

# Réseaux et Protocoles

Auteur Dr. Younes Jabrane  
ENSA, Marrakech  
Université Cadi Ayyad

## Chapitre 2 : Les protocoles de liaison de données

# Plan du cours :

- 1 Rôle et fonctions d'un protocole de liaison
  - Mise en forme des données

# Plan du cours :

- 1 Rôle et fonctions d'un protocole de liaison
  - Mise en forme des données
  - Contrôle de la validité de l'information transmise

# Plan du cours :

- 1 Rôle et fonctions d'un protocole de liaison
  - Mise en forme des données
  - Contrôle de la validité de l'information transmise
  - Modes d'exploitation d'une liaison de données

# Plan du cours :

- 1 Rôle et fonctions d'un protocole de liaison
  - Mise en forme des données
  - Contrôle de la validité de l'information transmise
  - Modes d'exploitation d'une liaison de données
- 2 Fonctionnalités d'un protocole de liaison

# Plan du cours :

- 1 Rôle et fonctions d'un protocole de liaison
  - Mise en forme des données
  - Contrôle de la validité de l'information transmise
  - Modes d'exploitation d'une liaison de données
- 2 Fonctionnalités d'un protocole de liaison
  - Représentation des échanges de données

# Plan du cours :

- 1 Rôle et fonctions d'un protocole de liaison
  - Mise en forme des données
  - Contrôle de la validité de l'information transmise
  - Modes d'exploitation d'une liaison de données
- 2 Fonctionnalités d'un protocole de liaison
  - Représentation des échanges de données
  - Contrôle de flux



# Plan du cours :

- 1 Rôle et fonctions d'un protocole de liaison
  - Mise en forme des données
  - Contrôle de la validité de l'information transmise
  - Modes d'exploitation d'une liaison de données
- 2 Fonctionnalités d'un protocole de liaison
  - Représentation des échanges de données
  - Contrôle de flux
  - Gestion des acquittements

# Plan du cours :

- 1 Rôle et fonctions d'un protocole de liaison
  - Mise en forme des données
  - Contrôle de la validité de l'information transmise
  - Modes d'exploitation d'une liaison de données
- 2 Fonctionnalités d'un protocole de liaison
  - Représentation des échanges de données
  - Contrôle de flux
  - Gestion des acquittements
  - Numérotation des trames d'information

# Plan du cours :

- 1 Rôle et fonctions d'un protocole de liaison
  - Mise en forme des données
  - Contrôle de la validité de l'information transmise
  - Modes d'exploitation d'une liaison de données
- 2 Fonctionnalités d'un protocole de liaison
  - Représentation des échanges de données
  - Contrôle de flux
  - Gestion des acquittements
  - Numérotation des trames d'information
  - Notion de fenêtre glissante

# Plan du cours :

- 1 Rôle et fonctions d'un protocole de liaison
  - Mise en forme des données
  - Contrôle de la validité de l'information transmise
  - Modes d'exploitation d'une liaison de données
- 2 Fonctionnalités d'un protocole de liaison
  - Représentation des échanges de données
  - Contrôle de flux
  - Gestion des acquittements
  - Numérotation des trames d'information
  - Notion de fenêtre glissante
- 3 Description du protocole HDLC (High level Data Link Control)

# Plan du cours :

- 1 Rôle et fonctions d'un protocole de liaison
  - Mise en forme des données
  - Contrôle de la validité de l'information transmise
  - Modes d'exploitation d'une liaison de données
- 2 Fonctionnalités d'un protocole de liaison
  - Représentation des échanges de données
  - Contrôle de flux
  - Gestion des acquittements
  - Numérotation des trames d'information
  - Notion de fenêtre glissante
- 3 Description du protocole HDLC (High level Data Link Control)
  - Structure d'une trame HDLC

# Plan du cours :

- 1 Rôle et fonctions d'un protocole de liaison
  - Mise en forme des données
  - Contrôle de la validité de l'information transmise
  - Modes d'exploitation d'une liaison de données
- 2 Fonctionnalités d'un protocole de liaison
  - Représentation des échanges de données
  - Contrôle de flux
  - Gestion des acquittements
  - Numérotation des trames d'information
  - Notion de fenêtre glissante
- 3 Description du protocole HDLC (High level Data Link Control)
  - Structure d'une trame HDLC
  - Différents types de trames HDLC

# Plan du cours :

- 1 Rôle et fonctions d'un protocole de liaison
  - Mise en forme des données
  - Contrôle de la validité de l'information transmise
  - Modes d'exploitation d'une liaison de données
- 2 Fonctionnalités d'un protocole de liaison
  - Représentation des échanges de données
  - Contrôle de flux
  - Gestion des acquittements
  - Numérotation des trames d'information
  - Notion de fenêtre glissante
- 3 Description du protocole HDLC (High level Data Link Control)
  - Structure d'une trame HDLC
  - Différents types de trames HDLC
- 4 Cas particulier du protocole PPP (Point to Point Protocol)

# Plan du cours :

- 1 Rôle et fonctions d'un protocole de liaison
  - Mise en forme des données
  - Contrôle de la validité de l'information transmise
  - Modes d'exploitation d'une liaison de données
- 2 Fonctionnalités d'un protocole de liaison
  - Représentation des échanges de données
  - Contrôle de flux
  - Gestion des acquittements
  - Numérotation des trames d'information
  - Notion de fenêtre glissante
- 3 Description du protocole HDLC (High level Data Link Control)
  - Structure d'une trame HDLC
  - Différents types de trames HDLC
- 4 Cas particulier du protocole PPP (Point to Point Protocol)



## Chapitre II

### Les protocoles de liaison de données

- Le circuit de données : altérer les informations transportées
- Le protocole de liaison de données le supervise et définit un ensemble de règles pour assurer la fiabilité des échanges sur une liaison de données

## Chapitre II

### Les protocoles de liaison de données

- Le circuit de données : altérer les informations transportées
- Le protocole de liaison de données le supervise et définit un ensemble de règles pour assurer la fiabilité des échanges sur une liaison de données
- Le protocole spécifie le format des unités de données échangées (les trames), leur délimitation, les moyens de contrôler leur validité (parité, code polynomial...), ainsi que le mode de correction des erreurs détectées

## Chapitre II

### Les protocoles de liaison de données

- Le circuit de données : altérer les informations transportées
- Le protocole de liaison de données le supervise et définit un ensemble de règles pour assurer la fiabilité des échanges sur une liaison de données
- Le protocole spécifie le format des unités de données échangées (les trames), leur délimitation, les moyens de contrôler leur validité (parité, code polynomial...), ainsi que le mode de correction des erreurs détectées
- Il fixe également les règles du dialogue entre les deux extrémités de la liaison

## Chapitre II

### Les protocoles de liaison de données

- Le circuit de données : altérer les informations transportées
- Le protocole de liaison de données le supervise et définit un ensemble de règles pour assurer la fiabilité des échanges sur une liaison de données
- Le protocole spécifie le format des unités de données échangées (les trames), leur délimitation, les moyens de contrôler leur validité (parité, code polynomial...), ainsi que le mode de correction des erreurs détectées
- Il fixe également les règles du dialogue entre les deux extrémités de la liaison
- Il exerce en outre deux fonctions importantes : le contrôle de flux (mécanisme vérifiant le rythme d'envoi des informations)

# Chapitre II

## Les protocoles de liaison de données

- Le circuit de données : altérer les informations transportées
- Le protocole de liaison de données le supervise et définit un ensemble de règles pour assurer la fiabilité des échanges sur une liaison de données
- Le protocole spécifie le format des unités de données échangées (les trames), leur délimitation, les moyens de contrôler leur validité (parité, code polynomial...), ainsi que le mode de correction des erreurs détectées
- Il fixe également les règles du dialogue entre les deux extrémités de la liaison
- Il exerce en outre deux fonctions importantes : le contrôle de flux (mécanisme vérifiant le rythme d'envoi des informations) et la gestion des acquittements (mécanisme validant la réception des informations)

## Chapitre II

### Les protocoles de liaison de données

- Le circuit de données : altérer les informations transportées
- Le protocole de liaison de données le supervise et définit un ensemble de règles pour assurer la fiabilité des échanges sur une liaison de données
- Le protocole spécifie le format des unités de données échangées (les trames), leur délimitation, les moyens de contrôler leur validité (parité, code polynomial...), ainsi que le mode de correction des erreurs détectées
- Il fixe également les règles du dialogue entre les deux extrémités de la liaison
- Il exerce en outre deux fonctions importantes : le contrôle de flux (mécanisme vérifiant le rythme d'envoi des informations) et la gestion des acquittements (mécanisme validant la réception des informations)

## Chapitre II

### Les protocoles de liaison de données

HDLC : High  
level Data Link  
Control

C'est un exemple de protocole normalisé très répandu  
qui sera traité vers la fin de ce chapitre

## Chapitre II

### Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison

Un protocole est un ensemble de règles et de formats de données à respecter pour échanger des données dans de bonnes conditions entre deux équipements ou deux programmes. Un protocole de liaison de données a pour objet de rendre fiable le circuit de données

1

Le circuit de données transmet des éléments binaires



## Chapitre II

### Les protocoles de liaison de données

#### Rôle et fonctions d'un protocole de liaison

Un protocole est un ensemble de règles et de formats de données à respecter pour échanger des données dans de bonnes conditions entre deux équipements ou deux programmes. Un protocole de liaison de données a pour objet de rendre fiable le circuit de données

1 Le circuit de données transmet des éléments binaires

2 Le protocole de liaison de données travaille sur des blocs d'éléments binaires appelés trames

## Chapitre II

### Les protocoles de liaison de données

#### Rôle et fonctions d'un protocole de liaison

Un protocole est un ensemble de règles et de formats de données à respecter pour échanger des données dans de bonnes conditions entre deux équipements ou deux programmes. Un protocole de liaison de données a pour objet de rendre fiable le circuit de données

- 1 Le circuit de données transmet des éléments binaires
- 2 Le protocole de liaison de données travaille sur des blocs d'éléments binaires appelés trames
- 3 La trame est donc l'unité de données que gère le protocole :

## Chapitre II

### Les protocoles de liaison de données

#### Rôle et fonctions d'un protocole de liaison

Un protocole est un ensemble de règles et de formats de données à respecter pour échanger des données dans de bonnes conditions entre deux équipements ou deux programmes. Un protocole de liaison de données a pour objet de rendre fiable le circuit de données

- 1 Le circuit de données transmet des éléments binaires
- 2 Le protocole de liaison de données travaille sur des blocs d'éléments binaires appelés trames
- 3 La trame est donc l'unité de données que gère le protocole :

- Une trame compte différents champs

## Chapitre II

### Les protocoles de liaison de données

#### Rôle et fonctions d'un protocole de liaison

Un protocole est un ensemble de règles et de formats de données à respecter pour échanger des données dans de bonnes conditions entre deux équipements ou deux programmes. Un protocole de liaison de données a pour objet de rendre fiable le circuit de données

- 1 Le circuit de données transmet des éléments binaires
- 2 Le protocole de liaison de données travaille sur des blocs d'éléments binaires appelés trames
- 3 La trame est donc l'unité de données que gère le protocole :
  - Une trame compte différents champs
  - Chacun d'eux est un bloc d'éléments binaires dont la signification et l'interprétation sont précisées dans la définition du protocole

## Chapitre II

### Les protocoles de liaison de données

#### Rôle et fonctions d'un protocole de liaison

Un protocole est un ensemble de règles et de formats de données à respecter pour échanger des données dans de bonnes conditions entre deux équipements ou deux programmes. Un protocole de liaison de données a pour objet de rendre fiable le circuit de données

- 1 Le circuit de données transmet des éléments binaires
- 2 Le protocole de liaison de données travaille sur des blocs d'éléments binaires appelés trames
- 3 La trame est donc l'unité de données que gère le protocole :
  - Une trame compte différents champs
  - Chacun d'eux est un bloc d'éléments binaires dont la signification et l'interprétation sont précisées dans la définition du protocole

## Chapitre II

# Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison

Ainsi un protocole sert à préciser :

1

Le format des trames échangées

2

Les conditions de délimitation des trames (début et fin)  
et leur validité

## Chapitre II

# Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison

Ainsi un protocole sert à préciser :

- 1 Le format des trames échangées
- 2 Les conditions de délimitation des trames (début et fin) et leur validité
- 3 La position et la signification des différents champs d'une trame

## Chapitre II

# Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison

Ainsi un protocole sert à préciser :

- 1 Le format des trames échangées
- 2 Les conditions de délimitation des trames (début et fin) et leur validité
- 3 La position et la signification des différents champs d'une trame
- 4 La technique de détection d'erreur utilisée



## Chapitre II

# Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison

Ainsi un protocole sert à préciser :

- 1 Le format des trames échangées
- 2 Les conditions de délimitation des trames (début et fin) et leur validité
- 3 La position et la signification des différents champs d'une trame
- 4 La technique de détection d'erreur utilisée
- 5 Les règles du dialogue (supervision de la liaison)

## Chapitre II

# Les protocoles de liaison de données

### Rôle et fonctions d'un protocole de liaison

Ainsi un protocole sert à préciser :

- 1 Le format des trames échangées
- 2 Les conditions de délimitation des trames (début et fin) et leur validité
- 3 La position et la signification des différents champs d'une trame
- 4 La technique de détection d'erreur utilisée
- 5 Les règles du dialogue (supervision de la liaison)
- 6 Les procédures à respecter après détection d'erreurs ou de panne de la liaison

## Chapitre II

### Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison

Ainsi un protocole sert à préciser :

