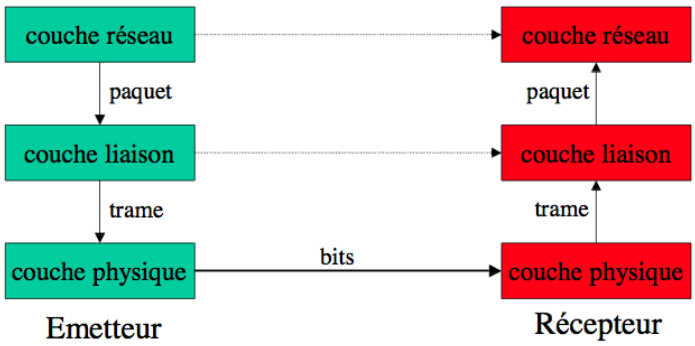
Chapitre 5

Couche Liaison de données

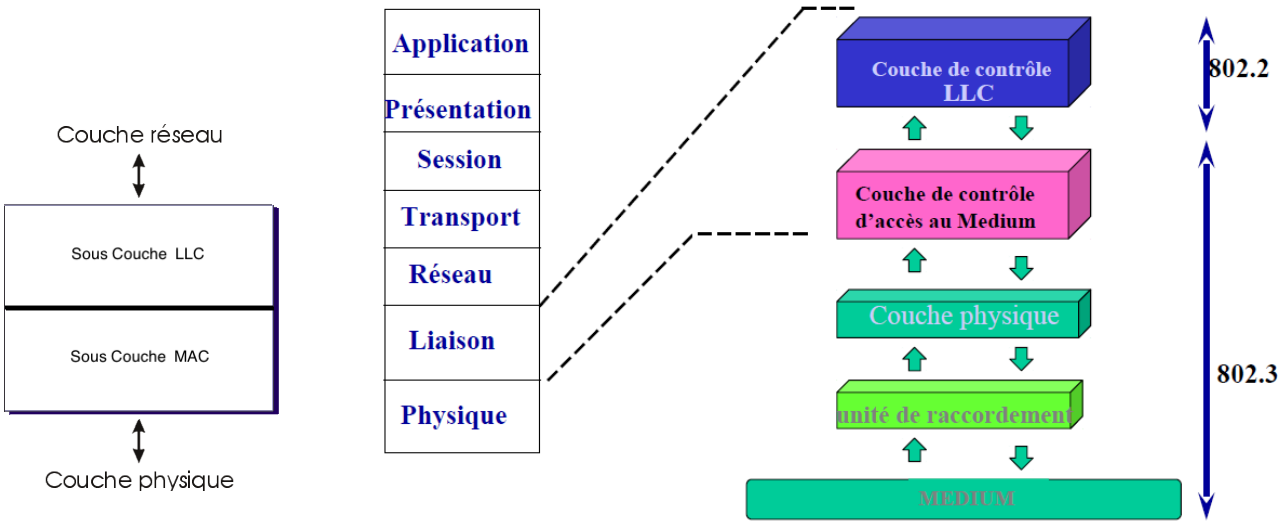
La couche physique permet de transmettre un flot de bits entre deux systèmes distants. La couche liaison (couche n° 2) doit s'assurer de la corrections des bits transmis et les rassembler en paquets pour les passer à la couche Réseaux. La couche liaison récupère des paquets de données de la couche réseau, les enveloppe en des trames qui les envoie une à une à la couche physique.



Pour rendre la transmission fiable, la couche liaison doit assurer :

- La délimitation des blocs de données échangées ;
- Le contrôle de l'intégrité des données reçues ;
- L'organisation et le contrôle de l'échange.
- Le contrôle d'accès à un canal partagé

En fait, la couche liaison se compose de deux sous-couches : LLC (Logical Link Control) et MAC (Media Access Control). La sous-couche LLC Assure les trois premières fonctions tandis que la sous-couche MAC assure la dernière.



5.1 Délimitation de trames

A l'instar des transmissions asynchrones où les bits de start et de stop encadrent les bits d'information, en transmission synchrone une information spéciale permet de repérer le début et la fin des données transmises. Il existe deux méthodes :

5.1.1 Comptage des caractères

On utilise un champ dans l'entête de la trame pour indiquer le nombre de caractères de la trame.

06	'S'	'U'	'P'	'E'	'R'	03	'L'	'E'	06	'C'	'O'	'U'	'R'	'S'

Un problème sérieux peut se poser avec cette méthode si la valeur du champ ajouté est modifiée au cours de la transmission. Généralement, cette méthode est rarement utilisée seule.

