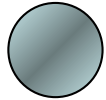




# Liaison de données

---

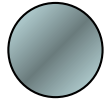


1. Caractéristiques / Fonctions
2. Gestion de liaison multipoint
3. Contrôle de flux
4. Reprise sur erreurs
- 5. Contrôle d'erreurs**
6. Protocole BSC
7. Protocole HDLC





# Détection d'erreurs

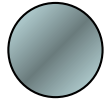


- Principe de base : **ajout de bits de contrôle** aux bits de données qui munit l'unité de données à envoyer d'une caractéristique qui sera vérifiée à la réception
- Codes détecteurs :
  - Contrôle sur un caractère par ajout de bit de parité : **VRC** (Vertical Redundancy Checking)
  - Contrôle sur un bloc de caractères par ajout d'un caractère de parité : **LRC** (Longitudinal Redundancy Checking)
  - Contrôle sur une trame (ou bloc) par ajout d'un code cyclique calculé par une division polynômiale : **CRC** (Cyclic Redundancy Check) ou **FCS** (Frame Check Sequence)





# Codes détecteurs d'erreurs (VRC)

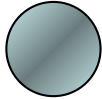


- Technique du « bit de parité » (VRC, Vertical Redundancy Check)
  - Ajout d'un bit de contrôle à la séquence binaire à transmettre telle que la somme des bits transmis soit paire (ou impaire, on parle alors de parité « impaire »)
  - Exemple : 0 1 0 1 0 0 0 **0**  

Données

P





# Codes détecteurs d'erreurs (LRC)

- Technique du LRC (Longitudinal Redundancy Check)
  - Technique du bit de parité complétée par un octet de contrôle calculé avec la parité des bits de même rang de chaque caractère
  - Possibilité de corriger une erreur (cf. plus loin distance de hamming)
  - Exemple (lettres codées en ASCII 7 bits) :

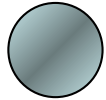
	T	E	S	T	LRC
Bit 1	1	1	1	1	0
Bit 2	0	0	0	0	0
Bit 3	1	0	1	1	1
Bit 4	0	0	0	0	0
Bit 5	1	1	0	1	1
Bit 6	0	0	1	0	1
Bit 7	0	1	1	0	0
<b>VRC</b>	<b>1</b>	<b>1</b>	<b>0</b>	<b>1</b>	<b>1</b>

- Train binaire transmis : T1 E1 S0 T1 0 0 1 0 1 1 0 1
- ↑     ↑     ↑     ↑  
VRC  
LRC





# Codes détecteurs d'erreurs (CRC)



- Code cyclique CRC (Cyclic Redundancy Check)  
(aussi appelé code polynomial)
- Principe général
  - On considère un bloc de  $k$  bits à transmettre comme un polynôme de degré  $(k-1)$  appelé  $P(x)$

Si  $b_n \dots b_0$  est la suite de bits à transmettre alors  $P(x) = b_n x^n + b_{n-1} x^{n-1} + \dots + b_0 x^0$

- Les bits de contrôle (CRC) correspondent au reste de la division de  $x^r * P(x)$  par  $G(x)$  où  $G(x)$  est appelé polynôme générateur de degré  $r$

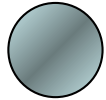
$$R(x) = \text{Reste} (x^r * P(x) / G(x)) \text{ où } r = \text{degré}(G(x))$$

- Le polynôme  $G(x)$  est connu a priori par l'émetteur et le récepteur  
(plusieurs polynômes  $G(x)$  ont été normalisés : CRC-12, CRC-16,...)





## Codes détecteurs d'erreurs (CRC, suite)



- Exemple : on souhaite transmettre la séquence 1 1 0 1 1 1

On suppose le polynôme générateur  $G(x) = x^2 + x + 1$

$$x^r * P(x) = x^2 * (x^5 + x^4 + x^2 + x + 1) = 1\ 1\ 0\ 1\ 1\ 1\ 0\ 0$$

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

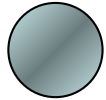
Reste = CRC

Message transmis = 1 1 0 1 1 1 1 1





# Détection et correction d'erreurs

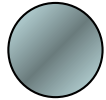


- Objectifs :
  - Effectuer une détection d'erreurs mais également proposer une correction des erreurs
  - (Réseaux) Eviter des retransmissions d'information en cas d'erreurs
  - (Sécurité) Localiser des modifications apportées à des documents
- Principes :
  - Redondance d'information
  - Effectuer des hypothèses sur le nombre d'erreurs probables et les possibilités de correction
- Exemple : Code de Hamming (1950)





# Code de Hamming



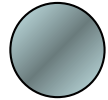
- Mot (ou mot de code) = { ensemble de bits à transmettre : données + contrôle }
- Information reçue = { ensemble de mots }
- **Distance de Hamming** = nombre de bits qui diffèrent entre deux mots de code
- Propriétés :
  - On note «  $d_h$  » la distance minimale de Hamming
  - Nb d'erreurs détectables =  $d_h - 1$  bits
  - Nb d'erreurs corrigeables =  $(d_h - 1) / 2$  bits



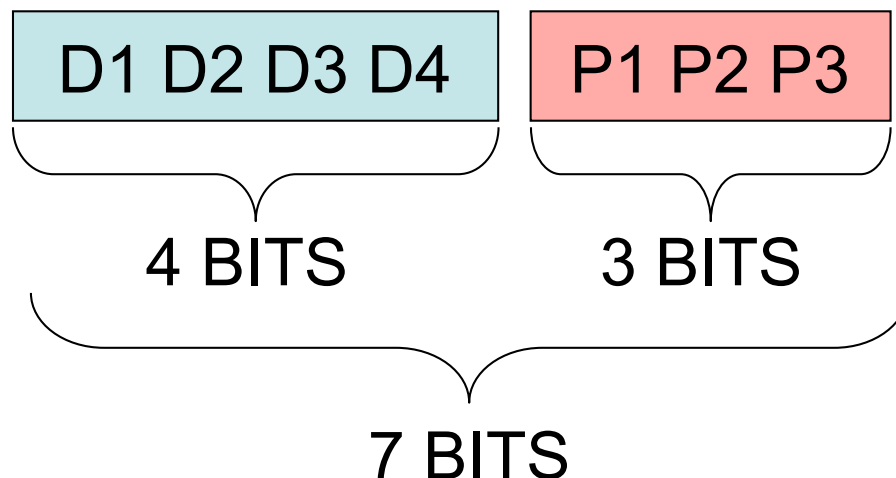




## Exemple : Hamming(7,4)



- Mot de code = 4 bits de données + 3 bits de parité
- 1 bit de donnée intervient dans le calcul (d'au moins) 2 bits de parité
- Distance de Hamming min = 3
  - Possibilité de détecter 1 ou 2 bits d'erreur
  - Possibilité de corriger 1 bit d'erreur