- 1 Le format des trames échangées
- 2 Les conditions de délimitation des trames (début et fin) et leur validité
- 3 La position et la signification des différents champs d'une trame
- 4 La technique de détection d'erreur utilisée
- 5 Les règles du dialogue (supervision de la liaison)
- 6 Les procédures à respecter après détection d'erreurs ou de panne de la liaison

## Chapitre II

### Les protocoles de liaison de données

#### Mise en forme des données

Rôle et  
fonctions d'un  
protocole de  
liaison



#### Délimitation par une séquence spécifique d'éléments binaires



Les trames ayant un nombre quelconque de bits, une séquence spécifique, appelée fanion (ou flag), sert à indiquer le début aussi bien que la fin des trames

## Chapitre II

### Les protocoles de liaison de données

#### Mise en forme des données

#### Rôle et fonctions d'un protocole de liaison

- Délimitation par une séquence spécifique d'éléments binaires
- Les trames ayant un nombre quelconque de bits, une séquence spécifique, appelée fanion (ou flag), sert à indiquer le début aussi bien que la fin des trames
- Le fanion se compose de l'octet 01111110

## Chapitre II

### Les protocoles de liaison de données

#### Mise en forme des données

#### Rôle et fonctions d'un protocole de liaison

- Délimitation par une séquence spécifique d'éléments binaires
  - Les trames ayant un nombre quelconque de bits, une séquence spécifique, appelée fanion (ou flag), sert à indiquer le début aussi bien que la fin des trames
  - Le fanion se compose de l'octet 01111110
  - Un mécanisme de transparence est nécessaire pour éviter de retrouver cette séquence à l'intérieur d'une trame

## Chapitre II

### Les protocoles de liaison de données

#### Mise en forme des données

#### Rôle et fonctions d'un protocole de liaison

- **Délimitation par une séquence spécifique d'éléments binaires**
  - Les trames ayant un nombre quelconque de bits, une séquence spécifique, appelée fanion (ou flag), sert à indiquer le début aussi bien que la fin des trames
  - Le fanion se compose de l'octet 01111110
  - Un mécanisme de transparence est nécessaire pour éviter de retrouver cette séquence à l'intérieur d'une trame
  - À l'émission, on insère dans le corps de la trame un élément binaire 0 après avoir rencontré cinq éléments binaires consécutifs de valeur 1

# Chapitre II

## Les protocoles de liaison de données

### Mise en forme des données

#### Rôle et fonctions d'un protocole de liaison

- **Délimitation par une séquence spécifique d'éléments binaires**
  - Les trames ayant un nombre quelconque de bits, une séquence spécifique, appelée fanion (ou flag), sert à indiquer le début aussi bien que la fin des trames
  - Le fanion se compose de l'octet 01111110
  - Un mécanisme de transparence est nécessaire pour éviter de retrouver cette séquence à l'intérieur d'une trame
  - À l'émission, on insère dans le corps de la trame un élément binaire 0 après avoir rencontré cinq éléments binaires consécutifs de valeur 1
  - En réception, il faut supprimer l'élément binaire de valeur 0 après avoir rencontré cinq éléments binaires consécutifs de valeur 1



# Chapitre II

## Les protocoles de liaison de données

### Mise en forme des données

#### Rôle et fonctions d'un protocole de liaison

- **Délimitation par une séquence spécifique d'éléments binaires**
  - Les trames ayant un nombre quelconque de bits, une séquence spécifique, appelée fanion (ou flag), sert à indiquer le début aussi bien que la fin des trames
  - Le fanion se compose de l'octet 01111110
  - Un mécanisme de transparence est nécessaire pour éviter de retrouver cette séquence à l'intérieur d'une trame
  - À l'émission, on insère dans le corps de la trame un élément binaire 0 après avoir rencontré cinq éléments binaires consécutifs de valeur 1
  - En réception, il faut supprimer l'élément binaire de valeur 0 après avoir rencontré cinq éléments binaires consécutifs de valeur 1

## Chapitre II

### Les protocoles de liaison de données

#### Mise en forme des données

Rôle et  
fonctions d'un  
protocole de  
liaison

#### Délimitation par une séquence spécifique d'éléments binaires

Un tel mécanisme (le bit stuffing) interdit l'émission de plus de cinq éléments binaires de valeur 1 dans le corps de la trame, puisque cette configuration est réservée à sa délimitation

## Chapitre II

### Les protocoles de liaison de données

#### Mise en forme des données

#### Délimitation par une séquence spécifique d'éléments binaires

- Un tel mécanisme (le bit stuffing) interdit l'émission de plus de cinq éléments binaires de valeur 1 dans le corps de la trame, puisque cette configuration est réservée à sa délimitation

#### Exemple

Prenons les données utiles suivantes :

0110 1111 1110 1001

## Chapitre II

### Les protocoles de liaison de données

#### Mise en forme des données

#### Délimitation par une séquence spécifique d'éléments binaires

- Un tel mécanisme (le bit stuffing) interdit l'émission de plus de cinq éléments binaires de valeur 1 dans le corps de la trame, puisque cette configuration est réservée à sa délimitation

#### Exemple

- Prenons les données utiles suivantes :  
0110 1111 1110 1001

Précédées et suivies de fanions, elles seront réellement émises sous la forme :

# Chapitre II

## Les protocoles de liaison de données

### Mise en forme des données

Rôle et  
fonctions d'un  
protocole de  
liaison

### Délimitation par une séquence spécifique d'éléments binaires

- Un tel mécanisme (le bit stuffing) interdit l'émission de plus de cinq éléments binaires de valeur 1 dans le corps de la trame, puisque cette configuration est réservée à sa délimitation

### Exemple

Prenons les données utiles suivantes :

- 0110 1111 1110 1001

Précédées et suivies de fanions, elles seront réellement émises sous la forme :

- 01111110 0110 1111 10110 1001 01111110

Dans cette séquence, les fanions sont soulignés et le bit inséré pour la transparence est en vert

# Chapitre II

## Les protocoles de liaison de données

### Mise en forme des données

Rôle et  
fonctions d'un  
protocole de  
liaison

### Délimitation par une séquence spécifique d'éléments binaires

- Un tel mécanisme (le bit stuffing) interdit l'émission de plus de cinq éléments binaires de valeur 1 dans le corps de la trame, puisque cette configuration est réservée à sa délimitation

### Exemple

- Prenons les données utiles suivantes :  
0110 1111 1110 1001
- Précédées et suivies de fanions, elles seront réellement émises sous la forme :  
01111110 0110 1111 10110 1001 01111110

Dans cette séquence, les fanions sont soulignés et le bit inséré pour la transparence est en vert

## Chapitre II

### Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison

#### Mise en forme des données

#### Délimitation par transmission de la longueur du champ de données

Consiste à indiquer, dans un champ particulier, le nombre d'octets utiles contenus dans la trame

## Chapitre II

### Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison

#### Mise en forme des données

#### Délimitation par transmission de la longueur du champ de données

- Consiste à indiquer, dans un champ particulier, le nombre d'octets utiles contenus dans la trame
- Après une séquence de début de trame, un ou plusieurs octets indiquent sa longueur (l'emplacement de ce champ est fixé par rapport au début de trame)



## Chapitre II

### Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison

#### Mise en forme des données

#### Délimitation par transmission de la longueur du champ de données

- Consiste à indiquer, dans un champ particulier, le nombre d'octets utiles contenus dans la trame
- Après une séquence de début de trame, un ou plusieurs octets indiquent sa longueur (l'emplacement de ce champ est fixé par rapport au début de trame)
- S'exprime généralement en octets ou en nombre de mots (de 16 ou 32 bits, par exemple)

## Chapitre II

### Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison

#### Mise en forme des données

#### Délimitation par transmission de la longueur du champ de données

- Consiste à indiquer, dans un champ particulier, le nombre d'octets utiles contenus dans la trame
- Après une séquence de début de trame, un ou plusieurs octets indiquent sa longueur (l'emplacement de ce champ est fixé par rapport au début de trame)
- S'exprime généralement en octets ou en nombre de mots (de 16 ou 32 bits, par exemple)
- On évite les problèmes de transparence puisque le récepteur n'interprète en aucun cas les données reçues comme des délimiteurs

## Chapitre II

### Les protocoles de liaison de données

#### Rôle et fonctions d'un protocole de liaison

#### Mise en forme des données

#### Délimitation par transmission de la longueur du champ de données

- Consiste à indiquer, dans un champ particulier, le nombre d'octets utiles contenus dans la trame
- Après une séquence de début de trame, un ou plusieurs octets indiquent sa longueur (l'emplacement de ce champ est fixé par rapport au début de trame)
- S'exprime généralement en octets ou en nombre de mots (de 16 ou 32 bits, par exemple)
- On évite les problèmes de transparence puisque le récepteur n'interprète en aucun cas les données reçues comme des délimiteurs

## Chapitre II

### Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison



#### Contrôle de la validité de l'information transmise

Le contrôle d'erreurs consiste à vérifier la validité des données transmises

Si on admet que le service de transmission n'est pas fiable, il faut se protéger contre d'éventuelles erreurs, donc les détecter puis les corriger

## Chapitre II

### Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison

#### Contrôle de la validité de l'information transmise

Le contrôle d'erreurs consiste à vérifier la validité des données transmises

Si on admet que le service de transmission n'est pas fiable, il faut se protéger contre d'éventuelles erreurs, donc les détecter puis les corriger

Protection au niveau du code

## Chapitre II

### Les protocoles de liaison de données

#### Rôle et fonctions d'un protocole de liaison

#### Contrôle de la validité de l'information transmise

Le contrôle d'erreurs consiste à vérifier la validité des données transmises

- Si on admet que le service de transmission n'est pas fiable, il faut se protéger contre d'éventuelles erreurs, donc les détecter puis les corriger
- Protection au niveau du code
- La protection au niveau du code consiste à organiser une redondance interne à celui-ci

## Chapitre II

### Les protocoles de liaison de données

#### Rôle et fonctions d'un protocole de liaison

#### Contrôle de la validité de l'information transmise

Le contrôle d'erreurs consiste à vérifier la validité des données transmises

- Si on admet que le service de transmission n'est pas fiable, il faut se protéger contre d'éventuelles erreurs, donc les détecter puis les corriger
- Protection au niveau du code
- La protection au niveau du code consiste à organiser une redondance interne à celui-ci
- Ce type de protection est possible lorsque l'émission des données se fait par caractère
- On introduit une redondance pour chaque caractère transmis

## Chapitre II

### Les protocoles de liaison de données

#### Rôle et fonctions d'un protocole de liaison

#### Contrôle de la validité de l'information transmise

Le contrôle d'erreurs consiste à vérifier la validité des données transmises

- Si on admet que le service de transmission n'est pas fiable, il faut se protéger contre d'éventuelles erreurs, donc les détecter puis les corriger
- Protection au niveau du code
- La protection au niveau du code consiste à organiser une redondance interne à celui-ci
- Ce type de protection est possible lorsque l'émission des données se fait par caractère
- On introduit une redondance pour chaque caractère transmis



## Chapitre II

### Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison



#### Contrôle de la validité de l'information transmise

On ajoute à chaque caractère un bit de parité dit parité verticale ou VRC (Vertical Redundancy Check) selon :

Pour chaque caractère, on somme modulo 2 ses bits

Si le nombre de bits 1 est pair (impair), on ajoute 0 (1)  
à la fin du caractère

## Chapitre II

### Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison



#### Contrôle de la validité de l'information transmise

On ajoute à chaque caractère un bit de parité dit parité verticale ou VRC (Vertical Redundancy Check) selon :

Pour chaque caractère, on somme modulo 2 ses bits

Si le nombre de bits 1 est pair (impair), on ajoute 0 (1)  
à la fin du caractère

## Chapitre II

### Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison

#### Contrôle de la validité de l'information transmise

##### Exemple

On ajoute à chaque caractère un bit de parité dit parité verticale ou VRC (Vertical Redundancy Check), calculé comme suit :

- Pour le caractère M codé par 1001101, le bit de parité vaut 0

## Chapitre II

### Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison

#### Contrôle de la validité de l'information transmise

##### Exemple

On ajoute à chaque caractère un bit de parité dit parité verticale ou VRC (Vertical Redundancy Check), calculé comme suit :

- Pour le caractère M codé par 1001101, le bit de parité vaut 0
- Construction : 10011010

## Chapitre II

### Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison

#### Contrôle de la validité de l'information transmise

##### Exemple

On ajoute à chaque caractère un bit de parité dit parité verticale ou VRC (Vertical Redundancy Check), calculé comme suit :

- Pour le caractère M codé par 1001101, le bit de parité vaut 0
- Construction : 10011010
- On transmet : 10110010

## Chapitre II

### Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison

#### Contrôle de la validité de l'information transmise

##### Exemple

On ajoute à chaque caractère un bit de parité dit parité verticale ou VRC (Vertical Redundancy Check), calculé comme suit :

- Pour le caractère M codé par 1001101, le bit de parité vaut 0
- Construction : 10011010
- On transmet : 10110010

## Chapitre II

### Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison



Contrôle de la validité de l'information transmise

Protection au niveau de la trame

1

Contrôle de parité longitudinale ou LRC (Longitudinal Redundancy Check) en plus du VRC

## Chapitre II

### Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison

#### Contrôle de la validité de l'information transmise

##### ■ Protection au niveau de la trame

- 1 Contrôle de parité longitudinale ou LRC (Longitudinal Redundancy Check) en plus du VRC
- 2 On procède de la même façon pour chaque caractère (VRC) puis la somme modulo 2 de tous les bits de même rang de tous les caractères à transmettre (LRC)



