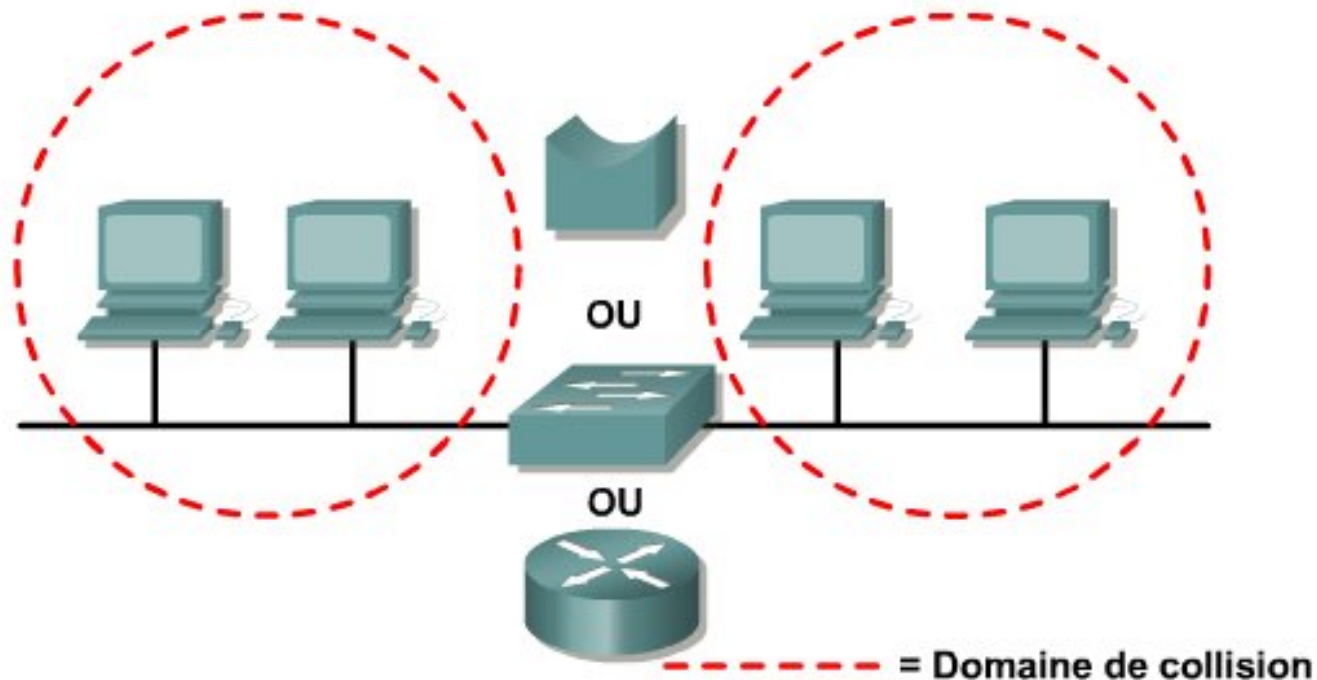
└─ Domaine de diffusion

■ **Domaine de diffusion** (***broadcast domain***): c'est une zone d'un réseau informatique où n'importe quel ordinateur connecté au réseau peut directement transmettre à tous les autres ordinateurs du même domaine, sans devoir passer par un routeur.

- Le **broadcast** se fait par l'envoi d'une trame dont l'adresse MAC de destination est **FF:FF: FF: FF: FF: FF** .
- Généralement l'hôte qui reçoit une trame **broadcast** doit alors envoyer une réponse à cette adresse source.