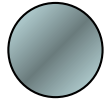


$$\begin{aligned} P1 &= D1 \oplus D2 \oplus D3 \\ P2 &= D2 \oplus D3 \oplus D4 \\ P3 &= D3 \oplus D4 \oplus D1 \end{aligned}$$





## Exemple : Hamming(7,4)



- Exemples de mots :

D1	D2	D3	D4	P1	P2	P3
0	0	1	0	1	1	1



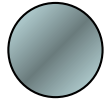
D1	D2	D3	D4	P1	P2	P3
0	1	1	0	0	0	1

0000 000	1000 101
0001 011	1001 110
0010 111	1010 010
0011 100	1011 001
0100 110	1100 011
0101 101	1101 000
0110 001	1110 100
0111 010	1111 111





## Hamming(7,4) : Cherchez l'erreur...



- **Hypothèse** : il n'y a pas plus d'une erreur dans le mot reçu

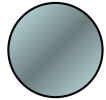
ER	MOT EMIS							MOT RECU						
	D1	D2	D3	D4	P1	P2	P3	ELEMENTS ERRONES						
D1	1	0	0	1	1	1	0	0	0	0	1	X	1	X
D2	0	1	0	1	1	0	1	0	0	0	1	X	X	1
D3	0	0	1	1	1	0	0	0	0	0	1	X	X	X
D4	0	0	0	0	0	0	0	0	0	0	1	0	X	X
P1	0	0	0	1	0	1	1	0	0	0	1	1	1	1
P2	0	0	1	0	1	1	1	0	0	1	0	1	0	1
P3	0	1	0	0	1	1	0	0	1	0	0	1	1	1





# Liaison de données

---

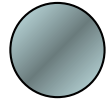


1. Caractéristiques / Fonctions
2. Gestion de liaison multipoint
3. Contrôle de flux
4. Reprise sur erreurs
5. Contrôle d'erreurs
- 6. Protocole BSC**
7. Protocole HDLC





# Panorama des protocoles de liaison



**Orienté caractère**

IBM **BSC** (Binary Synchronous Communication)

**Orienté bit**

IBM **SDLC** (Synchronous Data Link Control)

ISO

**HDLC** (High-level Data Link Control)

IETF

**PPP**

(Point to Point Protocol)  
(peut être aussi  
utilisé en mode  
caractère)

IEEE

**LLC**

(Logical Link Control)

ITU-T

**LAP**

(Link Access Procedure)

**LAP-D**

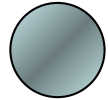
**LAP-B**

**LAP-F**





# Protocole BSC

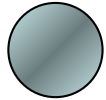


- Binary Synchronous Communication (ou « BiSync »)
  - IBM (~70)
- Protocole synchrone orienté caractères (l'unité d'information est le caractère)
  - Encodage des caractères avec code ASCII ou EBCDIC
  - Découpage des messages en blocs de caractères
- Mode point à point ou multipoint
- Half-duplex





# BSC : caractères de contrôle



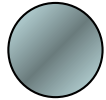
- **Formatage de blocs**
  - SOH (*Start Of Heading*) : début d'en-tête
  - STX (*Start of TeXt*) : début de texte
  - ETB (*End of Bloc*) : fin de bloc (qui n'est pas le dernier bloc d'un message)
  - ETX (*End Of TeXt*) : fin de texte
- **Contrôle de l'échange**
  - ACK / NACK : Acquittement positif / négatif
  - ENQ (*Enquiry*) : Demande d'établissement d'échange
  - EOT (*End Of Transmission*) : Fin de transmission (met la ligne à l'état disponible)
- **Synchronisation**
  - SYN : caractère de synchronisation pour transmission synchrone
- **Transparence et extension**
  - DLE (*Data Link Escape*) : caractère d'échappement





# BSC : caractères de contrôle

---



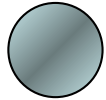
- Découpage des messages en blocs
- Structure d'un bloc
  - Début de bloc (STX)
  - Contenu du bloc
  - Fin de bloc (ETB ou ETX)
  - Contrôle de bloc (BCC ("Block Check Character"))  
(Ex. ASCII : LRC d'1 caractère, EBCDIC : CRC de 2 caractères)







## BSC : illustration échange de données



SYN SYN <caractère de contrôle> SYN SYN  
SYN SYN <caractère de début> <caractère de contrôle> SYN SYN

SYNSYN SOH <en-tête> ETB BCC  
SYN SYN SOH <en-tête> STX <texte> ETB BCC  
SYN SYN SOH <en-tête> STX <texte> ETX BCC

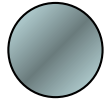
SYN SYN STX <texte> ETB BCC  
SYN SYN STX <texte> ETX BCC

SYN SYN DLE STX <texte transparent> DLE ETB BCC  
SYN SYN DLE STX <texte transparent> DLE ETX BCC

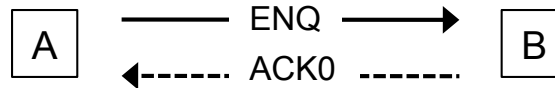




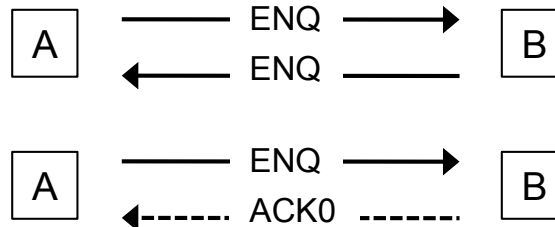
# BSC : acquisition de la voie (1)



- Point à point
  - Demande et acceptation de début de transmission

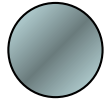


- Cas de collision d'ouverture  
L'une des deux stations s'efface





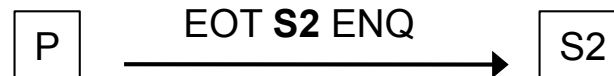
## BSC : acquisition de la voie (2)



- Multipoint

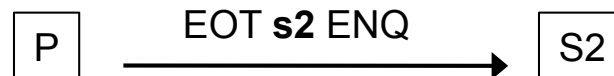
- **Invitation à émettre** (Scrutation, « polling »)

Nom de la station secondaire en majuscule



- **Invitation à recevoir** (Sélection, « selecting »)