## Chapitre II

### Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison

#### Contrôle de la validité de l'information transmise

##### ■ Protection au niveau de la trame

- 1 Contrôle de parité longitudinale ou LRC (Longitudinal Redundancy Check) en plus du VRC
- 2 On procède de la même façon pour chaque caractère (VRC) puis la somme modulo 2 de tous les bits de même rang de tous les caractères à transmettre (LRC)

## Chapitre II

### Les protocoles de liaison de données

#### Contrôle de la validité de l'information transmise

Rôle et  
fonctions d'un  
protocole de  
liaison

#### Exemple :

- Pour les caractères L, 2 et M codés par 1001100, 0110010 et 1001101, les bits de parité valent resp 1, 1 et 0
- Construction : 10011001, 01100101 et 10011010

## Chapitre II

### Les protocoles de liaison de données

#### Contrôle de la validité de l'information transmise

Rôle et  
fonctions d'un  
protocole de  
liaison

#### Exemple :

- Pour les caractères L, 2 et M codé par 1001100, 0110010 et 1001101, les bits de parité valent resp 1, 1 et 0
- Construction : 10011001, 01100101 et 10011010
- Puis :

1001100 1 Caractère L + parité VRC  
 0110010 1 Caractère 2 + parité VRC  
1001101 0 Caractère M + parité VRC  
 0110011 0 LRC + parité VRC

Donnée à transmettre :

00110011 01001101 10110010 11001100

## Chapitre II

### Les protocoles de liaison de données

#### Contrôle de la validité de l'information transmise

Rôle et  
fonctions d'un  
protocole de  
liaison

#### Exemple :

- Pour les caractères L, 2 et M codés par 1001100, 0110010 et 1001101, les bits de parité valent resp 1, 1 et 0
- Construction : 10011001, 01100101 et 10011010
- Puis :

1001100 1 Caractère L + parité VRC  
 0110010 1 Caractère 2 + parité VRC  
1001101 0 Caractère M + parité VRC  
 0110011 0 LRC + parité VRC

Donnée à transmettre :

00110011 01001101 10110010 11001100

## Chapitre II

### Les protocoles de liaison de données

#### Contrôle de la validité de l'information transmise

Rôle et  
fonctions d'un  
protocole de  
liaison

- Protection au niveau de la trame
- Contrôle polynomial

## Chapitre II

### Les protocoles de liaison de données

#### Contrôle de la validité de l'information transmise

Rôle et  
fonctions d'un  
protocole de  
liaison

- Protection au niveau de la trame
- Contrôle polynomial
- Appelé couramment par abus de langage code cyclique ou CRC (Cyclic Redundancy Check)

## Chapitre II

### Les protocoles de liaison de données

#### Contrôle de la validité de l'information transmise

Rôle et  
fonctions d'un  
protocole de  
liaison

- Protection au niveau de la trame
  - Contrôle polynomial
  - Appelé couramment par abus de langage code cyclique ou CRC (Cyclic Redundancy Check)
  - Très utilisé dans les protocoles modernes car il permet de détecter les erreurs sur plusieurs bits

## Chapitre II

### Les protocoles de liaison de données

#### Contrôle de la validité de l'information transmise

Rôle et  
fonctions d'un  
protocole de  
liaison

- Protection au niveau de la trame
  - Contrôle polynomial
  - Appelé couramment par abus de langage code cyclique ou CRC (Cyclic Redundancy Check)
  - Très utilisé dans les protocoles modernes car il permet de détecter les erreurs sur plusieurs bits
  - Construction : Comment ?



## Chapitre II

### Les protocoles de liaison de données

#### Contrôle de la validité de l'information transmise

Rôle et  
fonctions d'un  
protocole de  
liaison

- Protection au niveau de la trame
  - Contrôle polynomial
  - Appelé couramment par abus de langage code cyclique ou CRC (Cyclic Redundancy Check)
  - Très utilisé dans les protocoles modernes car il permet de détecter les erreurs sur plusieurs bits
  - Construction : Comment ?
  - On considère la trame à transmettre comme un groupe de bits auquel on fait correspondre un polynôme  $P(x)$

# Chapitre II

## Les protocoles de liaison de données

### Contrôle de la validité de l'information transmise

Rôle et  
fonctions d'un  
protocole de  
liaison

- Protection au niveau de la trame
  - Contrôle polynomial
  - Appelé couramment par abus de langage code cyclique ou CRC (Cyclic Redundancy Check)
  - Très utilisé dans les protocoles modernes car il permet de détecter les erreurs sur plusieurs bits
  - Construction : Comment ?
  - On considère la trame à transmettre comme un groupe de bits auquel on fait correspondre un polynôme  $P(x)$
  - Les algorithmes de calcul se font modulo 2 sur les polynômes [par exemple :  $(x^7 + x^3) + (x^3 + x) = x^7 + x$ ]

# Chapitre II

## Les protocoles de liaison de données

### Contrôle de la validité de l'information transmise

#### Rôle et fonctions d'un protocole de liaison

- Protection au niveau de la trame
  - Contrôle polynomial
  - Appelé couramment par abus de langage code cyclique ou CRC (Cyclic Redundancy Check)
  - Très utilisé dans les protocoles modernes car il permet de détecter les erreurs sur plusieurs bits
  - Construction : Comment ?
  - On considère la trame à transmettre comme un groupe de bits auquel on fait correspondre un polynôme  $P(x)$
  - Les algorithmes de calcul se font modulo 2 sur les polynômes [par exemple :  $(x^7 + x^3) + (x^3 + x) = x^7 + x$ ]
  - On choisit un polynôme  $G(x)$  de degré  $r$ , appelé polynôme générateur, caractéristique du contrôle

# Chapitre II

## Les protocoles de liaison de données

### Contrôle de la validité de l'information transmise

#### Rôle et fonctions d'un protocole de liaison

- Protection au niveau de la trame
  - Contrôle polynomial
  - Appelé couramment par abus de langage code cyclique ou CRC (Cyclic Redundancy Check)
  - Très utilisé dans les protocoles modernes car il permet de détecter les erreurs sur plusieurs bits
  - Construction : Comment ?
  - On considère la trame à transmettre comme un groupe de bits auquel on fait correspondre un polynôme  $P(x)$
  - Les algorithmes de calcul se font modulo 2 sur les polynômes [par exemple :  $(x^7 + x^3) + (x^3 + x) = x^7 + x$ ]
  - On choisit un polynôme  $G(x)$  de degré  $r$ , appelé polynôme générateur, caractéristique du contrôle

## Chapitre II

### Les protocoles de liaison de données

#### Contrôle de la validité de l'information transmise

#### Rôle et fonctions d'un protocole de liaison

- Protection au niveau de la trame
  - Contrôle polynomial
  - Appelé couramment par abus de langage code cyclique ou CRC (Cyclic Redundancy Check)
  - Très utilisé dans les protocoles modernes car il permet de détecter les erreurs sur plusieurs bits
  - Construction : Comment ?
  - On considère la trame à transmettre comme un groupe de bits auquel on fait correspondre un polynôme  $P(x)$
  - Les algorithmes de calcul se font modulo 2 sur les polynômes [par exemple :  $(x^7 + x^3) + (x^3 + x) = x^7 + x$ ]
  - On choisit un polynôme  $G(x)$  de degré  $r$ , appelé polynôme générateur, caractéristique du contrôle

## Chapitre II

### Les protocoles de liaison de données

#### Contrôle de la validité de l'information transmise

Rôle et  
fonctions d'un  
protocole de  
liaison

- À l'émission, on multiplie  $P(x)$  par  $x^r$  et on divise le polynôme obtenu par  $G(x)$
- Le reste noté  $R(x)$ , obtenu par division euclidienne, est de degré strictement inférieur à  $r$  et est ajouté à la fin de la trame comme code de contrôle

## Chapitre II

### Les protocoles de liaison de données

#### Contrôle de la validité de l'information transmise

Rôle et  
fonctions d'un  
protocole de  
liaison

- À l'émission, on multiplie  $P(x)$  par  $x^r$  et on divise le polynôme obtenu par  $G(x)$
- Le reste noté  $R(x)$ , obtenu par division euclidienne, est de degré strictement inférieur à  $r$  et est ajouté à la fin de la trame comme code de contrôle

$$x^r * P(x) = G(x) * Q(x) + R(x) \quad (1)$$

## Chapitre II

### Les protocoles de liaison de données

#### Contrôle de la validité de l'information transmise

Rôle et  
fonctions d'un  
protocole de  
liaison

- À l'émission, on multiplie  $P(x)$  par  $x^r$  et on divise le polynôme obtenu par  $G(x)$
- Le reste noté  $R(x)$ , obtenu par division euclidienne, est de degré strictement inférieur à  $r$  et est ajouté à la fin de la trame comme code de contrôle

$$x^r * P(x) = G(x) * Q(x) + R(x) \quad (1)$$

$$\text{On transmet : } T(x) = x^r * P(x) + R(x) \quad (2)$$



## Chapitre II

### Les protocoles de liaison de données

#### Contrôle de la validité de l'information transmise

Rôle et  
fonctions d'un  
protocole de  
liaison

- À l'émission, on multiplie  $P(x)$  par  $x^r$  et on divise le polynôme obtenu par  $G(x)$
- Le reste noté  $R(x)$ , obtenu par division euclidienne, est de degré strictement inférieur à  $r$  et est ajouté à la fin de la trame comme code de contrôle

$$x^r * P(x) = G(x) * Q(x) + R(x) \quad (1)$$

$$\text{On transmet : } T(x) = x^r * P(x) + R(x) \quad (2)$$

D'après (1) et (2) puis modulo 2, ce polynôme vérifie

## Chapitre II

### Les protocoles de liaison de données

#### Contrôle de la validité de l'information transmise

Rôle et  
fonctions d'un  
protocole de  
liaison

- À l'émission, on multiplie  $P(x)$  par  $x^r$  et on divise le polynôme obtenu par  $G(x)$
- Le reste noté  $R(x)$ , obtenu par division euclidienne, est de degré strictement inférieur à  $r$  et est ajouté à la fin de la trame comme code de contrôle

$$x^r * P(x) = G(x) * Q(x) + R(x) \quad (1)$$

$$\text{On transmet : } T(x) = x^r * P(x) + R(x) \quad (2)$$

D'après (1) et (2) puis modulo 2, ce polynôme vérifie  
 $T(x) = G(x) * Q(x)$  Il est donc divisible par  $G(x)$

## Chapitre II

### Les protocoles de liaison de données

#### Contrôle de la validité de l'information transmise

Rôle et  
fonctions d'un  
protocole de  
liaison

- À l'émission, on multiplie  $P(x)$  par  $x^r$  et on divise le polynôme obtenu par  $G(x)$
- Le reste noté  $R(x)$ , obtenu par division euclidienne, est de degré strictement inférieur à  $r$  et est ajouté à la fin de la trame comme code de contrôle

$$x^r * P(x) = G(x) * Q(x) + R(x) \quad (1)$$

$$\text{On transmet : } T(x) = x^r * P(x) + R(x) \quad (2)$$

D'après (1) et (2) puis modulo 2, ce polynôme vérifie  $T(x) = G(x) * Q(x)$  Il est donc divisible par  $G(x)$

## Chapitre II

### Les protocoles de liaison de données

#### Contrôle de la validité de l'information transmise

Rôle et  
fonctions d'un  
protocole de  
liaison

#### Exemple

- À l'information 1000001110000100 est associée  
 $P(x) = x^{15} + x^9 + x^8 + x^7 + x^2$
- Soit le polynôme générateur de degré 12 :  
 $G(x) = x^{12} + x^{11} + x^3 + x^2 + x + 1$

## Chapitre II

### Les protocoles de liaison de données

#### Contrôle de la validité de l'information transmise

Rôle et  
fonctions d'un  
protocole de  
liaison

#### Exemple

- À l'information 1000001110000100 est associée  

$$P(x) = x^{15} + x^9 + x^8 + x^7 + x^2$$
- Soit le polynôme générateur de degré 12 :  

$$G(x) = x^{12} + x^{11} + x^3 + x^2 + x + 1$$
- La division de  $x^{12} * P(x)$  par  $G(x)$  donne :  

$$R(x) = x^{11} + x^9 + x^8 + x^7 + x^6 + x^4 + 1$$

# Chapitre II

## Les protocoles de liaison de données

### Contrôle de la validité de l'information transmise

Rôle et  
fonctions d'un  
protocole de  
liaison

#### Exemple

- À l'information 1000001110000100 est associée  

$$P(x) = x^{15} + x^9 + x^8 + x^7 + x^2$$
- Soit le polynôme générateur de degré 12 :  

$$G(x) = x^{12} + x^{11} + x^3 + x^2 + x + 1$$
- La division de  $x^{12} * P(x)$  par  $G(x)$  donne :  

$$R(x) = x^{11} + x^9 + x^8 + x^7 + x^6 + x^4 + 1$$

$$\begin{array}{rcl}
 x^{12} * P(x) & = & 1000001110000100 \quad 000000000000 \\
 R(x) & = & \phantom{1000001110000100} \quad 101111010001 \\
 T(x) & = & 1000001110000100 \quad 101111010001
 \end{array}$$

# Chapitre II

## Les protocoles de liaison de données

### Contrôle de la validité de l'information transmise

Rôle et  
fonctions d'un  
protocole de  
liaison

#### Exemple

- À l'information 1000001110000100 est associée  
 $P(x) = x^{15} + x^9 + x^8 + x^7 + x^2$
- Soit le polynôme générateur de degré 12 :  
 $G(x) = x^{12} + x^{11} + x^3 + x^2 + x + 1$
- La division de  $x^{12} * P(x)$  par  $G(x)$  donne :  
 $R(x) = x^{11} + x^9 + x^8 + x^7 + x^6 + x^4 + 1$

$$\begin{array}{rcl}
 x^{12} * P(x) & = & 1000001110000100 \quad 000000000000 \\
 R(x) & = & \quad \quad \quad 101111010001 \\
 T(x) & = & 1000001110000100 \quad 101111010001
 \end{array}$$

# Chapitre II

## Les protocoles de liaison de données

### Contrôle de la validité de l'information transmise

Rôle et  
fonctions d'un  
protocole de  
liaison

#### Exemple

- À l'information 1000001110000100 est associée  
 $P(x) = x^{15} + x^9 + x^8 + x^7 + x^2$
- Soit le polynôme générateur de degré 12 :  
 $G(x) = x^{12} + x^{11} + x^3 + x^2 + x + 1$
- La division de  $x^{12} * P(x)$  par  $G(x)$  donne :  
 $R(x) = x^{11} + x^9 + x^8 + x^7 + x^6 + x^4 + 1$

$$\begin{array}{rcl}
 x^{12} * P(x) & = & 1000001110000100 \quad 000000000000 \\
 R(x) & = & \quad \quad \quad 101111010001 \\
 T(x) & = & 1000001110000100 \quad 101111010001
 \end{array}$$



## Chapitre II

# Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison

Contrôle de la validité de l'information transmise

### Exemple

- À la réception, on vérifie que le reste de la division Euclidienne par  $G(x)$  est nul
- Division de quoi ?