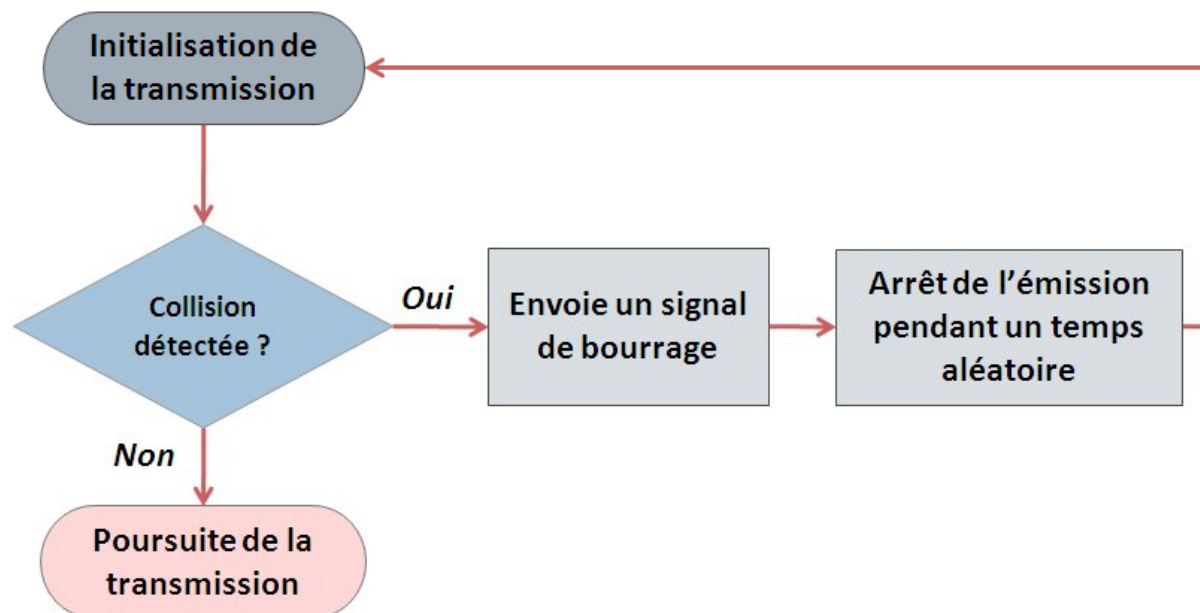
Segmentation des domaines de collision

- **Segmentation** : permet de réduire la taille des domaines de collision en les segmentant à l'aide des équipements de réseau intelligents (commutateur, pont et routeur).



CSMA/CD

- **CSMA/CD** (*Carrier Sense Multiple Access / Collision Detection*) : permet de gérer l'accès au média.
- **CSMA** : Permet de déterminer si une autre station n'est pas déjà en train de transmettre une trame de données par détection d'une tension électrique ou présence de lumière.
- **CD** : Lors d'une collision, l'émission est stoppée immédiatement. Le système se remet en attente pendant un délais aléatoire avant de lancer une nouvelle séquence de CSMA pour tenter la réémission de la trame.

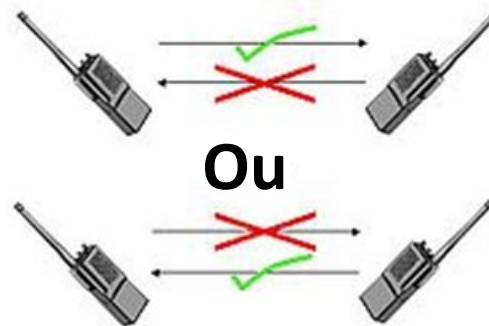


Canaux de Communication (duplex)

- Nous distinguons trois types de canaux de communication (duplex):
 - **simplex**: canal **unidirectionnel** qui transporte l'information dans un seul sens.
 - **half-duplex**: également appelé à **l'alternat**, canal **bidirectionnel** qui permet de transporter des informations dans les deux sens, **mais pas simultanément**.
 - **full-duplex**: canal **bidirectionnel** où l'information peut être transportée simultanément dans **les deux sens**.