5.1.2 Utilisation des fanions

La trame est délimitée par une séquence particulière de bits appelée **fanion** ou flag.

Fanion	DONNEES	Fanion

Pour garantir le bon fonctionnement de cette méthode, des bits de transparence sont nécessaires pour qu'une séquence binaire dans la trame ne corresponde accidentellement au fanion. Par exemple, si le fanion est l'octet 01111110, un bit de transparence "0" est inséré après toute séquence de cinq 1 successifs dans la trame.

Exemple :

Données :

01011001111110

Trame :

01111110 010110011111010 **01111110**

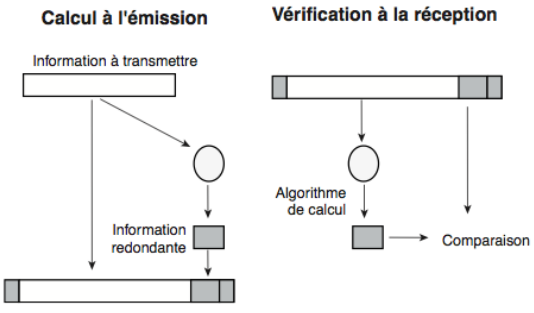
Cette technique est utilisée également en considérant des caractères de délimitation et des caractères de transparence.

L'avantages des fanions est qu'ils permettent de trouver toujours la synchronisation et d'envoyer des trames de tailles quelconques.

5.2 Détection et Correction d'erreurs

D'une manière générale on doit, lors d'une transmission de données, s'assurer que les données reçues n'ont pas été altérées durant la transmission. Plusieurs facteurs peuvent modifier le contenu des données tel que les interférences causées par des rayonnements électromagnétiques ou la distorsion des câbles de transmissions.

Les méthodes de protection exploitent la redondance de données en ajoutant des bits de contrôle aux bits de données. Les bits de contrôle sont calculés, au niveau de l'émetteur, par un algorithme spécifié dans le protocole à partir du bloc de données. À la réception, on exécute le même algorithme pour vérifier si la redondance est cohérente. Si c'est le cas, on considère qu'il n'y a pas d'erreur de transmission et l'information reçue est traitée ; sinon, on est certain que l'information est invalide.



Plusieurs méthodes de protection contre les erreurs peuvent être utilisées :

5.2.1 Duplication de données

Un exemple des bits de contrôle est la duplication des bits transmis (détection par répétition). Le message codé est un double exemplaire du message initial, le récepteur sait qu'il y a eu erreur si les exemplaires ne sont pas identiques, il demande alors la retransmission du message. Si la même erreur se passe sur les deux exemplaires, l'erreur ne sera pas détectée.

Si on envoie le message en trois exemplaires, le récepteur pourra même corriger l'erreur en prenant les valeurs des deux copies identiques sans demander la retransmission de l'émetteur.

5.2.2 Code de contrôle de parité

C'est un code dans lequel un bit (le bit de parité) est ajouté au mot initial pour assurer la parité. Son rendement est faible lorsque k est petit.

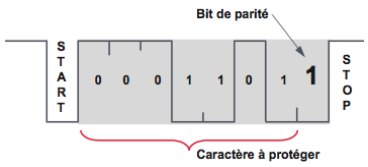
Exemple : Transmission de caractères utilisant un code de représentation (le code ASCII sur 7 bits).

Lettre	Code ASCII	Mot codé (parité paire)	Mode codé (parité impaire)
E	1010001	10100011	10100010
V	0110101	01101010	01101011
A	1000001	10000010	10000011

Ce code est capable de détecter toutes les erreurs en nombre impair mais il ne détecte pas les erreurs en nombre pair. Il permet de détecter une erreur de parité, mais ne permet pas de la localiser.

5.2.2.1 Parité verticale

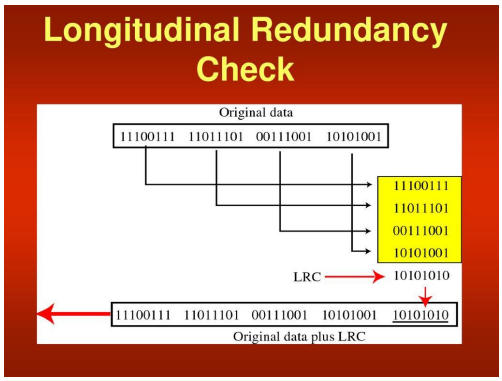
À chaque caractère on rajoute un bit (bit de redondance verticale ou bit de parité, VRC :Vertical Redondancy Check)



5.2.2.2 Parité longitudinale

A chaque bloc de caractère, on ajoute un champ de contrôle supplémentaire (LRC : Longitudinal Redondancy Check)

La parité longitudinale était initialement utilisée pour les bandes magnétiques pour compléter la détection des erreurs de parité verticale.