## Chapitre II

### Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison

Contrôle de la validité de l'information transmise

#### Exemple

- À la réception, on vérifie que le reste de la division Euclidienne par  $G(x)$  est nul
- Division de quoi ?
- Réponse :

## Chapitre II

### Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison

Contrôle de la validité de l'information transmise

#### Exemple

- À la réception, on vérifie que le reste de la division Euclidienne par  $G(x)$  est nul
- Division de quoi ?
- Réponse :
- Division de  $T'(x)$  (reçu) par  $G(x)$

## Chapitre II

### Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison

Contrôle de la validité de l'information transmise

#### Exemple

- À la réception, on vérifie que le reste de la division Euclidienne par  $G(x)$  est nul
- Division de quoi ?
- Réponse :
- Division de  $T'(x)$  (reçu) par  $G(x)$

## Chapitre II

### Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison

Contrôle de la validité de l'information transmise

#### Exemple

- À la réception, on vérifie que le reste de la division Euclidienne par  $G(x)$  est nul
- Division de quoi ?
- Réponse :
- Division de  $T'(x)$  (reçu) par  $G(x)$

# Chapitre II

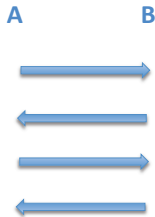
## Les protocoles de liaison de données

Rôle et  
fonctions d'un  
protocole de  
liaison

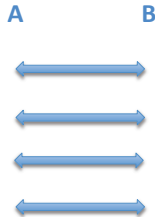
### Modes d'exploitation d'une liaison de données



**Simplex**



**Semi-Duplex  
Half-Duplex**



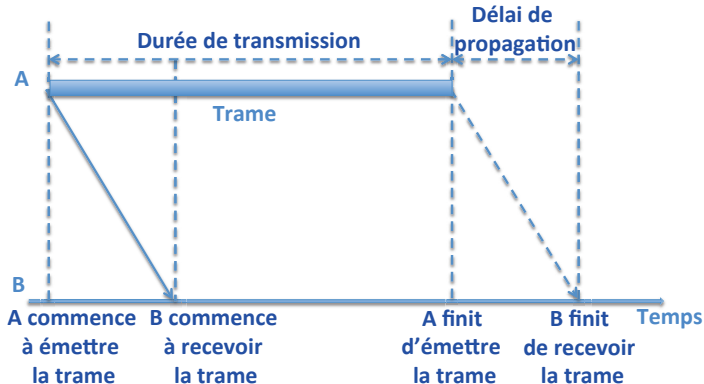
**Duplex Intégral  
Full-Duplex**

# Chapitre II

## Les protocoles de liaison de données

Fonctionnalités  
d'un protocole  
de liaison

Représentation des échanges de données



## Chapitre II

### Les protocoles de liaison de données

#### Contrôle de flux

#### Fonctionnalités d'un protocole de liaison

- Considérons 2 équipements A et B reliés par un circuit de données sur lequel A veut envoyer des données à B
- A découpe les données en trames, appelées trames d'information, et les transmet les unes à la suite des autres



## Chapitre II

### Les protocoles de liaison de données

#### Contrôle de flux

#### Fonctionnalités d'un protocole de liaison

- Considérons 2 équipements A et B reliés par un circuit de données sur lequel A veut envoyer des données à B
- A découpe les données en trames, appelées trames d'information, et les transmet les unes à la suite des autres
- Elles sont repérées par la lettre I sur la figure ci-après
- On suppose que toutes les données sont délivrées sans erreur à l'équipement B qui les stocke pour les exploiter

## Chapitre II

### Les protocoles de liaison de données

#### Contrôle de flux

#### Fonctionnalités d'un protocole de liaison

- Considérons 2 équipements A et B reliés par un circuit de données sur lequel A veut envoyer des données à B
- A découpe les données en trames, appelées trames d'information, et les transmet les unes à la suite des autres
- Elles sont repérées par la lettre I sur la figure ci-après
- On suppose que toutes les données sont délivrées sans erreur à l'équipement B qui les stocke pour les exploiter
- Supposons que A soit un ordinateur et B une imprimante lente, dotée d'une capacité mémoire limitée

## Chapitre II

### Les protocoles de liaison de données

#### Contrôle de flux

#### Fonctionnalités d'un protocole de liaison

- Considérons 2 équipements A et B reliés par un circuit de données sur lequel A veut envoyer des données à B
- A découpe les données en trames, appelées trames d'information, et les transmet les unes à la suite des autres
- Elles sont repérées par la lettre I sur la figure ci-après
- On suppose que toutes les données sont délivrées sans erreur à l'équipement B qui les stocke pour les exploiter
- Supposons que A soit un ordinateur et B une imprimante lente, dotée d'une capacité mémoire limitée
- A impose à B de garder en mémoire toutes les informations envoyées tant qu'elles ne sont pas imprimées

## Chapitre II

### Les protocoles de liaison de données

#### Contrôle de flux

#### Fonctionnalités d'un protocole de liaison

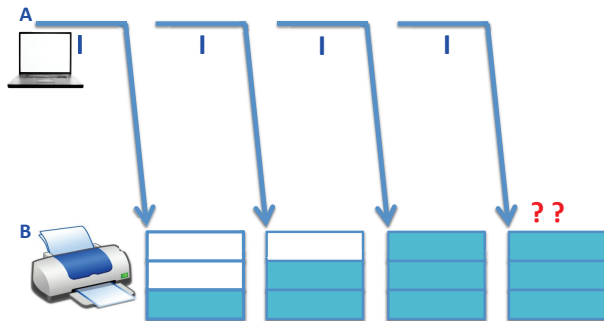
- Considérons 2 équipements A et B reliés par un circuit de données sur lequel A veut envoyer des données à B
- A découpe les données en trames, appelées trames d'information, et les transmet les unes à la suite des autres
- Elles sont repérées par la lettre I sur la figure ci-après
- On suppose que toutes les données sont délivrées sans erreur à l'équipement B qui les stocke pour les exploiter
- Supposons que A soit un ordinateur et B une imprimante lente, dotée d'une capacité mémoire limitée
- A impose à B de garder en mémoire toutes les informations envoyées tant qu'elles ne sont pas imprimées

# Chapitre II

## Les protocoles de liaison de données

### Contrôle de flux

Fonctionnalités  
d'un protocole  
de liaison



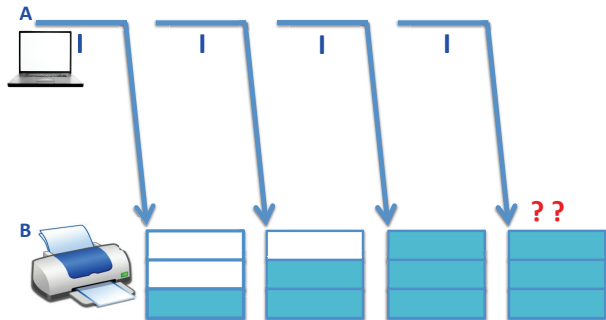
- Si Rythme d'envoi des informations  $>$  Rythme d'impression

# Chapitre II

## Les protocoles de liaison de données

### Contrôle de flux

### Fonctionnalités d'un protocole de liaison



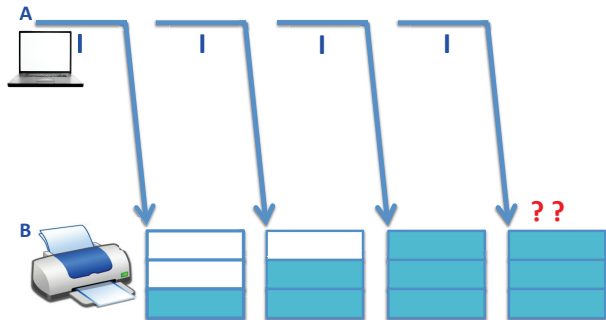
- Si Rythme d'envoi des informations  $>$  Rythme d'impression
- Saturation de la mémoire et perte d'informations par B

# Chapitre II

## Les protocoles de liaison de données

### Contrôle de flux

### Fonctionnalités d'un protocole de liaison



- Si Rythme d'envoi des informations  $>$  Rythme d'impression
- Saturation de la mémoire et perte d'informations par B





## Chapitre II

# Les protocoles de liaison de données

### Fonctionnalités d'un protocole de liaison

#### Contrôle de flux

- Pour réaliser le contrôle de flux, on introduit deux trames de supervision
  - 1 RR (Receiver Ready)
  - 2 RNR (Receiver Not Ready)

## Chapitre II

# Les protocoles de liaison de données

### Fonctionnalités d'un protocole de liaison

#### Contrôle de flux

- Pour réaliser le contrôle de flux, on introduit deux trames de supervision
  - 1 RR (Receiver Ready)
  - 2 RNR (Receiver Not Ready)
- Ces trames ne transportent aucune information utile et ne servent qu'à la gestion du dialogue

## Chapitre II

# Les protocoles de liaison de données

### Fonctionnalités d'un protocole de liaison

#### Contrôle de flux

- Pour réaliser le contrôle de flux, on introduit deux trames de supervision
  - 1 RR (Receiver Ready)
  - 2 RNR (Receiver Not Ready)
- Ces trames ne transportent aucune information utile et ne servent qu'à la gestion du dialogue
- Elles sont générées et exploitées par le protocole de liaison et sont invisibles pour l'utilisateur

## Chapitre II

# Les protocoles de liaison de données

### Fonctionnalités d'un protocole de liaison

#### Contrôle de flux

- Pour réaliser le contrôle de flux, on introduit deux trames de supervision
  - 1 RR (Receiver Ready)
  - 2 RNR (Receiver Not Ready)
- Ces trames ne transportent aucune information utile et ne servent qu'à la gestion du dialogue
- Elles sont générées et exploitées par le protocole de liaison et sont invisibles pour l'utilisateur

## Chapitre II

# Les protocoles de liaison de données

### Fonctionnalités d'un protocole de liaison

#### Contrôle de flux

Le mécanisme est le suivant :

- À chaque réception de trame, B envoie une trame RR s'il est prêt à accepter d'autres trames

## Chapitre II

# Les protocoles de liaison de données

### Fonctionnalités d'un protocole de liaison

#### Contrôle de flux

Le mécanisme est le suivant :

- À chaque réception de trame, B envoie une trame RR s'il est prêt à accepter d'autres trames
- Ou une trame RNR s'il ne veut plus en recevoir

## Chapitre II

### Les protocoles de liaison de données

#### Fonctionnalités d'un protocole de liaison

##### Contrôle de flux

Le mécanisme est le suivant :

- À chaque réception de trame, B envoie une trame RR s'il est prêt à accepter d'autres trames
- Ou une trame RNR s'il ne veut plus en recevoir
- Dans ce dernier cas, B envoie RR dès qu'il est prêt à accepter de nouvelles trames

## Chapitre II

### Les protocoles de liaison de données

#### Fonctionnalités d'un protocole de liaison

##### Contrôle de flux

Le mécanisme est le suivant :

- À chaque réception de trame, B envoie une trame RR s'il est prêt à accepter d'autres trames
- Ou une trame RNR s'il ne veut plus en recevoir
- Dans ce dernier cas, B envoie RR dès qu'il est prêt à accepter de nouvelles trames

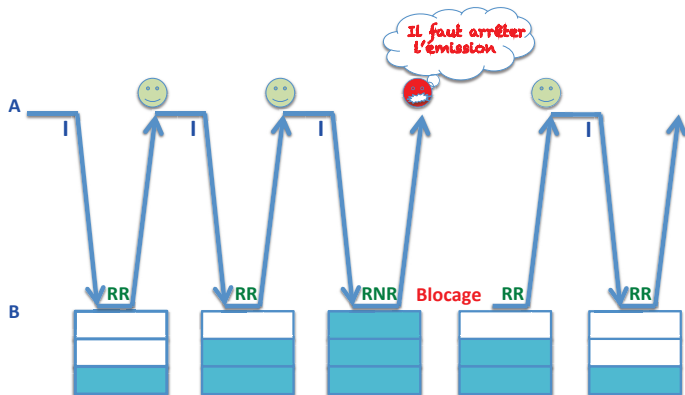


# Chapitre II

## Les protocoles de liaison de données

Fonctionnalités  
d'un protocole  
de liaison

Contrôle de flux



## Chapitre II

### Les protocoles de liaison de données

#### Gestion des acquittements

#### Fonctionnalités d'un protocole de liaison

- Supposons maintenant que le circuit ne soit pas totalement fiable et introduise des erreurs
- Au mécanisme de contrôle de flux décrit précédemment, il faut ajouter un processus d'acquittement des trames d'information reçues