Nom de la station secondaire en minuscule

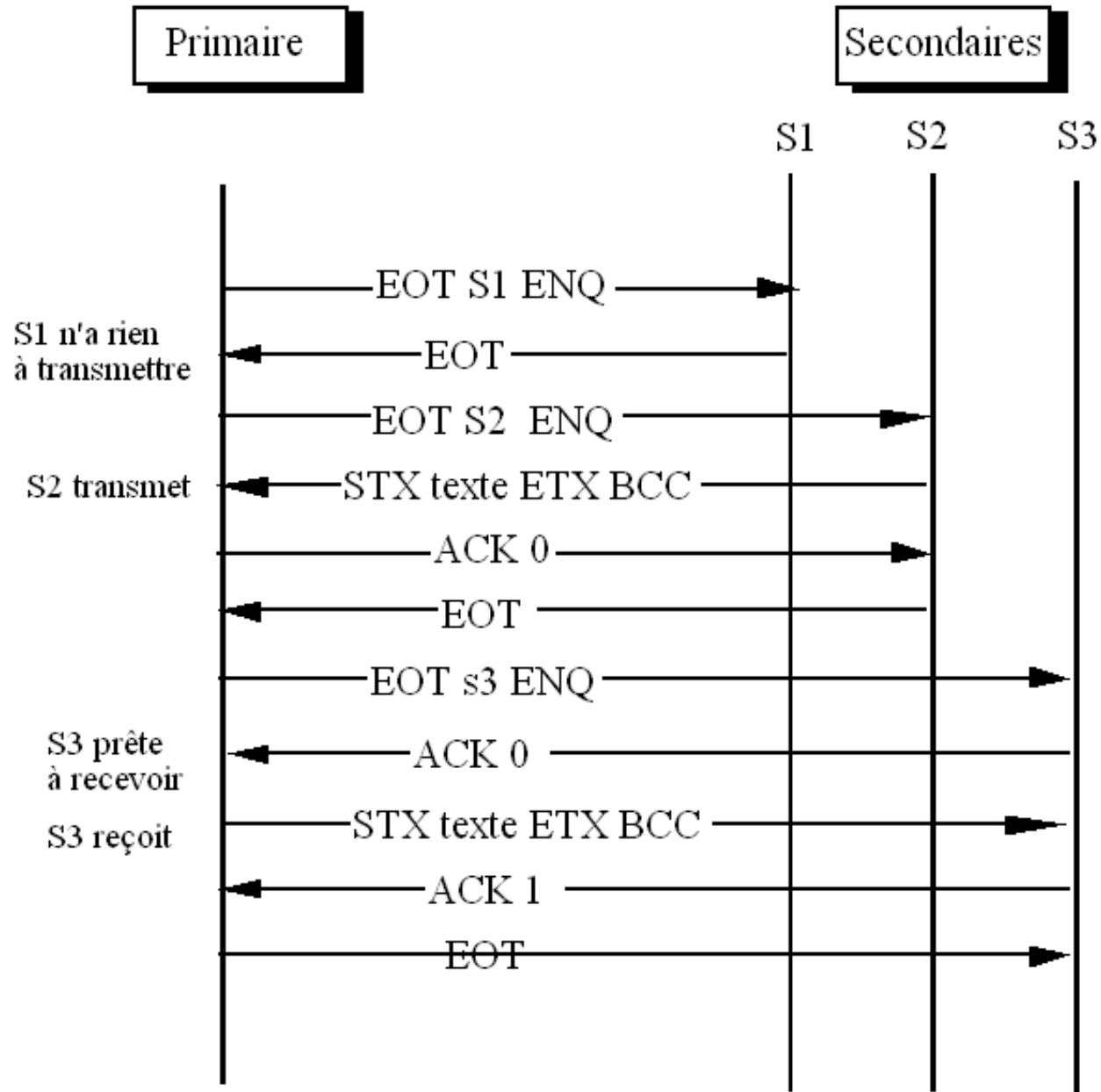
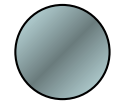


- Acceptation : ACK0
- Rejet : NAK





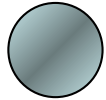
# BSC : illustration multipoint





# Liaison de données

---



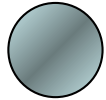
1. Caractéristiques / Fonctions
2. Gestion de liaison multipoint
3. Contrôle de flux
4. Reprise sur erreurs
5. Contrôle d'erreurs
6. Protocole BSC

## **7. Protocole HDLC**





# Protocole HDLC

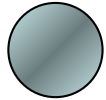


- High-level Data Link Control
  - Normalisé par l'ISO
- Protocole synchrone **orienté bit** (frames structurées en champs avec des codes de contrôle sous forme de séquences de bits)
- Contrôle de flux par anticipation
- Liaison point à point ou multipoint
- A la base de nombreux protocoles de niveau 2





# HDLC : les modes de fonctionnement

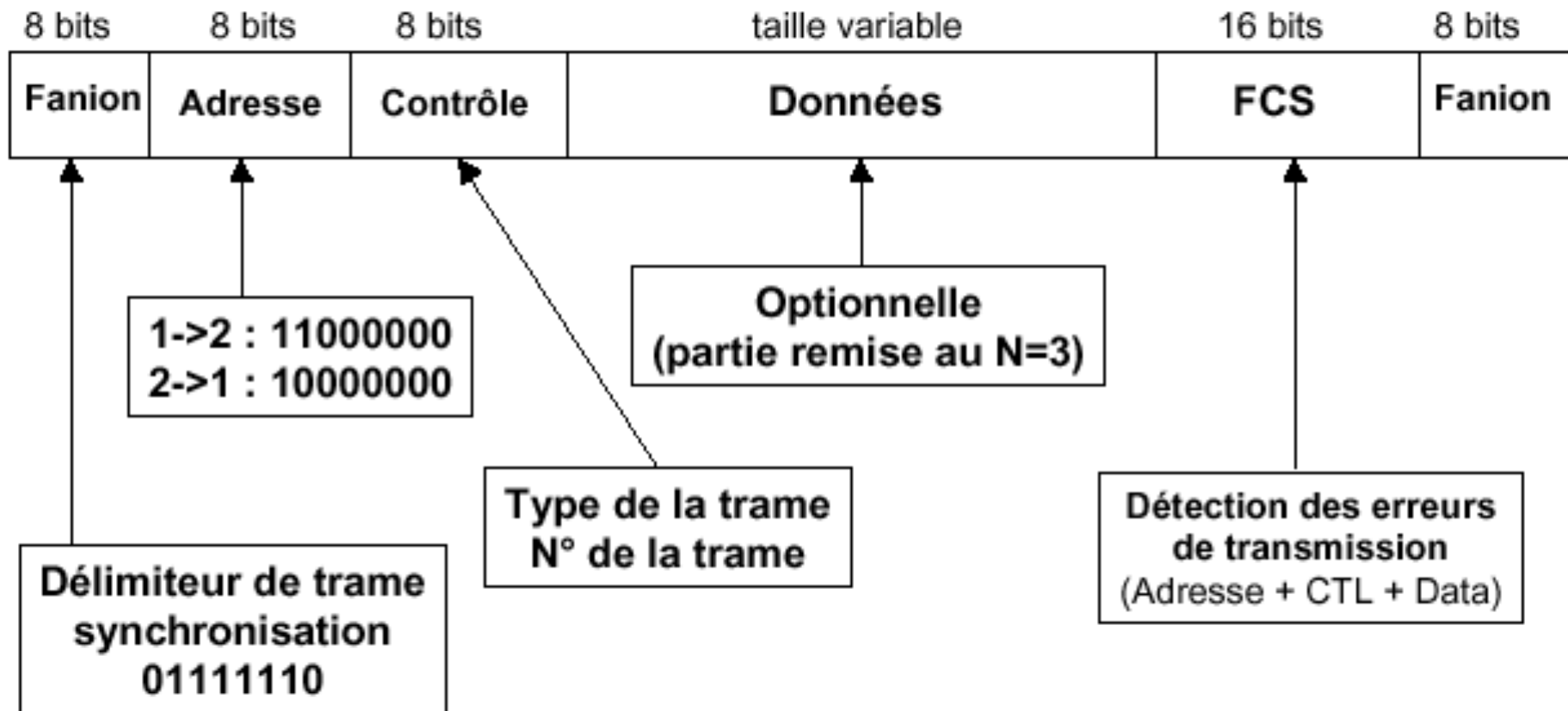
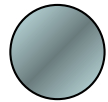