Simplex



Half-duplex



full-duplex

↳ Duplex & CSMA/CD

■ **Half Duplex**

- Utilisation d'une seule paire du câble
- Emission/Réception alternées
- Problème de collisions (utilisation de **CSMA/CD**)

■ **Full Duplex**

- Utilisation de deux paires du câble
- Emission/Réception simultanées
- Pas de collisions (pas besoin de **CSMA/CD**)

Préambule & délimiteur

- Dans le cas d'une Trame Ethernet, les données de la trame sont délimitées par une suite de bits particulière définissant:
 - **Préambule** (en **7 octets**): qui sert à la synchronisation du signal.
 - **Délimiteur** (en **1 octet**): appelé également drapeau ou fanion, et qui sert à signaler le début des données de la trame.

Exemple de fanion: **01111110**

Pour assurer la transparence binaire un « 0 » doit toujours être inséré après toute suite de cinq « 1 » consécutifs (Technique utilisée dans: HDLC et PPP).

Données →

0011011111100101011111011



Trame →

01111110

0011011111101001010111110011

01111110

└ Préambule & délimiteur: Exercice

Exercice:

On s'intéresse à la structuration en trame des bits transmis et à leur transmission via une procédure HDLC. On veut transmettre les données suivantes (fragment d'une trame incomplète) :

01111000111111001111110111000001111100001

Quelle est la suite de bits émise sur le support?

└ Séquence de contrôle de la trame (FCS)

- Le **FCS** (***Frame Check Sequence***) est un champ de 4 octets qui sert à contrôler l'intégrité de la trame, c'est-à-dire la détection, et éventuellement la correction des erreurs pouvant survenir pendant la transmission.

■ On distingue plusieurs techniques de contrôle d'erreur:

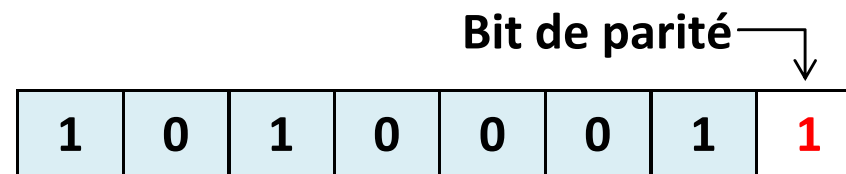
- Contrôle par bit de parité (**VRC** & **LRC**).
- Contrôle de redondance cyclique (**CRC**).
- Distance de Hamming.
- ...

└─ Contrôle de parité

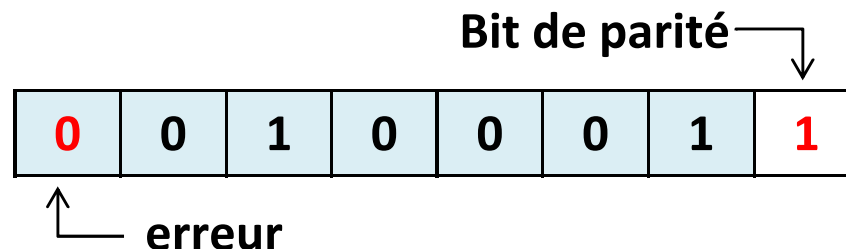
- Le **Contrôle de parité** consiste à ajouter un bit supplémentaire (appelé bit de parité) à un certain nombre de bits de données tel que le nombre total de bits à 1 soit pair.
- Nous utilisons deux types de contrôle de parité:
 - **VRC (Vertical Redundancy Checking)**: Vérification par redondance verticale et qui ne permet de détecter que les erreurs portant sur un nombre impair de bits.
 - **LRC (Longitudinal Redundancy Check)**: contrôle longitudinal de redondance: il s'applique à la totalité des données. le **LRC** est lui-même protégé par un bit de parité VRC.

Contrôle de parité: VRC

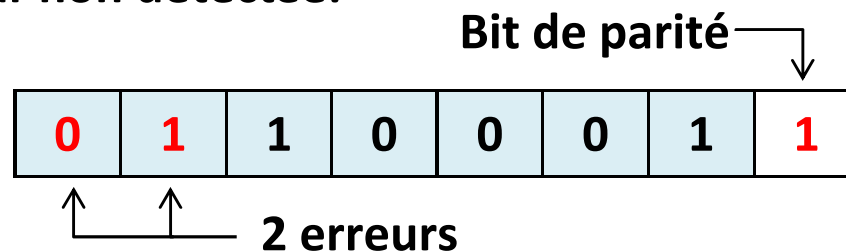
- **VRC** consiste à ajouter un 1 si le nombre de bits « 1 » du mot de code est impair, 0 dans le cas contraire.