## Chapitre II

### Les protocoles de liaison de données

#### Gestion des acquittements

#### Fonctionnalités d'un protocole de liaison

- Supposons maintenant que le circuit ne soit pas totalement fiable et introduise des erreurs
- Au mécanisme de contrôle de flux décrit précédemment, il faut ajouter un processus d'acquittement des trames d'information reçues
- 1 ■ Lorsque B reçoit correctement une trame, il envoie une trame d'acquittement et ne fait rien en cas de mauvaise réception

## Chapitre II

### Les protocoles de liaison de données

#### Gestion des acquittements

#### Fonctionnalités d'un protocole de liaison

- Supposons maintenant que le circuit ne soit pas totalement fiable et introduise des erreurs
- Au mécanisme de contrôle de flux décrit précédemment, il faut ajouter un processus d'acquiescement des trames d'information reçues
  - 1 Lorsque B reçoit correctement une trame, il envoie une trame d'acquiescement et ne fait rien en cas de mauvaise réception
  - 2 Lorsqu'une trame est mal reçue, B envoie une demande de retransmission à l'émetteur et ne fait rien en cas de bonne réception

Attention

Dans la deuxième stratégie l'absence de réponse est considérée comme un acquiescement

## Chapitre II

### Les protocoles de liaison de données

#### Gestion des acquittements

#### Fonctionnalités d'un protocole de liaison

- Supposons maintenant que le circuit ne soit pas totalement fiable et introduise des erreurs
- Au mécanisme de contrôle de flux décrit précédemment, il faut ajouter un processus d'acquiescement des trames d'information reçues
  - 1 Lorsque B reçoit correctement une trame, il envoie une trame d'acquiescement et ne fait rien en cas de mauvaise réception
  - 2 Lorsqu'une trame est mal reçue, B envoie une demande de retransmission à l'émetteur et ne fait rien en cas de bonne réception

Attention

Dans la deuxième stratégie l'absence de réponse est considérée comme un acquiescement

## Chapitre II

### Les protocoles de liaison de données

#### Gestion des acquittements

Le principe de gestion des trames est donc le suivant :

- À l'émission de chaque trame d'information, A arme un temporisateur (attente maximale de l'acquittement de B) qui sera désarmé à la réception de l'acquittement
- À l'échéance de cette temporisation, si B n'a pas répondu, A réémet la trame et réitère le processus

Fonctionnalités  
d'un protocole  
de liaison

## Chapitre II

### Les protocoles de liaison de données

#### Gestion des acquittements

Le principe de gestion des trames est donc le suivant :

- À l'émission de chaque trame d'information, A arme un temporisateur (attente maximale de l'acquittement de B) qui sera désarmé à la réception de l'acquittement
- À l'échéance de cette temporisation, si B n'a pas répondu, A réémet la trame et réitère le processus
- Le nombre de répétitions autorisées est limité : au-delà d'un certain seuil, on considère qu'un incident grave s'est produit (rupture totale de liaison, panne de l'équipement B, panne de l'équipement A...etc)

Fonctionnalités  
d'un protocole  
de liaison

## Chapitre II

### Les protocoles de liaison de données

#### Gestion des acquittements

Le principe de gestion des trames est donc le suivant :

- À l'émission de chaque trame d'information, A arme un temporisateur (attente maximale de l'acquittement de B) qui sera désarmé à la réception de l'acquittement
- À l'échéance de cette temporisation, si B n'a pas répondu, A réémet la trame et réitère le processus
- Le nombre de répétitions autorisées est limité : au-delà d'un certain seuil, on considère qu'un incident grave s'est produit (rupture totale de liaison, panne de l'équipement B, panne de l'équipement A...etc
- Avertir l'utilisateur que la liaison de données est rompue

Fonctionnalités  
d'un protocole  
de liaison



## Chapitre II

### Les protocoles de liaison de données

#### Gestion des acquittements

Le principe de gestion des trames est donc le suivant :

- À l'émission de chaque trame d'information, A arme un temporisateur (attente maximale de l'acquittement de B) qui sera désarmé à la réception de l'acquittement
- À l'échéance de cette temporisation, si B n'a pas répondu, A réémet la trame et réitère le processus
- Le nombre de répétitions autorisées est limité : au-delà d'un certain seuil, on considère qu'un incident grave s'est produit (rupture totale de liaison, panne de l'équipement B, panne de l'équipement A...etc
- Avertir l'utilisateur que la liaison de données est rompue

#### Problème

Dans ce cas, le circuit de transmission peut corrompre aussi bien les trames émises par A que celles émises par B

## Chapitre II

### Les protocoles de liaison de données

#### Gestion des acquittements

Le principe de gestion des trames est donc le suivant :

- À l'émission de chaque trame d'information, A arme un temporisateur (attente maximale de l'acquittement de B) qui sera désarmé à la réception de l'acquittement
- À l'échéance de cette temporisation, si B n'a pas répondu, A réémet la trame et réitère le processus
- Le nombre de répétitions autorisées est limité : au-delà d'un certain seuil, on considère qu'un incident grave s'est produit (rupture totale de liaison, panne de l'équipement B, panne de l'équipement A...etc
- Avertir l'utilisateur que la liaison de données est rompue

#### Problème

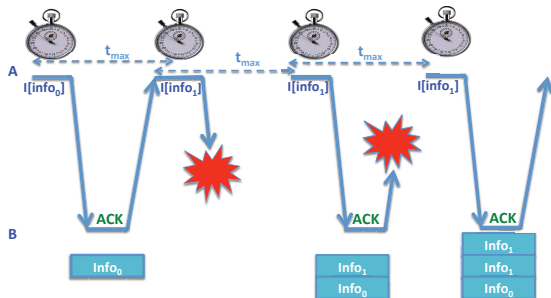
Dans ce cas, le circuit de transmission peut corrompre aussi bien les trames émises par A que celles émises par B

# Chapitre II

## Les protocoles de liaison de données

### Gestion des acquittements

### Fonctionnalités d'un protocole de liaison



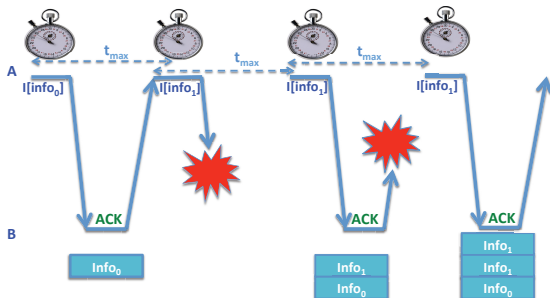
- A envoie vers B une trame contenant l'information 1
- B répond bien reçue par un acquittement ACK qui est mal transmis

# Chapitre II

## Les protocoles de liaison de données

### Gestion des acquittements

### Fonctionnalités d'un protocole de liaison



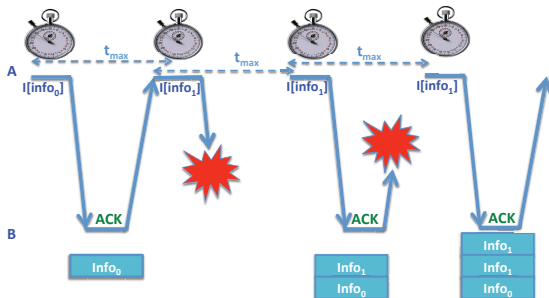
- A envoie vers B une trame contenant l'information 1
- B répond bien reçue par un acquittement ACK qui est mal transmis
- A réémet la même trame qui sera dupliquée dans B

# Chapitre II

## Les protocoles de liaison de données

### Gestion des acquittements

### Fonctionnalités d'un protocole de liaison



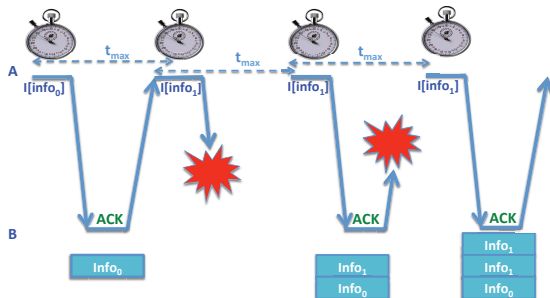
- A envoie vers B une trame contenant l'information 1
- B répond bien reçue par un acquittement ACK qui est mal transmis
- A réémet la même trame qui sera dupliquée dans B
- B n'a aucun moyen de détecter la duplication, puisqu'en aucun cas il n'analyse le contenu de la trame

# Chapitre II

## Les protocoles de liaison de données

### Gestion des acquittements

### Fonctionnalités d'un protocole de liaison



- A envoie vers B une trame contenant l'information 1
- B répond bien reçue par un acquittement ACK qui est mal transmis
- A réémet la même trame qui sera dupliquée dans B
- B n'a aucun moyen de détecter la duplication, puisqu'en aucun cas il n'analyse le contenu de la trame

## Chapitre II

# Les protocoles de liaison de données

### Fonctionnalités d'un protocole de liaison

#### Gestion des acquittements

- Puisque le protocole de liaison de données doit être complètement indépendant du contenu des trames transférées
- Il faut introduire un mécanisme supplémentaire qui distingue deux trames successives différentes

## Chapitre II

# Les protocoles de liaison de données

### Fonctionnalités d'un protocole de liaison

#### Gestion des acquittements

- Puisque le protocole de liaison de données doit être complètement indépendant du contenu des trames transférées
- Il faut introduire un mécanisme supplémentaire qui distingue deux trames successives différentes
- L'introduction d'un champ supplémentaire de numérotation ou d'indication de retransmission répond à ce besoin



## Chapitre II

# Les protocoles de liaison de données

### Fonctionnalités d'un protocole de liaison

#### Gestion des acquittements

- Puisque le protocole de liaison de données doit être complètement indépendant du contenu des trames transférées
- Il faut introduire un mécanisme supplémentaire qui distingue deux trames successives différentes
- L'introduction d'un champ supplémentaire de numérotation ou d'indication de retransmission répond à ce besoin

## Chapitre II

# Les protocoles de liaison de données

### Fonctionnalités d'un protocole de liaison

#### Numérotation des trames d'information

- Le protocole numérote chaque trame d'information
- La numérotation est placée dans l'en-tête

## Chapitre II

### Les protocoles de liaison de données

#### Fonctionnalités d'un protocole de liaison

#### Numérotation des trames d'information

- Le protocole numérote chaque trame d'information
- La numérotation est placée dans l'en-tête
- Deux trames possédant des numéros différents :

Transportent des unités de données distinctes

## Chapitre II

### Les protocoles de liaison de données

#### Fonctionnalités d'un protocole de liaison

#### Numérotation des trames d'information

- Le protocole numérote chaque trame d'information
- La numérotation est placée dans l'en-tête
- Deux trames possédant des numéros différents :

Transportent des unités de données distinctes

## Chapitre II

### Les protocoles de liaison de données

#### Numérotation des trames d'information

#### Fonctionnalités d'un protocole de liaison

- On appelle  $N(S)$  [S pour send] la variable donnant le numéro de la trame
- Cette variable, codée sur quelques bits, est prise modulo M un entier qui peut prendre les valeurs : 2 (deux différentes trames successives), 8 ou 128

## Chapitre II

### Les protocoles de liaison de données

#### Numérotation des trames d'information

#### Fonctionnalités d'un protocole de liaison

- On appelle  $N(S)$  [S pour send] la variable donnant le numéro de la trame
- Cette variable, codée sur quelques bits, est prise modulo M un entier qui peut prendre les valeurs : 2 (deux différentes trames successives), 8 ou 128
- L'introduction de numéros dans les trames impose des compteurs dans chaque station

## Chapitre II

### Les protocoles de liaison de données

#### Numérotation des trames d'information

#### Fonctionnalités d'un protocole de liaison

- On appelle  $N(S)$  [S pour send] la variable donnant le numéro de la trame
- Cette variable, codée sur quelques bits, est prise modulo M un entier qui peut prendre les valeurs : 2 (deux différentes trames successives), 8 ou 128
- L'introduction de numéros dans les trames impose des compteurs dans chaque station
- Le processus fonctionne de la façon suivante :

## Chapitre II

### Les protocoles de liaison de données

#### Numérotation des trames d'information

#### Fonctionnalités d'un protocole de liaison

- On appelle  $N(S)$  [ $S$  pour send] la variable donnant le numéro de la trame
- Cette variable, codée sur quelques bits, est prise modulo  $M$  un entier qui peut prendre les valeurs : 2 (deux différentes trames successives), 8 ou 128
- L'introduction de numéros dans les trames impose des compteurs dans chaque station
- Le processus fonctionne de la façon suivante :
  - 1 A possède un compteur interne  $V(S)$  donnant le numéro de la prochaine trame à émettre



## Chapitre II

### Les protocoles de liaison de données

#### Numérotation des trames d'information

#### Fonctionnalités d'un protocole de liaison

- On appelle  $N(S)$  [ $S$  pour send] la variable donnant le numéro de la trame
- Cette variable, codée sur quelques bits, est prise modulo  $M$  un entier qui peut prendre les valeurs : 2 (deux différentes trames successives), 8 ou 128
- L'introduction de numéros dans les trames impose des compteurs dans chaque station
- Le processus fonctionne de la façon suivante :
  - 1 A possède un compteur interne  $V(S)$  donnant le numéro de la prochaine trame à émettre
  - 2 A émet cette trame en copiant  $V(S)$  dans le champ  $N(S)$ , puis il incrémente  $V(S)$