- Deux catégories de configuration
  - Non équilibré (*unbalanced*) : 1 primaire et N secondaires
  - Équilibré (*balanced*) : 2 terminaux d'égal à égal
- Trois modes de fonctionnement (mode d'échange)
  - **NRM** (*Normal Response Mode*)  
Configuration non équilibrée dans laquelle seul le primaire peut initier les transferts de données (*polling*)
  - **ARM** (*Asynchronous Response Mode*)  
Configuration non équilibrée dans laquelle les secondaires peuvent émettre sans autorisation du primaire
  - **ABM** (*Asynchronous Balanced Mode*)  
Configuration équilibrée dans laquelle chacune des deux stations peut initier la transmission (mode le plus répandu)





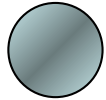
# HDLC : format des trames







# HDLC : types de trames

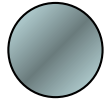


- Le protocole HDLC distingue **3 types de trames**
  - Les trames d'**information (I)** – *Information frames*  
Données de la couche réseau  
Eventuellement des acquittements (*piggybacking*)
  - Les trames de **supervision (S)** – *Supervisory frames*  
Supervision de l'échange des données (prêt ou non prêt à recevoir, rejet, acquittements)  
Pas de donnée
  - Les trames **non numérotées (U)** – *Unnumbered frames*  
Gestion de la liaison (établissement, libération)





## HDLC : types de trames (2)



Fanion	Adresse	Contrôle	Information	FCS	Fanion
01111110	8 bits	8 bits (ou 16)	Long. Variable ( $\geq 0$ )	16 bits (ou 32)	01111110

	B7	B6	B5	B4	B3	B2	B1	B0
Trame I	N(R)			P/F	N(S)			0
Trame S	N(R)			P/F	type		0	1
Trame U	type			P/F	type		1	1

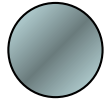
Format étendu (16 bits) :

	B7	B6	B5	B4	B3	B2	B1	B0	B7	B6	B5	B4	B3	B2	B1	B0
I	N(R)							P/F	N(S)							0
S	N(R)							P/F	0	0	0	0	type		0	1
U	0	0	0	0	0	0	0	P/F	type			U	type		1	1





# HDLC : les principales commandes

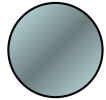


- **Trame I**
  - Trame d'information
- **Trame S**
  - **RR** : Receive Ready – type 00  
Prêt à recevoir la trame d'information N(R)
  - **RNR** : Receive Not Ready – type 01  
Non prêt à recevoir. Accusé de réception des trames jusqu'à N(R)-1
  - **REJ** : Reject – type 10  
Accusé de réception négatif.  
Demande de retransmission des trames à partir de N(R)
  - **SREJ** : Selective Reject – type 11  
Accusé de réception négatif.  
Demande de retransmission de la trame N(R)





## HDLC : les principales commandes (2)



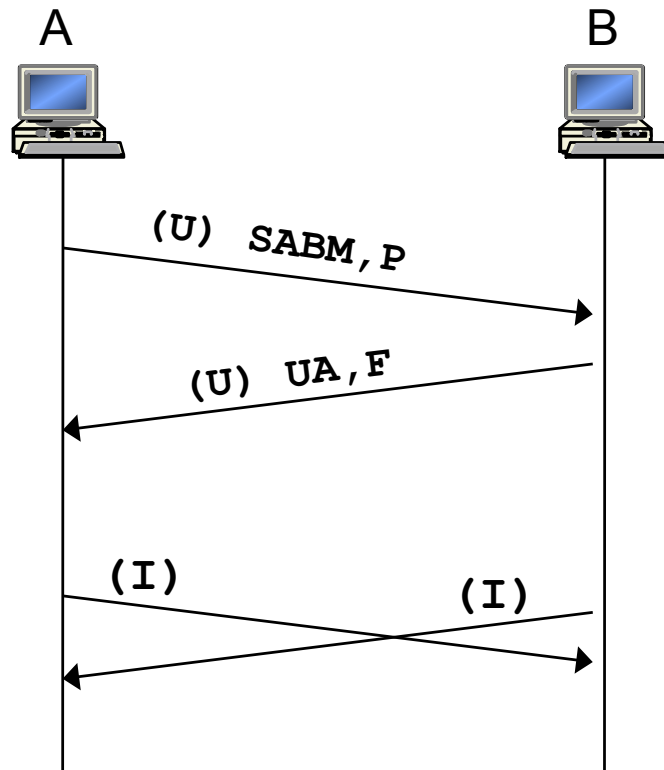
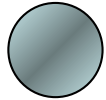
- **Trame U**

- **SNRM** : Set Normal Response Mode – type 100P0011  
Commande d'activation de mode.
- **SARM** : Set Asynchronous Response Mode – type 000P1111  
Commande d'activation de mode.
- **SABM** : Set Asynchronous Balanced Mode – type 001P1111  
Commande d'activation de mode.
- **DISC** : Disconnect – type 010P0011  
Commande de déconnexion.
- **UA** : Unnumbered Acknowledgment – type 011F0011  
Réponse. Accusé de réception d'une trame non numérotée (U).
- **FRMR** : Frame Reject – type 100F0111  
Réponse. Réception d'une trame non acceptée (raison du rejet dans le champ données de la trame).