➤ Exemple d'erreur détectée:



➤ Exemple d'erreur non détectée:



Contrôle de parité: LRC

- **LRC** (appelé également *Contrôle de parité croisé*) consiste non pas à contrôler l'intégrité des données d'un caractère, mais à contrôler l'intégrité des bits de parité d'un bloc de caractères.

Soit « **HELLO** » le message à transmettre, en utilisant le code ASCII standard. Voici les données telles qu'elles seront transmises avec le contrôle de parité **VRC** et **LRC**:

Lettr e	Code ASCII sur 7 bite							Bit de parité (LRC)
H	1	0	0	1	0	0	0	0
E	1	0	0	0	1	0	1	1
L	1	0	0	1	1	0	0	1
L	1	0	0	1	1	0	0	1
O	1	0	0	1	1	1	1	1
VRC	1	0	0	0	0	1	0	0

En absence d'erreur

Lettr e	Code ASCII sur 7 bite							Bit de parité (LRC)
H	1	0	0	1	0	0	0	0
E	1	0	0	0	1	1	1	1
L	1	0	0	1	1	0	0	1
L	1	0	0	1	1	0	0	1
O	1	0	0	1	1	1	1	1
VRC	1	0	0	0	0	1	0	0

En cas d'erreur

↳ Contrôle de redondance cyclique (CRC)

■ **CRC** (*Cyclic Redundancy Codes* ou *Codes à redondance cyclique*) : est un mécanisme qui consiste à contrôler l'intégrité des données de la trame, en se basant sur l'utilisation d'un polynôme de référence appelé polynôme générateur $G(x)$.

Ce code est basé sur le fait que toute chaîne binaire permet de construire un polynôme, chaque bit donne sa valeur au coefficient polynomial correspondant.

Exemple:

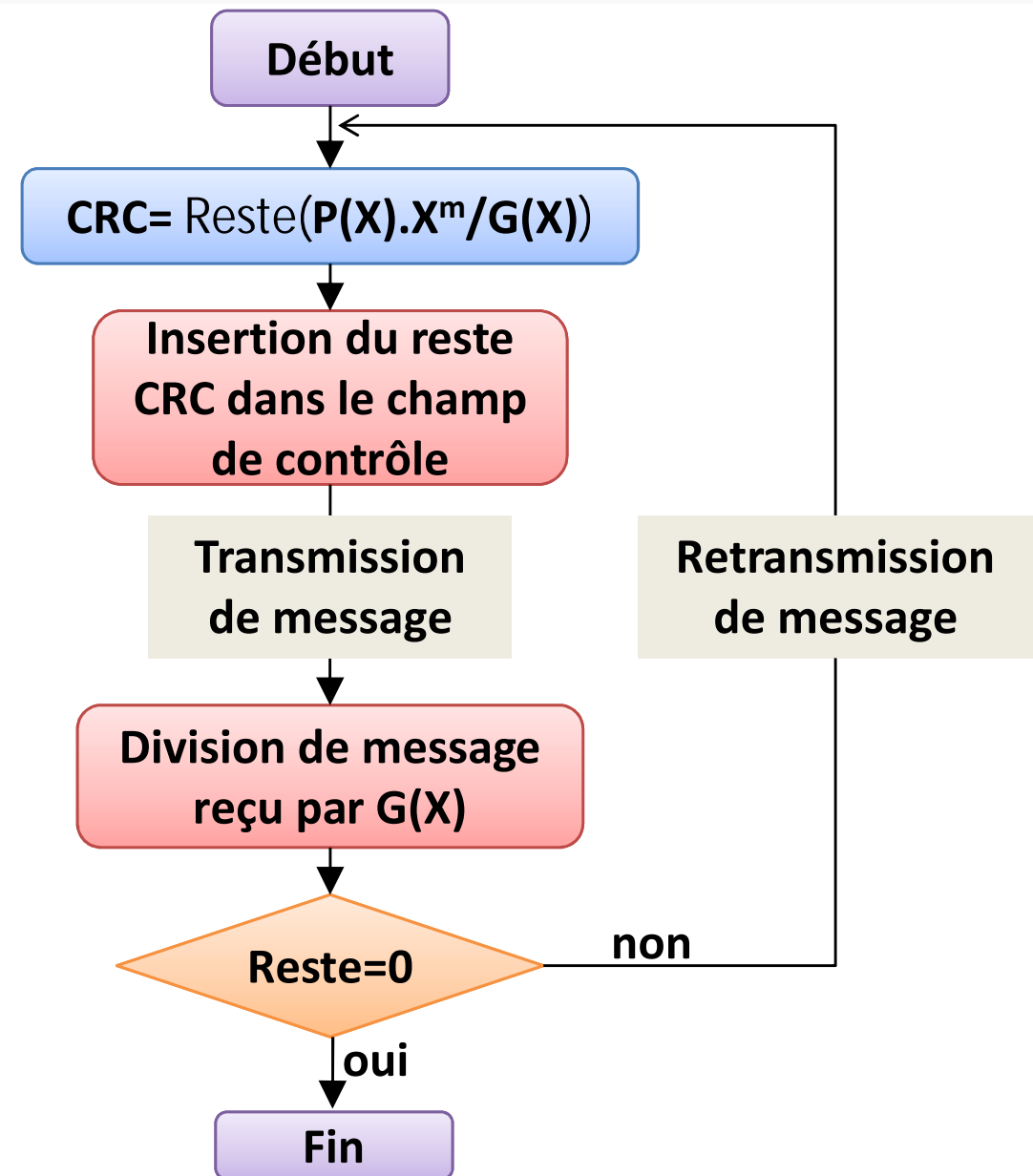
la séquence binaire **0110101001** peut être représentée sous la forme polynomiale suivante :

$$P(X) = 0 \cdot X^9 + 1 \cdot X^8 + 1 \cdot X^7 + 0 \cdot X^6 + 1 \cdot X^5 + 0 \cdot X^4 + 1 \cdot X^3 + 0 \cdot X^2 + 0 \cdot X^1 + 1 \cdot X^0$$



$$P(X) = X^8 + X^7 + X^5 + X^3 + 1$$

CRC: Algorithme



$$P(X).X^m = G(X).Q(X) + R(X)$$

Où

$R(X)$ = CRC

$G(X)$ = polynôme générateur

m = degré du polynôme de $G(x)$

$Q(X)$ = Quotient

CRC: Application pratique

Prenons le message M de 11 bits suivant : **10100101001**

G(X) = $X^3 + 1$: le polynôme générateur de degré **3** (en binaire **1001**)

$$P(X).X^3 = \mathbf{10100101001000}$$

$$\text{Reste}(P(X).X^3 / G(X)) = \mathbf{010}$$

Message à transmettre :

$$M' = \mathbf{10100101001010}$$

En absence d'erreur :