# Chapitre II

## Les protocoles de liaison de données

### Numérotation des trames d'information

#### Fonctionnalités d'un protocole de liaison

- On appelle  $N(S)$  [ $S$  pour send] la variable donnant le numéro de la trame
- Cette variable, codée sur quelques bits, est prise modulo  $M$  un entier qui peut prendre les valeurs : 2 (deux différentes trames successives), 8 ou 128
- L'introduction de numéros dans les trames impose des compteurs dans chaque station
- Le processus fonctionne de la façon suivante :
  - 1 A possède un compteur interne  $V(S)$  donnant le numéro de la prochaine trame à émettre
  - 2 A émet cette trame en copiant  $V(S)$  dans le champ  $N(S)$ , puis il incrémente  $V(S)$
  - 3 Pour toute répétition d'une trame d'information, A émet la trame sans modifier son numéro d'ordre

# Chapitre II

## Les protocoles de liaison de données

### Numérotation des trames d'information

#### Fonctionnalités d'un protocole de liaison

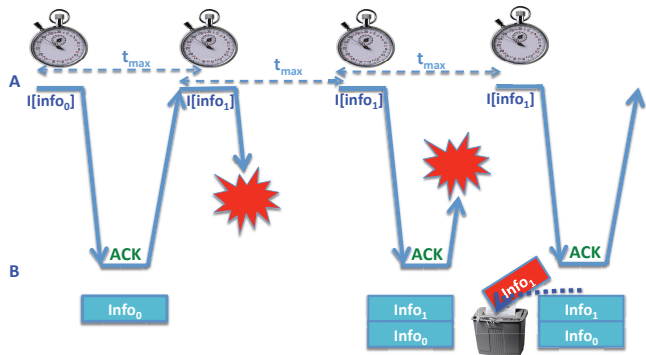
- On appelle  $N(S)$  [ $S$  pour send] la variable donnant le numéro de la trame
- Cette variable, codée sur quelques bits, est prise modulo  $M$  un entier qui peut prendre les valeurs : 2 (deux différentes trames successives), 8 ou 128
- L'introduction de numéros dans les trames impose des compteurs dans chaque station
- Le processus fonctionne de la façon suivante :
  - 1 A possède un compteur interne  $V(S)$  donnant le numéro de la prochaine trame à émettre
  - 2 A émet cette trame en copiant  $V(S)$  dans le champ  $N(S)$ , puis il incrémente  $V(S)$
  - 3 Pour toute répétition d'une trame d'information, A émet la trame sans modifier son numéro d'ordre

# Chapitre II

## Les protocoles de liaison de données

Fonctionnalités  
d'un protocole  
de liaison

### Numérotation des trames d'information



## Chapitre II

### Les protocoles de liaison de données

#### Numérotation des trames d'information

#### Fonctionnalités d'un protocole de liaison

#### Problème

Considérons le scénario où A envoie une trame  $I_0$  bien reçue par B mais dont l'acquittement lui arrive après expiration de la temporisation associée

- A émet à nouveau la trame  $I_0$
- Lorsqu'il reçoit l'acquittement de la première trame  $I_0$  émise, il considère qu'il s'agit de l'acquittement de la seconde et il émet la trame  $I_1$

## Chapitre II

### Les protocoles de liaison de données

#### Numérotation des trames d'information

#### Fonctionnalités d'un protocole de liaison

#### Problème

Considérons le scénario où A envoie une trame  $I_0$  bien reçue par B mais dont l'acquittement lui arrive après expiration de la temporisation associée

- A émet à nouveau la trame  $I_0$
- Lorsqu'il reçoit l'acquittement de la première trame  $I_0$  émise, il considère qu'il s'agit de l'acquittement de la seconde et il émet la trame  $I_1$
- A interprète l'acquittement qui suit comme l'acquittement de sa trame  $I_1$

## Chapitre II

### Les protocoles de liaison de données

#### Numérotation des trames d'information

#### Fonctionnalités d'un protocole de liaison

#### Problème

Considérons le scénario où A envoie une trame  $I_0$  bien reçue par B mais dont l'acquittement lui arrive après expiration de la temporisation associée

- A émet à nouveau la trame  $I_0$
- Lorsqu'il reçoit l'acquittement de la première trame  $I_0$  émise, il considère qu'il s'agit de l'acquittement de la seconde et il émet la trame  $I_1$
- A interprète l'acquittement qui suit comme l'acquittement de sa trame  $I_1$
- Si, par malheur,  $I_1$  est mal transmise, elle va manquer à B : A croit qu'elle est acquittée alors qu'en réalité B ne l'a pas reçue

## Chapitre II

### Les protocoles de liaison de données

#### Numérotation des trames d'information

#### Fonctionnalités d'un protocole de liaison

#### Problème

Considérons le scénario où A envoie une trame  $I_0$  bien reçue par B mais dont l'acquittement lui arrive après expiration de la temporisation associée

- A émet à nouveau la trame  $I_0$
- Lorsqu'il reçoit l'acquittement de la première trame  $I_0$  émise, il considère qu'il s'agit de l'acquittement de la seconde et il émet la trame  $I_1$
- A interprète l'acquittement qui suit comme l'acquittement de sa trame  $I_1$
- Si, par malheur,  $I_1$  est mal transmise, elle va manquer à B : A croit qu'elle est acquittée alors qu'en réalité B ne l'a pas reçue



## Chapitre II

### Les protocoles de liaison de données

#### Numérotation des trames d'information

#### Fonctionnalités d'un protocole de liaison

#### Problème

Considérons le scénario où A envoie une trame  $I_0$  bien reçue par B mais dont l'acquittement lui arrive après expiration de la temporisation associée

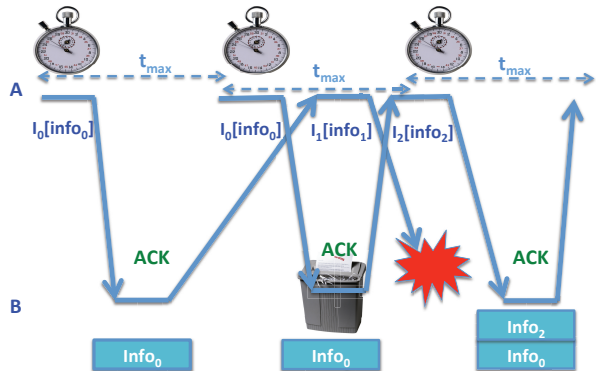
- A émet à nouveau la trame  $I_0$
- Lorsqu'il reçoit l'acquittement de la première trame  $I_0$  émise, il considère qu'il s'agit de l'acquittement de la seconde et il émet la trame  $I_1$
- A interprète l'acquittement qui suit comme l'acquittement de sa trame  $I_1$
- Si, par malheur,  $I_1$  est mal transmise, elle va manquer à B : A croit qu'elle est acquittée alors qu'en réalité B ne l'a pas reçue

# Chapitre II

## Les protocoles de liaison de données

### Numérotation des trames d'information

Fonctionnalités  
d'un protocole  
de liaison



**Solution**

On est donc obligé de préciser dans une trame d'acquittement le numéro de la trame d'information qu'elle acquitte

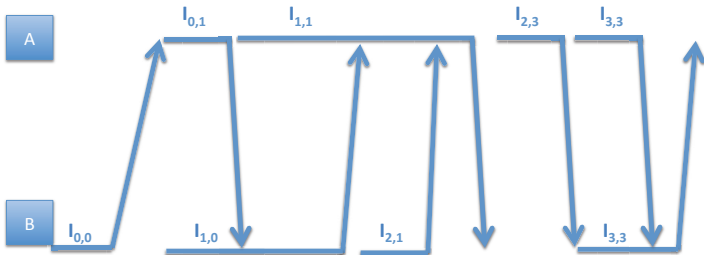
# Chapitre II

## Les protocoles de liaison de données

Fonctionnalités  
d'un protocole  
de liaison

### Numérotation des trames d'information

Stratégie : **Piggy-backing**



Chaque trame  $I$  possède deux numéros : un numéro  $N(S)$ , et un numéro  $N(R)$  acquittant les trames émises dans le sens opposé

## Chapitre II

### Les protocoles de liaison de données

#### Fonctionnalités d'un protocole de liaison

##### Notion de fenêtre glissante

Afin d'augmenter l'efficacité du dialogue, on introduit la notion d'anticipation

- Possibilité d'émettre plusieurs trames successives, sans avoir reçu l'acquittement des trames précédentes
- Ainsi, une trame de supervision n'acquitte plus une seule trame mais un ensemble de trames qui se suivent sans erreur

## Chapitre II

### Les protocoles de liaison de données

#### Fonctionnalités d'un protocole de liaison

##### Notion de fenêtre glissante

Afin d'augmenter l'efficacité du dialogue, on introduit la notion d'anticipation

- Possibilité d'émettre plusieurs trames successives, sans avoir reçu l'acquittement des trames précédentes
- Ainsi, une trame de supervision n'acquitte plus une seule trame mais un ensemble de trames qui se suivent sans erreur
- Le nombre de trames successives qu'on peut émettre sans réception d'acquittement est limité par une valeur notée  $n$

## Chapitre II

### Les protocoles de liaison de données

#### Fonctionnalités d'un protocole de liaison

##### Notion de fenêtre glissante

Afin d'augmenter l'efficacité du dialogue, on introduit la notion d'anticipation

- Possibilité d'émettre plusieurs trames successives, sans avoir reçu l'acquittement des trames précédentes
- Ainsi, une trame de supervision n'acquitte plus une seule trame mais un ensemble de trames qui se suivent sans erreur
- Le nombre de trames successives qu'on peut émettre sans réception d'acquittement est limité par une valeur notée  $n$

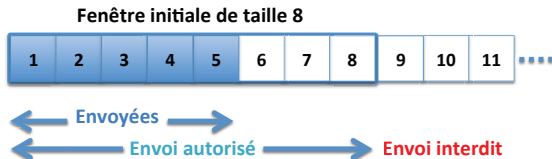
# Chapitre II

## Les protocoles de liaison de données

### Fonctionnalités d'un protocole de liaison

#### Notion de fenêtre glissante

- Emettre n trames sans attendre d'acquittement ( n est la taille de la fenêtre )

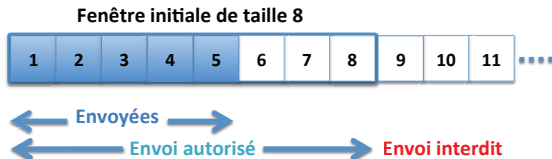


# Chapitre II

## Les protocoles de liaison de données

### Notion de fenêtre glissante

- Emettre n trames sans attendre d'acquittement ( n est la taille de la fenêtre )



- Un temporisateur par trame émise

Fonctionnalités  
d'un protocole  
de liaison

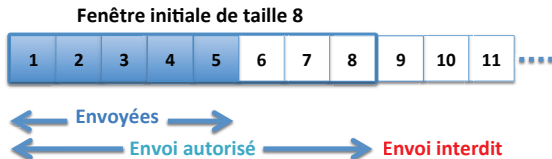


# Chapitre II

## Les protocoles de liaison de données

### Notion de fenêtre glissante

- Emettre n trames sans attendre d'acquittement ( n est la taille de la fenêtre )



- Un temporisateur par trame émise
- La réception de l'acquittement de la première trame de la fenêtre la fait glisser :

Fonctionnalités  
d'un protocole  
de liaison

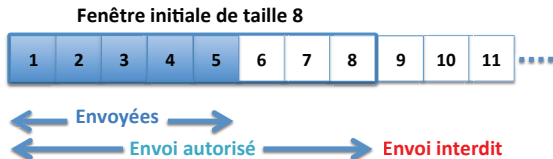
# Chapitre II

## Les protocoles de liaison de données

### Fonctionnalités d'un protocole de liaison

#### Notion de fenêtre glissante

- Emettre n trames sans attendre d'acquittement ( n est la taille de la fenêtre )



- Un temporisateur par trame émise
- La réception de l'acquittement de la première trame de la fenêtre la fait glisser :



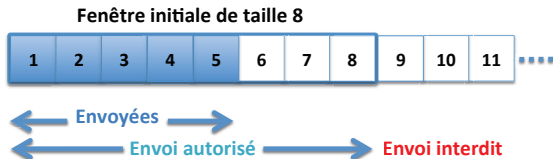
# Chapitre II

## Les protocoles de liaison de données

### Fonctionnalités d'un protocole de liaison

#### Notion de fenêtre glissante

- Emettre n trames sans attendre d'acquittement ( n est la taille de la fenêtre )



- Un temporisateur par trame émise
- La réception de l'acquittement de la première trame de la fenêtre la fait glisser :

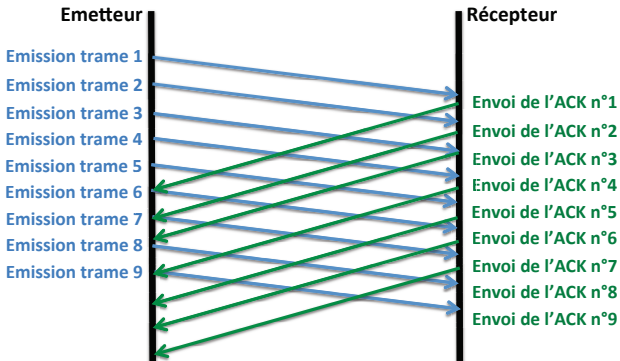


# Chapitre II

## Les protocoles de liaison de données

Fonctionnalités  
d'un protocole  
de liaison

Notion de fenêtre glissante



## Chapitre II

# Les protocoles de liaison de données

### Fonctionnalités d'un protocole de liaison

#### Notion de fenêtre glissante

S'il y a perte de donnée, alors deux possibilités de rejets :