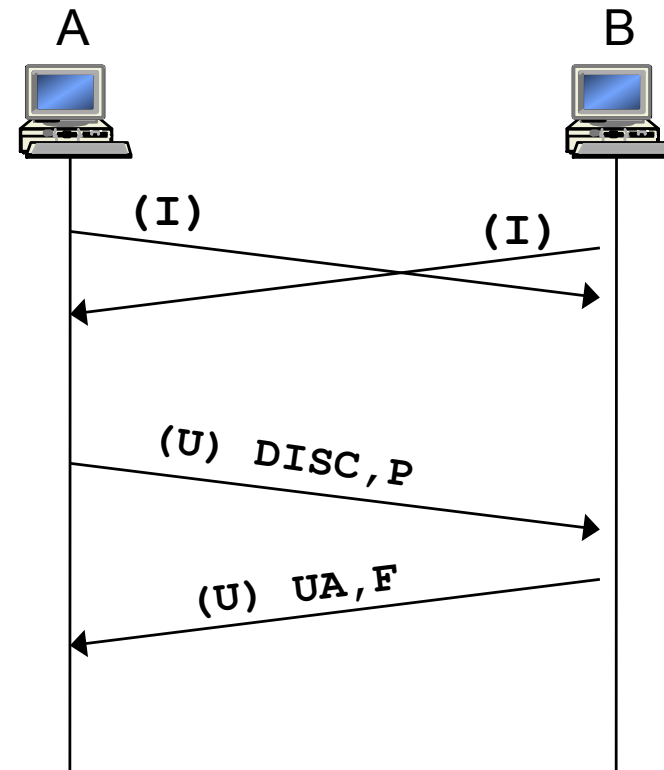




# HDLC : établissement et libération de connexion



Connexion

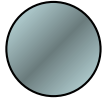


Déconnexion

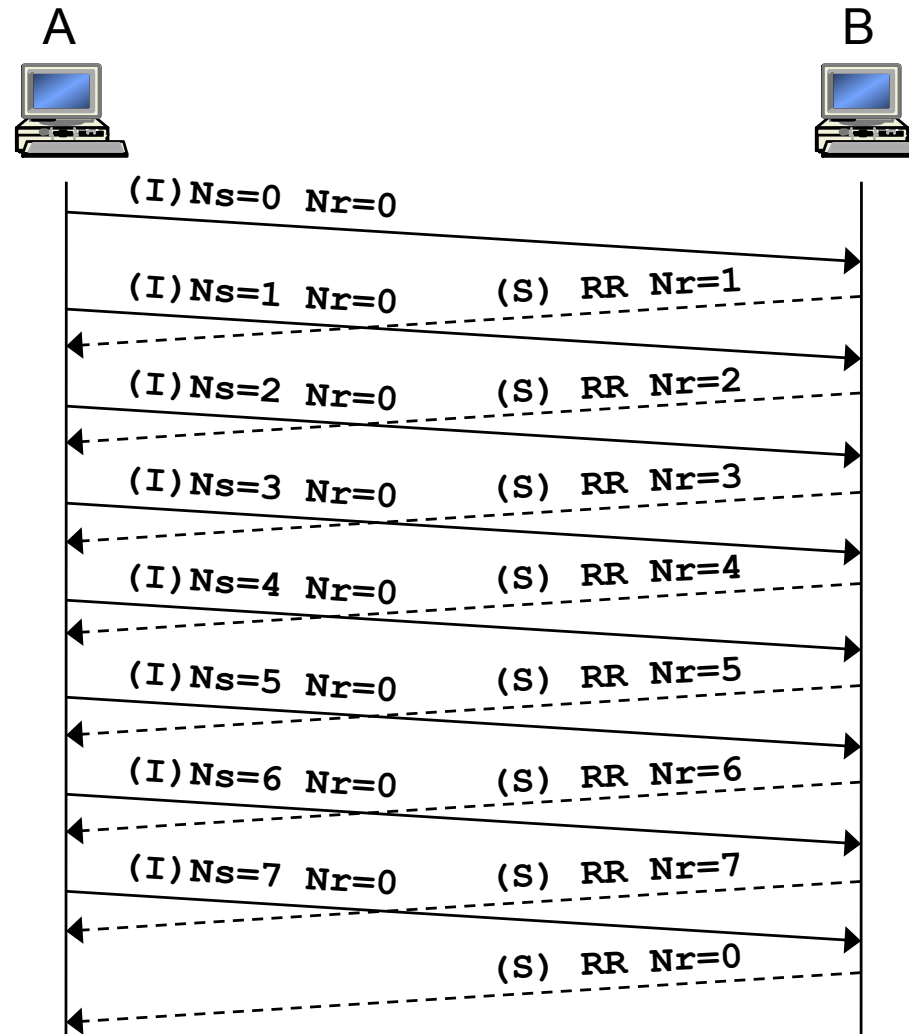