$$\text{Reste}(M'/G(X)) = \mathbf{0}$$

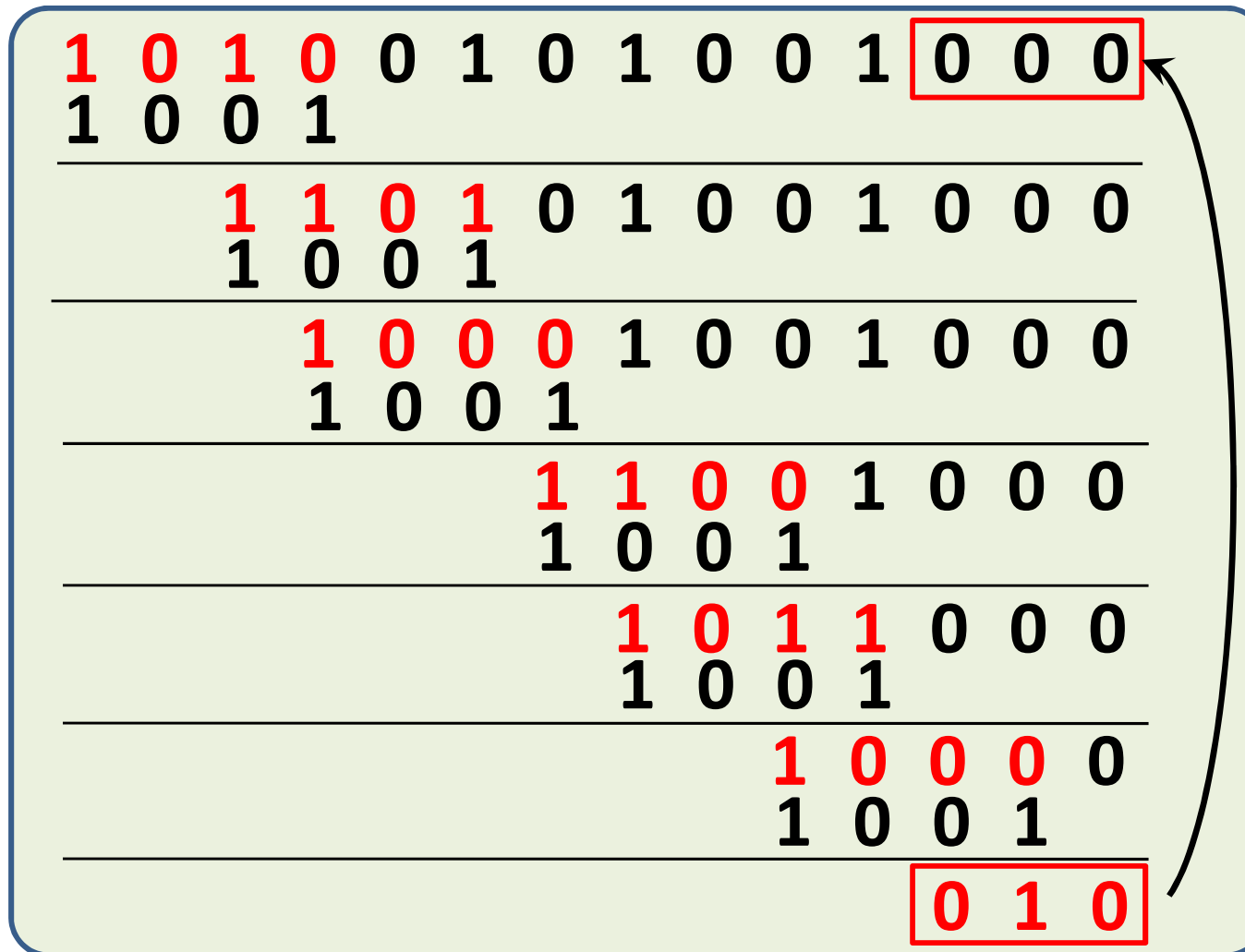
En cas d'erreur :

$$M'' = \mathbf{10101101001010}$$

$$\text{Reste}(M''/G(X)) = \mathbf{1 \neq 0}$$

CRC: Calcul de CRC

■ Exemple du calcul du CRC avant émission d'un mot :



■ Le **CRC** est donc **010** et le mot à transmettre **10100101001010**.

CRC: Application pratique

```

10100101001010
1001

```

```

110101001010
1001

```

```

10001001010
1001

```

```

11001010
1001

```

```

1011010
1001

```

```

10010
1001

```

0

En absence d'erreur

```

10101101001010
1001

```

```

111101001010
1001

```

```

11001001010
1001

```

```

1011001010
1001

```

```

10001010
1001

```

```

11010
1001

```

```

1000
1001

```

1

En cas d'erreur

└ CRC: Application pratique

Exercice: (voir TD2)

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 ?