- Rejet total dit aussi global

## Chapitre II

# Les protocoles de liaison de données

### Fonctionnalités d'un protocole de liaison

#### Notion de fenêtre glissante

S'il y a perte de donnée, alors deux possibilités de rejets :

- Rejet total dit aussi global
- Rejet sélectif

## Chapitre II

# Les protocoles de liaison de données

### Fonctionnalités d'un protocole de liaison

#### Notion de fenêtre glissante

S'il y a perte de donnée, alors deux possibilités de rejets :

- Rejet total dit aussi global
- Rejet sélectif

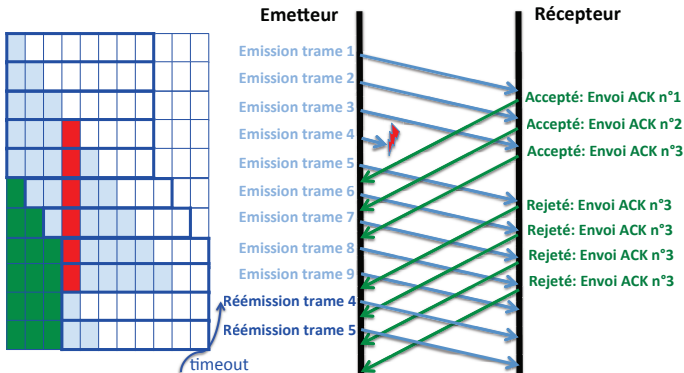
# Chapitre II

## Les protocoles de liaison de données

### Fonctionnalités d'un protocole de liaison

#### Notion de fenêtre glissante

**Rejet total (global) :** Aucune trame suivant celle perdue n'est acquittée





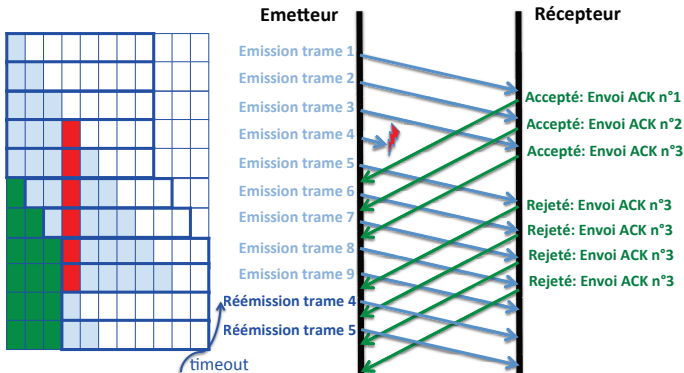
# Chapitre II

## Les protocoles de liaison de données

### Fonctionnalités d'un protocole de liaison

#### Notion de fenêtre glissante

**Rejet total (global) :** Aucune trame suivant celle perdue n'est acquittée



## Chapitre II

### Les protocoles de liaison de données

#### Fonctionnalités d'un protocole de liaison

#### Notion de fenêtre glissante

**Rejet sélectif** : Soit la trame perdue i

- L'idée est d'introduire la notion fenêtre en réception

## Chapitre II

### Les protocoles de liaison de données

#### Fonctionnalités d'un protocole de liaison

#### Notion de fenêtre glissante

**Rejet sélectif** : Soit la trame perdue  $i$

- L'idée est d'introduire la notion fenêtre en réception
- La fenêtre en réception a une certaine taille  $m$  (souvent  $m=n$ )

## Chapitre II

### Les protocoles de liaison de données

#### Fonctionnalités d'un protocole de liaison

#### Notion de fenêtre glissante

**Rejet sélectif** : Soit la trame perdue  $i$

- L'idée est d'introduire la notion fenêtre en réception
- La fenêtre en réception a une certaine taille  $m$  (souvent  $m=n$ )
- A la réception, on ne garde que les trames qui entrent dans la fenêtre, les autres sont ignorées

## Chapitre II

### Les protocoles de liaison de données

#### Fonctionnalités d'un protocole de liaison

#### Notion de fenêtre glissante

**Rejet sélectif** : Soit la trame perdue  $i$

- L'idée est d'introduire la notion fenêtre en réception
- La fenêtre en réception a une certaine taille  $m$  (souvent  $m=n$ )
- A la réception, on ne garde que les trames qui entrent dans la fenêtre, les autres sont ignorées
- Les ACK des trames ont pour numéro  $i-1$

## Chapitre II

### Les protocoles de liaison de données

#### Fonctionnalités d'un protocole de liaison

#### Notion de fenêtre glissante

**Rejet sélectif** : Soit la trame perdue  $i$

- L'idée est d'introduire la notion fenêtre en réception
- La fenêtre en réception a une certaine taille  $m$  (souvent  $m=n$ )
- A la réception, on ne garde que les trames qui entrent dans la fenêtre, les autres sont ignorées
- Les ACK des trames ont pour numéro  $i-1$
- Si timeout, l'émetteur n'envoie que la première trame non acquittée

## Chapitre II

### Les protocoles de liaison de données

#### Fonctionnalités d'un protocole de liaison

##### Notion de fenêtre glissante

**Rejet sélectif** : Soit la trame perdue  $i$

- L'idée est d'introduire la notion fenêtre en réception
- La fenêtre en réception a une certaine taille  $m$  (souvent  $m=n$ )
- A la réception, on ne garde que les trames qui entrent dans la fenêtre, les autres sont ignorées
- Les ACK des trames ont pour numéro  $i-1$
- Si timeout, l'émetteur n'envoie que la première trame non acquittée
- Lorsque la trame est réémise et reçue, alors l'ACK renvoyé est :

# Chapitre II

## Les protocoles de liaison de données

### Fonctionnalités d'un protocole de liaison

#### Notion de fenêtre glissante

**Rejet sélectif** : Soit la trame perdue  $i$

- L'idée est d'introduire la notion fenêtre en réception
- La fenêtre en réception a une certaine taille  $m$  (souvent  $m=n$ )
- A la réception, on ne garde que les trames qui entrent dans la fenêtre, les autres sont ignorées
- Les ACK des trames ont pour numéro  $i-1$
- Si timeout, l'émetteur n'envoie que la première trame non acquittée
- Lorsque la trame est réémise et reçue, alors l'ACK renvoyé est :
  - 1 Celui de la dernière trame reçue



# Chapitre II

## Les protocoles de liaison de données

### Fonctionnalités d'un protocole de liaison

#### Notion de fenêtre glissante

**Rejet sélectif** : Soit la trame perdue  $i$

- L'idée est d'introduire la notion fenêtre en réception
- La fenêtre en réception a une certaine taille  $m$  (souvent  $m=n$ )
- A la réception, on ne garde que les trames qui entrent dans la fenêtre, les autres sont ignorées
- Les ACK des trames ont pour numéro  $i-1$
- Si timeout, l'émetteur n'envoie que la première trame non acquittée
- Lorsque la trame est réémise et reçue, alors l'ACK renvoyé est :
  - 1 Celui de la dernière trame reçue
  - 2 Celui avant une autre trame perdue

# Chapitre II

## Les protocoles de liaison de données

### Fonctionnalités d'un protocole de liaison

#### Notion de fenêtre glissante

**Rejet sélectif** : Soit la trame perdue  $i$

- L'idée est d'introduire la notion fenêtre en réception
- La fenêtre en réception a une certaine taille  $m$  (souvent  $m=n$ )
- A la réception, on ne garde que les trames qui entrent dans la fenêtre, les autres sont ignorées
- Les ACK des trames ont pour numéro  $i-1$
- Si timeout, l'émetteur n'envoie que la première trame non acquittée
- Lorsque la trame est réémise et reçue, alors l'ACK renvoyé est :
  - 1 Celui de la dernière trame reçue
  - 2 Celui avant une autre trame perdue

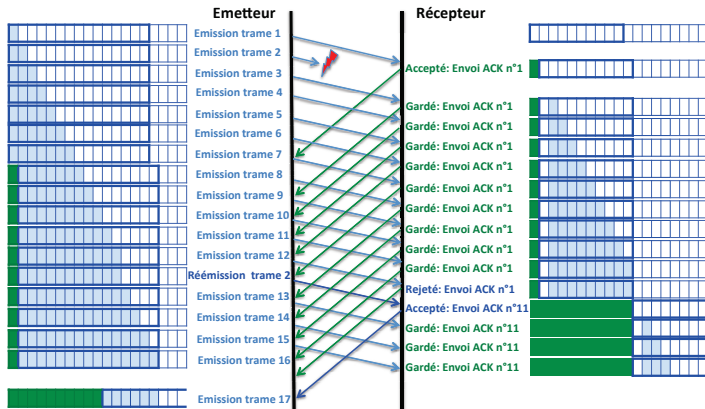
# Chapitre II

## Les protocoles de liaison de données

Fonctionnalités  
d'un protocole  
de liaison

Notion de fenêtre glissante

Rejet sélectif



## Chapitre II

# Les protocoles de liaison de données

### Description du protocole HDLC (High level Data Link Control)

#### Structure d'une trame HDLC

- HDLC est le protocole normalisé par l'ITU (International Telecommunications Union)
- Décrit une transmission en duplex intégral

## Chapitre II

### Les protocoles de liaison de données

#### Description du protocole HDLC (High level Data Link Control)

#### Structure d'une trame HDLC

- HDLC est le protocole normalisé par l'ITU (International Telecommunications Union)
- Décrit une transmission en duplex intégral
- Ce protocole met en oeuvre le mécanisme de transparence décrit en début de chapitre

# Chapitre II

## Les protocoles de liaison de données

### Description du protocole HDLC (High level Data Link Control)

#### Structure d'une trame HDLC

- HDLC est le protocole normalisé par l'ITU (International Telecommunications Union)
- Décrit une transmission en duplex intégral
- Ce protocole met en oeuvre le mécanisme de transparence décrit en début de chapitre
- Mécanisme de contrôle de flux

## Chapitre II

### Les protocoles de liaison de données

#### Description du protocole HDLC (High level Data Link Control)

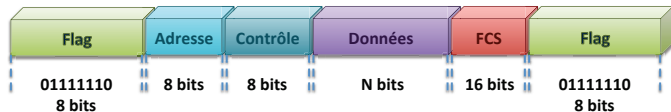
#### Structure d'une trame HDLC

- HDLC est le protocole normalisé par l'ITU (International Telecommunications Union)
- Décrit une transmission en duplex intégral
- Ce protocole met en oeuvre le mécanisme de transparence décrit en début de chapitre
- Mécanisme de contrôle de flux

# Chapitre II

## Les protocoles de liaison de données

### Structure d'une trame HDLC



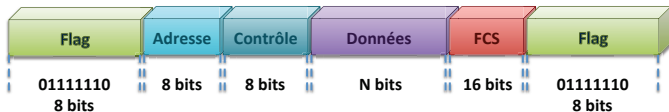
- Le champ Address s'étend sur un octet et identifie une des extrémités de la liaison
- Le champ Control décrit le type de la trame : il s'étend sur 1 octet



# Chapitre II

## Les protocoles de liaison de données

### Structure d'une trame HDLC



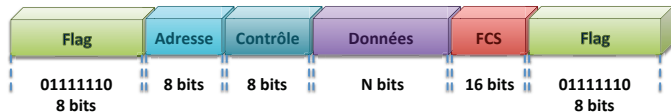
- Le champ Address s'étend sur un octet et identifie une des extrémités de la liaison
- Le champ Control décrit le type de la trame : il s'étend sur 1 octet
- Le champ Information contient les données

Description du  
protocole  
HDLC (High  
level Data Link  
Control)

# Chapitre II

## Les protocoles de liaison de données

### Structure d'une trame HDLC



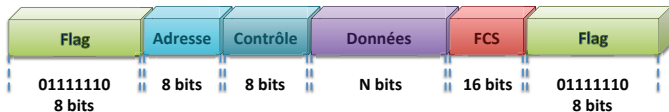
- Le champ Address s'étend sur un octet et identifie une des extrémités de la liaison
- Le champ Control décrit le type de la trame : il s'étend sur 1 octet
- Le champ Information contient les données
- Le champ FCS (Frame Control Sequence) obtenue par un contrôle polynomial dont le polynôme générateur vaut  $x^{16} + x^{12} + x^5 + 1$

Description du  
protocole  
HDLC (High  
level Data Link  
Control)

# Chapitre II

## Les protocoles de liaison de données

### Structure d'une trame HDLC

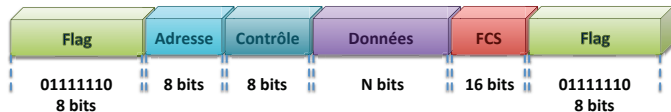


- Le champ Address s'étend sur un octet et identifie une des extrémités de la liaison
- Le champ Control décrit le type de la trame : il s'étend sur 1 octet
- Le champ Information contient les données
- Le champ FCS (Frame Control Sequence) obtenue par un contrôle polynomial dont le polynôme générateur vaut  $x^{16} + x^{12} + x^5 + 1$
- Les équipements émettent des suites de fanions pour maintenir la synchronisation entre eux