# HDLC : échange de données

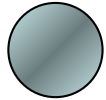


- Acquittements individuels

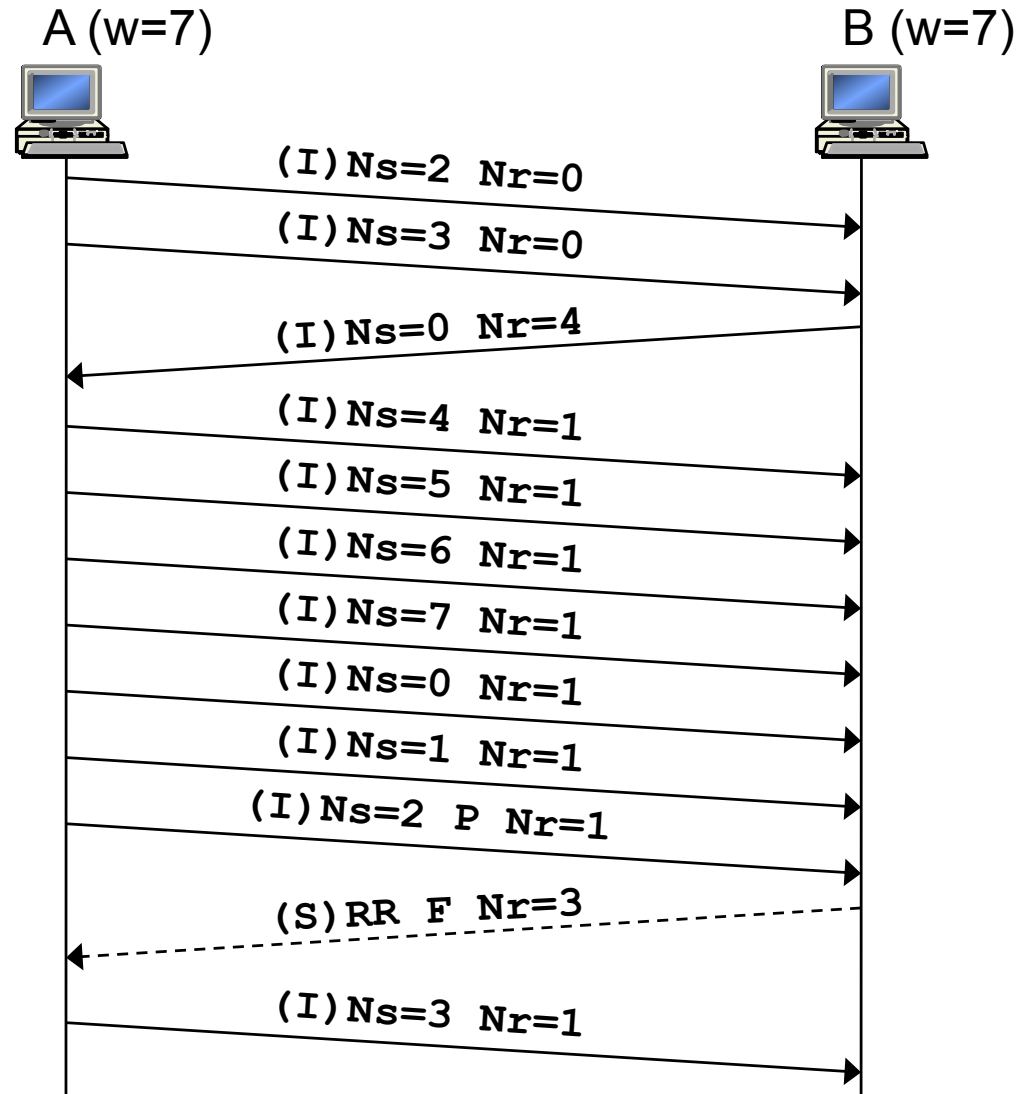




# HDLC : échange de données

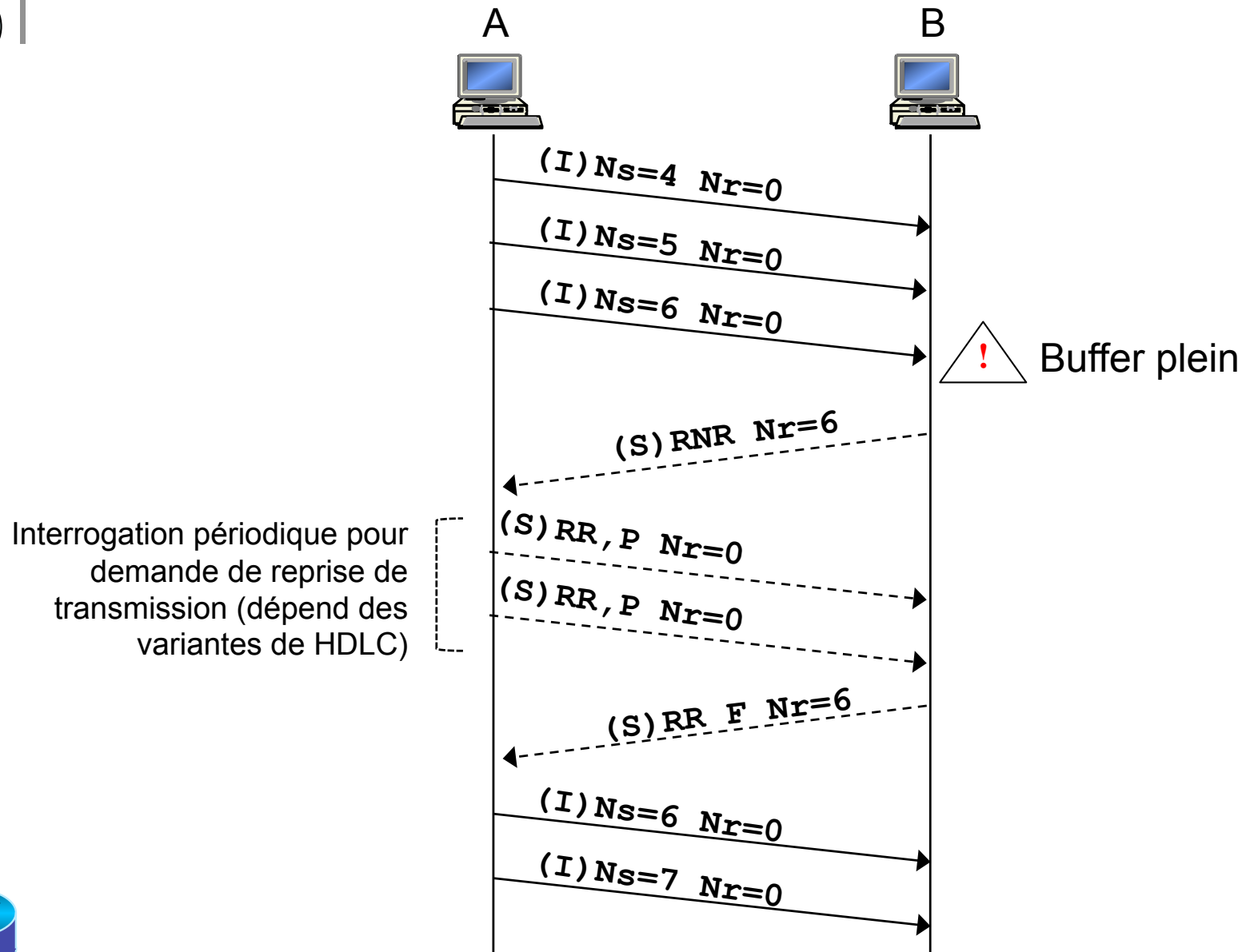
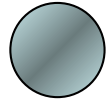