└ CRC: Polynômes générateurs

■ Les polynômes générateurs les plus couramment employés sont :

CRC-12 : $X^{12} + X^{11} + X^3 + X^2 + X + 1$

CRC-16 : $X^{16} + X^{15} + X^2 + 1$

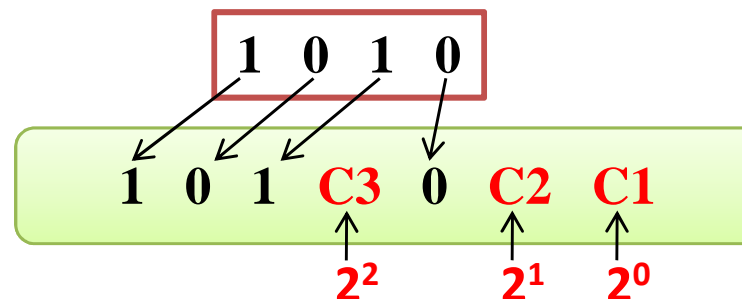
CRC-16-CCITT (HDLC, X.25): $X^{16} + X^{12} + X^5 + 1$

CRC-32 (Ethernet) :

$$X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$$

Code de hamming

- **Code de hamming**: est un code correcteur d'erreur permettant la détection et la correction des erreurs de transmission. il utilise des bits de contrôle correspondants aux positions 2^i (i.e. 1, 2, 4,...). Ces bits ressemblent aux bits de parité.
- Chaque bit du message sera contrôlé par sa décomposition en somme de 2^i .
- Exemple de transmission d'un message de 4 bits: **1010**



A/B est utilisé pour désigner la transmission de **A** bits avec **B** bits de contrôle.

Code de hamming

1 0 1 **C3** 0 **C2** **C1**

- Le bit **C1** contrôle la parité des bits **7** (111), **5** (101) et **3** (011).
- Le bit **C2** contrôle la parité des bits **7** (111), **6** (110) et **3** (011).
- Le bit **C3** contrôle la parité des bits **7** (111), **6** (110) et **5** (101).

C_i contrôle les bits dont la position en binaire correspond au cas où **C_i=1**

	Bits contrôlés	Valeur des bits contrôlé	Bit de Parité
C1	7,5,3	1,1,0	0
C2	7,6,3	1,0,0	1
C3	7,6,5	1,0,1	0

■ La séquence de bits à transmettre sera: **1010010**

Code de hamming

Exercice: (voir TD2)

En utilisant le code de hamming 7/4, Soit un mot de Hamming reçu:

1 0 1 1 0 0 0 1 0 0 1

1. Quels sont les bits de contrôle de parité ?
2. Est-ce que le message reçu correspond au message transmis ?
3. Si non, quel a été le message transmis ?

Distance de hamming & Taux d'erreur binaire

- **Distance de hamming** entre deux séquences binaires $m1$ et $m2$ de même taille est le nombre de bits de même rang par lesquels ces deux séquences diffèrent. Elle est notée $d(m1;m2)$.

Exemple :

$$d(1100110; 1010110) = 2$$

- **Taux d'erreur binaire (Teb)** ou **BER** (*Bit Error Rate*): le rapport entre le nombre de bits erronés et le nombre de bits transmis. Le taux d'erreur (BER) s'exprime en puissance négative 10^{-i} .

$$\text{BER} = \frac{\text{Nbre de bits erronés}}{\text{Nbre de bits transmis}}$$

Distance de hamming & Taux d'erreur binaire

Exemple:

- Message transmis : **M = 0110010011001001001010**
- Message reçu: **M' = 011011001110100001101010**

$$\text{BER} = \frac{d(M, M')}{24} = \frac{4}{24} = 0,1666$$

Soit **Teb** Taux d'erreur binaire ou la probabilité d'avoir un bit erroné dans un message:

- **Pc = (1-Teb)^N** est la probabilité de recevoir un message correct.
- **Pe = 1 - (1-Teb)^N** est la probabilité de recevoir un message erroné.

merci de votre attention