# Chapitre II

## Les protocoles de liaison de données

### Structure d'une trame HDLC



- Le champ Address s'étend sur un octet et identifie une des extrémités de la liaison
- Le champ Control décrit le type de la trame : il s'étend sur 1 octet
- Le champ Information contient les données
- Le champ FCS (Frame Control Sequence) obtenue par un contrôle polynomial dont le polynôme générateur vaut  $x^{16} + x^{12} + x^5 + 1$
- Les équipements émettent des suites de fanions pour maintenir la synchronisation entre eux

## Chapitre II

### Les protocoles de liaison de données

Description du  
protocole  
HDLC (High  
level Data Link  
Control)

#### Différents types de trames HDLC

Il existe trois types de trames identifiés par le champ  
Control :

- Les trames d'information ou trames I : Permettent la transmission de données de l'utilisateur

## Chapitre II

### Les protocoles de liaison de données

#### Description du protocole HDLC (High level Data Link Control)

#### Différents types de trames HDLC

Il existe trois types de trames identifiés par le champ Control :

- Les trames d'information ou trames I : Permettent la transmission de données de l'utilisateur
- Les trames de supervision ou trames S :

## Chapitre II

### Les protocoles de liaison de données

#### Description du protocole HDLC (High level Data Link Control)

#### Différents types de trames HDLC

Il existe trois types de trames identifiés par le champ Control :

- Les trames d'information ou trames I : Permettent la transmission de données de l'utilisateur
- Les trames de supervision ou trames S : Permettent l'acquittement et le contrôle de flux

## Chapitre II

### Les protocoles de liaison de données

#### Description du protocole HDLC (High level Data Link Control)

#### Différents types de trames HDLC

Il existe trois types de trames identifiés par le champ Control :

- Les trames d'information ou trames I : Permettent la transmission de données de l'utilisateur
- Les trames de supervision ou trames S : Permettent l'acquittement et le contrôle de flux
- Les trames non numérotées ou trames U (Unnumbered) :



## Chapitre II

### Les protocoles de liaison de données

#### Description du protocole HDLC (High level Data Link Control)

#### Différents types de trames HDLC

Il existe trois types de trames identifiés par le champ Control :

- Les trames d'information ou trames I : Permettent la transmission de données de l'utilisateur
- Les trames de supervision ou trames S : Permettent l'acquittement et le contrôle de flux
- Les trames non numérotées ou trames U (Unnumbered) : servent à commander la liaison : initialisation, libération, notification d'erreurs irrécupérables

## Chapitre II

### Les protocoles de liaison de données

#### Description du protocole HDLC (High level Data Link Control)

#### Différents types de trames HDLC

Il existe trois types de trames identifiés par le champ Control :

- Les trames d'information ou trames I : Permettent la transmission de données de l'utilisateur
- Les trames de supervision ou trames S : Permettent l'acquittement et le contrôle de flux
- Les trames non numérotées ou trames U (Unnumbered) : servent à commander la liaison : initialisation, libération, notification d'erreurs irrécupérables

## Chapitre II

### Les protocoles de liaison de données

Description du  
protocole  
HDLC (High  
level Data Link  
Control)

#### Différents types de trames HDLC

##### Les trames I :

- Pour transmettre des données
- Numérotées par la variable  $N(S)$

## Chapitre II

### Les protocoles de liaison de données

Description du  
protocole  
HDLC (High  
level Data Link  
Control)

#### Différents types de trames HDLC

##### Les trames I :

- Pour transmettre des données
- Numérotées par la variable  $N(S)$
- Contiennent également l'acquittement des trames reçues en sens inverse (procédé de piggy-backing), grâce au numéro  $N(R)$

## Chapitre II

### Les protocoles de liaison de données

Description du  
protocole  
HDLC (High  
level Data Link  
Control)

#### Différents types de trames HDLC

##### Les trames I :

- Pour transmettre des données
- Numérotées par la variable  $N(S)$
- Contiennent également l'acquittement des trames reçues en sens inverse (procédé de piggy-backing), grâce au numéro  $N(R)$

# Chapitre II

## Les protocoles de liaison de données

### Différents types de trames HDLC

#### Les trames I :

#### Format de l'octet Control des trames I

7	6	5	4	3	2	1	0
N(R)			P/F	N(S)		0	

- Le bit 1 de valeur 0 est spécifique à la trame I
- La valeur du bit P/F : P=1 requête en attente d'une réponse / F=1 réponse de la requête , P=F=0 sinon

Description du  
protocole  
HDLC (High  
level Data Link  
Control)

# Chapitre II

## Les protocoles de liaison de données

### Différents types de trames HDLC

Description du  
protocole  
HDLC (High  
level Data Link  
Control)

Les trames I :

Format de l'octet Control des trames I

7	6	5	4	3	2	1	0
N(R)			P/F	N(S)		0	

- Le bit 1 de valeur 0 est spécifique à la trame I
- La valeur du bit P/F : P=1 requête en attente d'une réponse / F=1 réponse de la requête , P=F=0 sinon

## Chapitre II

### Les protocoles de liaison de données

#### Description du protocole HDLC (High level Data Link Control)

#### Différents types de trames HDLC

**Les trames S** : Les trames S indiquent l'état de disponibilité des stations, les 3 trames de supervision sont :

- La trame RR indique que l'équipement est prêt à recevoir de nouvelles trames I
- La trame RNR indique que l'équipement n'est pas en mesure de recevoir de nouvelles trames I



## Chapitre II

### Les protocoles de liaison de données

#### Description du protocole HDLC (High level Data Link Control)

#### Différents types de trames HDLC

**Les trames S** : Les trames S indiquent l'état de disponibilité des stations, les 3 trames de supervision sont :

- La trame RR indique que l'équipement est prêt à recevoir de nouvelles trames I
- La trame RNR indique que l'équipement n'est pas en mesure de recevoir de nouvelles trames I
- La trame REJ sert à demander l'arrêt immédiat des émissions en cours et une retransmission à partir de la trame I portant le numéro indiqué dans  $N(R)$

## Chapitre II

### Les protocoles de liaison de données

#### Description du protocole HDLC (High level Data Link Control)

#### Différents types de trames HDLC

**Les trames S** : Les trames S indiquent l'état de disponibilité des stations, les 3 trames de supervision sont :

- La trame RR indique que l'équipement est prêt à recevoir de nouvelles trames I
- La trame RNR indique que l'équipement n'est pas en mesure de recevoir de nouvelles trames I
- La trame REJ sert à demander l'arrêt immédiat des émissions en cours et une retransmission à partir de la trame I portant le numéro indiqué dans  $N(R)$

# Chapitre II

## Les protocoles de liaison de données

Description du  
protocole  
HDLC (High  
level Data Link  
Control)

Différents types de trames HDLC

Les trames S :

Format de l'octet Control des trames S

7	6	5	4	3	2	1	0	Format général des trames S
N(R)			P/F	0	0	0	1	RR
N(R)			P/F	0	1	0	1	RNR
N(R)			P/F	1	0	0	1	REJ

## Chapitre II

### Les protocoles de liaison de données

#### Différents types de trames HDLC

**Les trames U** : On utilise les trames U pour les fonctions supplémentaires de commande de la liaison. Citons les principales :

- SABM (Set Asynchronous Balanced Mode) pour initialiser le fonctionnement en mode équilibré
- DISC (DISConnect) pour rompre logiquement la liaison

Description du  
protocole  
HDLC (High  
level Data Link  
Control)

## Chapitre II

### Les protocoles de liaison de données

#### Différents types de trames HDLC

**Les trames U** : On utilise les trames U pour les fonctions supplémentaires de commande de la liaison. Citons les principales :

- SABM (Set Asynchronous Balanced Mode) pour initialiser le fonctionnement en mode équilibré
- DISC (DISConnect) pour rompre logiquement la liaison
- UA (Unnumbered Acknowledgement) pour acquitter des commandes comme SABM ou DISC

Description du  
protocole  
HDLC (High  
level Data Link  
Control)

## Chapitre II

### Les protocoles de liaison de données

#### Différents types de trames HDLC

**Les trames U** : On utilise les trames U pour les fonctions supplémentaires de commande de la liaison. Citons les principales :

- SABM (Set Asynchronous Balanced Mode) pour initialiser le fonctionnement en mode équilibré
- DISC (DISConnect) pour rompre logiquement la liaison
- UA (Unnumbered Acknowledgement) pour acquitter des commandes comme SABM ou DISC
- FRMR (FRaMe Reject) pour rejeter une commande invalide (correcte du point de vue de la détection des erreurs mais incohérente par rapport à l'état du dialogue)

Description du  
protocole  
HDLC (High  
level Data Link  
Control)

## Chapitre II

### Les protocoles de liaison de données

#### Différents types de trames HDLC

**Les trames U** : On utilise les trames U pour les fonctions supplémentaires de commande de la liaison. Citons les principales :

- SABM (Set Asynchronous Balanced Mode) pour initialiser le fonctionnement en mode équilibré
- DISC (DISConnect) pour rompre logiquement la liaison
- UA (Unnumbered Acknowledgement) pour acquitter des commandes comme SABM ou DISC
- FRMR (FRaMe Reject) pour rejeter une commande invalide (correcte du point de vue de la détection des erreurs mais incohérente par rapport à l'état du dialogue)
- DM (Disconnect Mode) pour indiquer l'état de déconnexion d'une station, en particulier, pour répondre négativement à une demande d'initialisation par SABM

Description du  
protocole  
HDLC (High  
level Data Link  
Control)

# Chapitre II

## Les protocoles de liaison de données

### Différents types de trames HDLC

**Les trames U** : On utilise les trames U pour les fonctions supplémentaires de commande de la liaison. Citons les principales :

- SABM (Set Asynchronous Balanced Mode) pour initialiser le fonctionnement en mode équilibré
- DISC (DISConnect) pour rompre logiquement la liaison
- UA (Unnumbered Acknowledgement) pour acquitter des commandes comme SABM ou DISC
- FRMR (FRaMe Reject) pour rejeter une commande invalide (correcte du point de vue de la détection des erreurs mais incohérente par rapport à l'état du dialogue)
- DM (Disconnect Mode) pour indiquer l'état de déconnexion d'une station, en particulier, pour répondre négativement à une demande d'initialisation par SABM

Description du  
protocole  
HDLC (High  
level Data Link  
Control)



# Chapitre II

## Les protocoles de liaison de données

Description du  
protocole  
HDLC (High  
level Data Link  
Control)

### Différents types de trames HDLC

Les trames U :

Format de l'octet Control des trames U

7	6	5	4	3	2	1	0	
0	0	1	P/F	1	1	1	1	SABM
0	1	0	P/F	0	0	1	1	DISC
0	1	1	P/F	0	0	1	1	UA
1	0	0	P/F	0	1	1	1	FRMR
0	0	0	P/F	1	1	1	1	DM

## Chapitre II

# Les protocoles de liaison de données

### Cas particulier du protocole PPP (Point to Point Protocol

- Le protocole PPP est le protocole de liaison point à point utilisé dans Internet
- Il utilise les lignes téléphoniques de l'abonné pour accéder au réseau

## Chapitre II

# Les protocoles de liaison de données

### Cas particulier du protocole PPP (Point to Point Protocol

- Le protocole PPP est le protocole de liaison point à point utilisé dans Internet
- Il utilise les lignes téléphoniques de l'abonné pour accéder au réseau
- Il s'agit d'une version très simplifiée d'HDLC

## Chapitre II

### Les protocoles de liaison de données

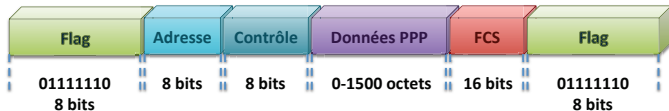
#### Cas particulier du protocole PPP (Point to Point Protocol

- Le protocole PPP est le protocole de liaison point à point utilisé dans Internet
- Il utilise les lignes téléphoniques de l'abonné pour accéder au réseau
- Il s'agit d'une version très simplifiée d'HDLC

## Chapitre II

### Les protocoles de liaison de données

#### Cas particulier du protocole PPP (Point to Point Protocol)

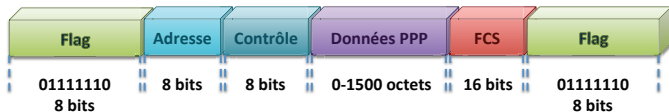


- Les 8 bits du champ Address sont à 1 (la liaison étant point à point, une seule valeur d'adresse suffit)

## Chapitre II

### Les protocoles de liaison de données

#### Cas particulier du protocole PPP (Point to Point Protocol)

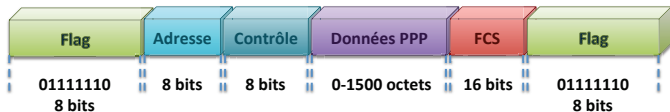


- Les 8 bits du champ Address sont à 1 (la liaison étant point à point, une seule valeur d'adresse suffit)
- Le champ Control a la même signification que dans HDLC

# Chapitre II

## Les protocoles de liaison de données

### Cas particulier du protocole PPP (Point to Point Protocol)

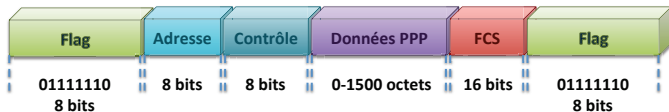


- Les 8 bits du champ Address sont à 1 (la liaison étant point à point, une seule valeur d'adresse suffit)
- Le champ Control a la même signification que dans HDLC
- Le champ Data PPP

## Chapitre II

### Les protocoles de liaison de données

#### Cas particulier du protocole PPP (Point to Point Protocol)



- Les 8 bits du champ Address sont à 1 (la liaison étant point à point, une seule valeur d'adresse suffit)
- Le champ Control a la même signification que dans HDLC
- Le champ Data PPP