- Acquittements cumulatifs





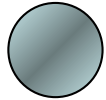
# HDLC : contrôle de flux



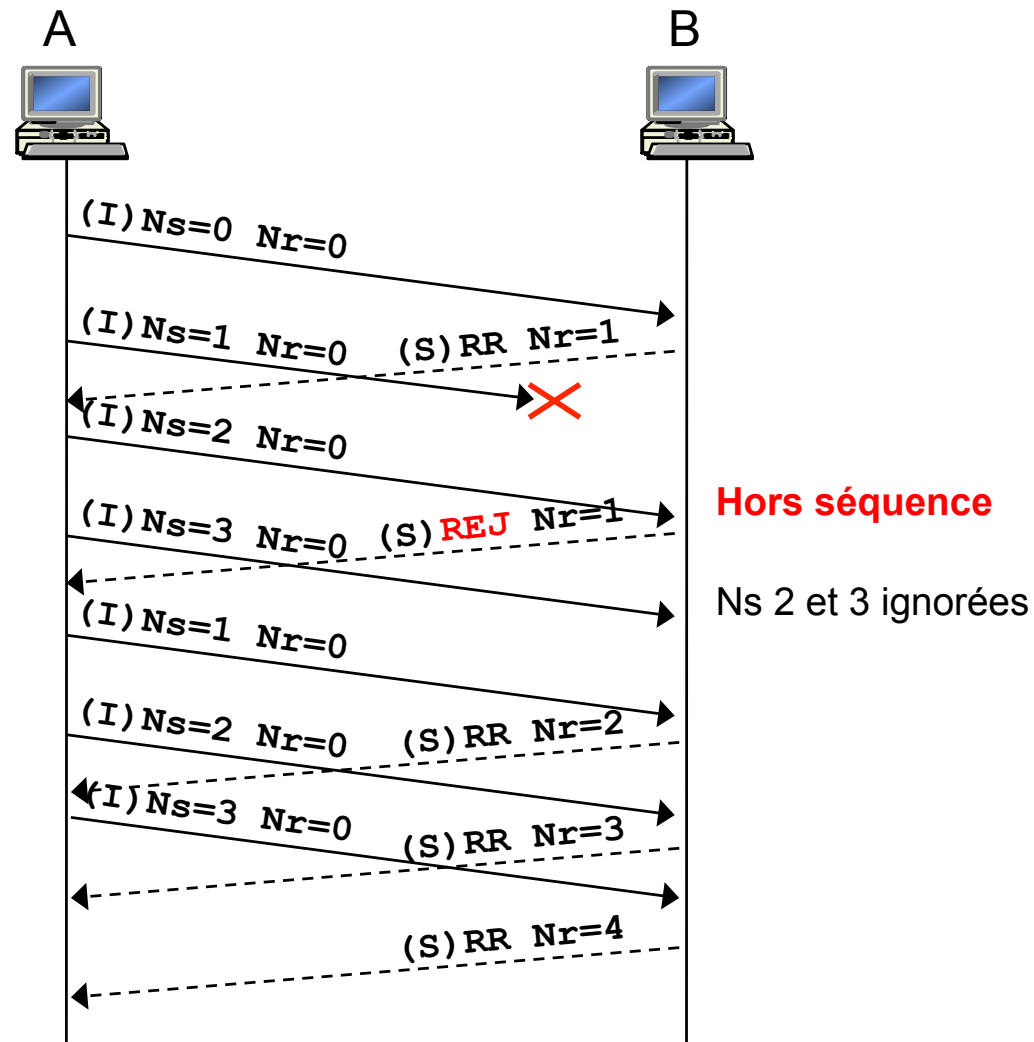




# HDLC : reprise sur erreurs

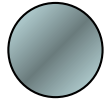


- Go Back N

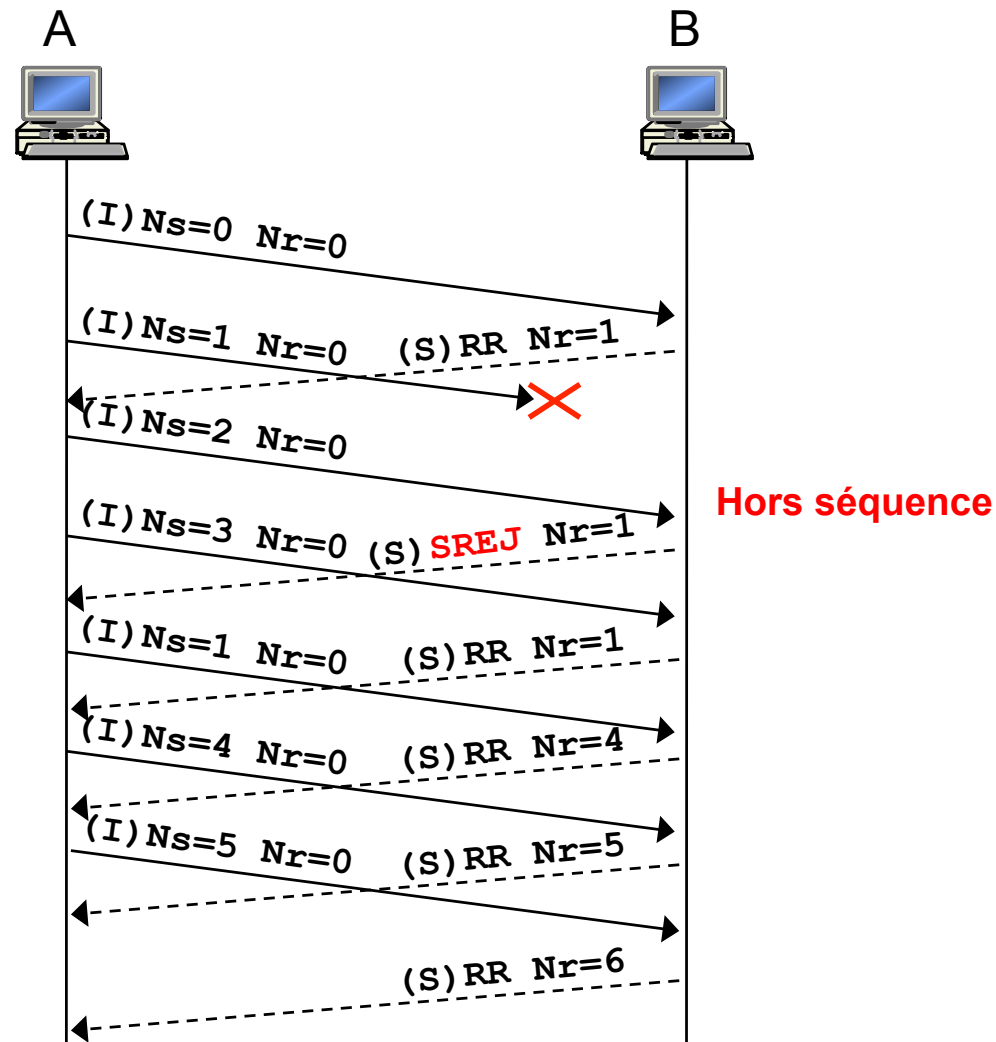




# HDLC : reprise sur erreurs

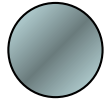


- Selective Reject

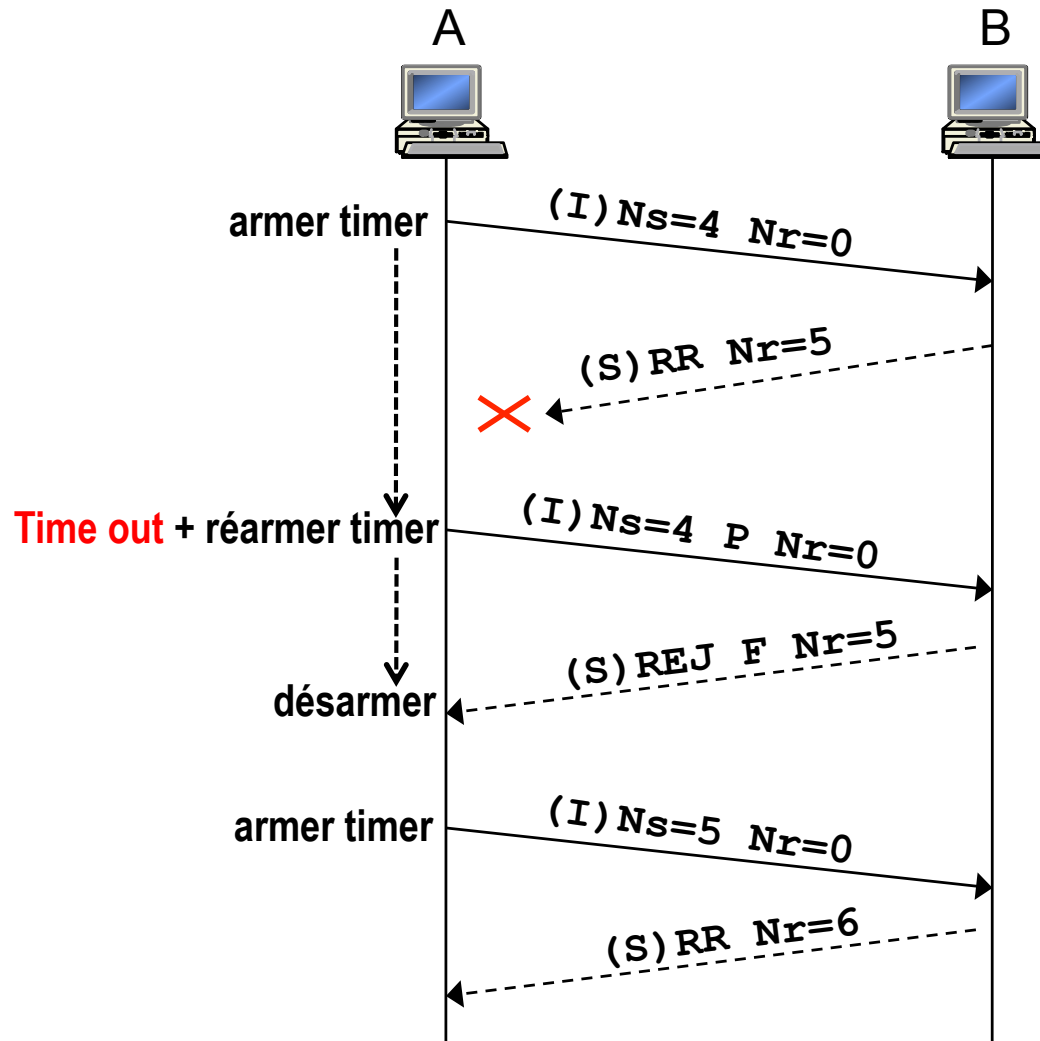




# HDLC : reprise sur erreurs

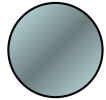


- Expiration temporisateur

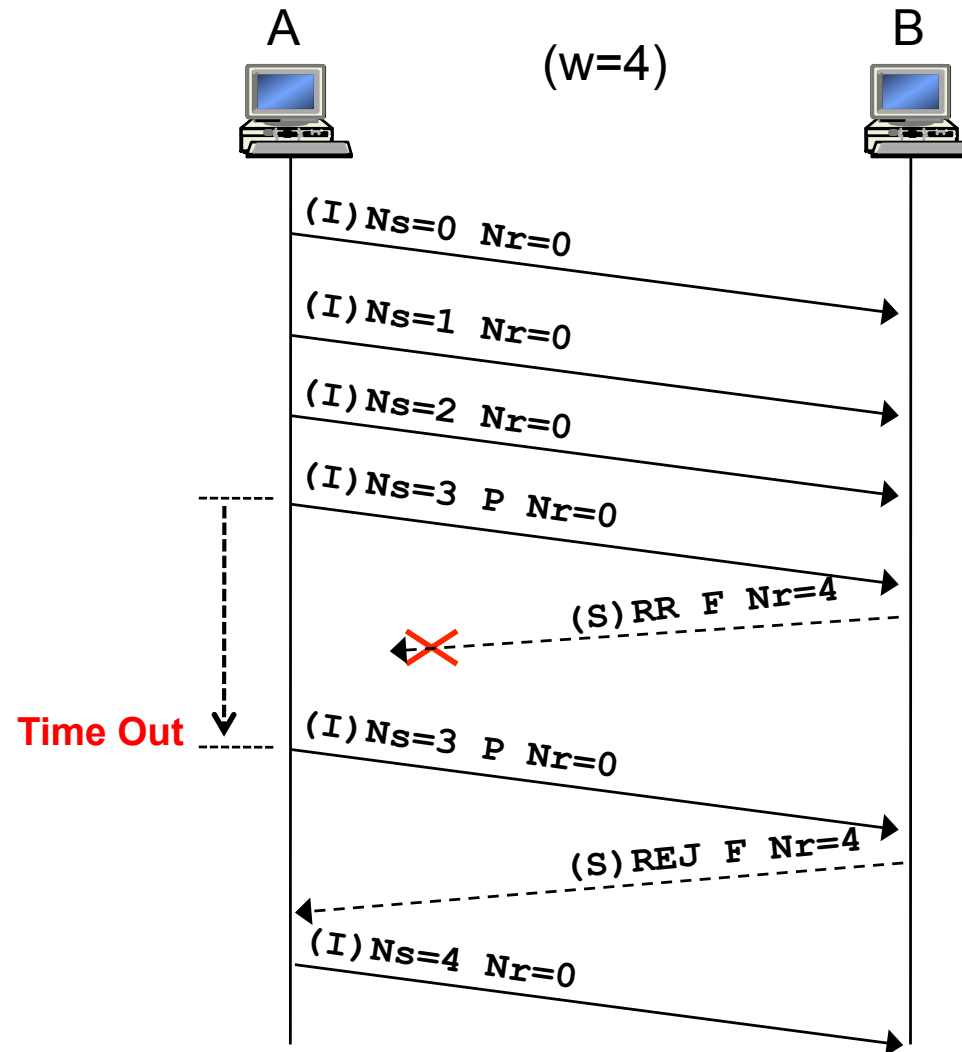




# HDLC : reprise sur erreurs

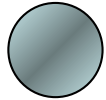


- Expiration temporisateur





# Variantes du protocole HDLC

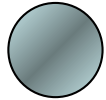


- Link Access Procedure/Protocol, Balanced (LAPB)
  - Réseau WAN X.25
- Link Access Procedure/Protocol, D-Channel (LAPD)
  - Réseau WAN RNIS
- Logical Link Control (LLC) – IEEE 802.2
  - Réseaux locaux IEEE 802
- Point to Point Protocol (PPP) sur liaison série – IETF
  - Ex. Réseau d'accès à un ISP via modem v90/v92





# Analogies protocoles liaison et transport



- Des analogies peuvent être faites entre les protocoles de la couche liaison (ex. HDLC) et les protocoles de niveau transport (ex. TCP) :
  - Numérotation de trames
  - Acquittements
  - Timer de retransmission
  - Fenêtre d'émission
  - Contrôle d'erreurs
- **Mais**
  - Liaison = traitement local
  - Transport = traitement de bout en bout