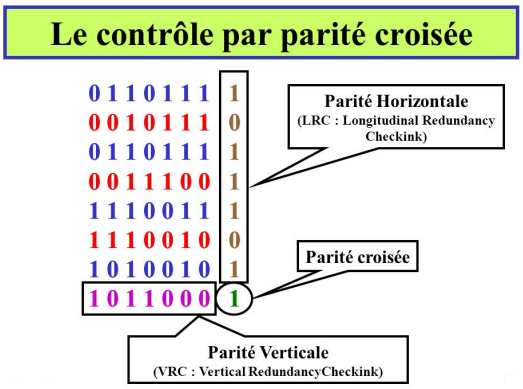


5.2.2.3 Parité longitudinale et verticale

Le bloc de données est disposé sous une forme matricielle ($k = a \bullet b$). On applique la parité sur chaque ligne et chaque colonne. On obtient une matrice $(a+1, b+1)$. Un caractère le LRC est ajouté au bloc transmis. Chaque bit du caractère LRC correspond à la parité des bits de chaque caractère de même rang : le premier bit du LRC est la parité de tous les 1^{er} bits de chaque caractère, le second de tous les 2^e bits... Le caractère ainsi constitué est ajouté au message. Le LRC est lui-même protégé par un bit de parité (VRC).

	H	E	L	L	O	*LRC →*
bit 0	0	1	0	0	1	0
bit 1	0	0	0	0	1	1
bit 2	0	1	1	1	1	0
bit 3	1	0	1	1	1	0
bit 4	0	0	0	0	0	0
bit 5	0	0	0	0	0	0
bit 6	1	1	1	1	1	1
VRC ↓	0	1	1	1	1	0

1001000	0	1000101	1	1001100	1	1001100	1	1001111	1	1000010	0
H		E		L		L		O		LRC	



5.2.3 Codes polynomiaux

La méthode des codes polynômiaux (ou le CRC : Cyclic Redondant Coding) est la méthode la plus utilisée pour détecter des erreurs groupées. Avant la transmission, on ajoute des bits de contrôle. Si des erreurs sont détectées à la réception, il faut retransmettre le message.

Dans ce code, une information de n bits est considérée comme la liste des coefficients binaires d'un polynôme de n termes, donc de degré $n-1$.

Exemple :

- $1101 \rightarrow x^3 + x^2 + 1$
- $110001 \rightarrow x^5 + x^4 + 1$

– $11001011 \rightarrow x^7 + x^6 + x^3 + x + 1$

Pour calculer les bits de contrôle, on effectue un certain nombre d'opérations avec ces polynômes à coefficients binaires. Toutes ces opérations sont effectuées modulo 2. C'est ainsi que, dans les additions et dans les soustractions, on ne tient pas compte de la retenue : Toute addition et toute soustraction sont donc identiques à une opération XOR. Par exemple :

	1	0	0	1	1	0	1	1
+	1	1	0	0	1	0	1	0
=	0	1	0	1	0	0	0	1

	1	1	1	1	0	0	0	0
-	1	0	1	0	0	1	1	0
=	0	1	0	1	0	1	1	0

La source et la destination choisissent un même polynôme $G(x)$ dit générateur car il est utilisé pour générer les bits de contrôle (Checksum).

L'algorithme pour calculer le message à envoyer est le suivant. Soit $M(x)$ le polynôme correspondant au message original, et soit r le degré du polynôme générateur $G(x)$ choisi :

– Multiplier $M(x)$ par x^r, ce qui revient à ajouter r zéros à la fin du message original

– Effectuer la division suivante modulo 2 :

$$\frac{M(x)x^r}{G(x)} = Q(x) + R(x)$$

– Le quotient $Q(x)$ est ignoré. Le reste $R(x)$ (Checksum) contient r bits (degré du reste $r-1$). On effectue alors la soustraction modulo 2 :

$$M(x).x^r - R(x) = T(x)$$

Le polynôme $T(x)$ est le polynôme cyclique : c'est le message prêt à être envoyé. Le polynôme cyclique est un multiple du polynôme générateur $T(x) = Q(x).G(x)$

A la réception, on effectue la division suivante :

$$\frac{T(x)}{G(x)}$$

– Si le reste = 0, il n'y a pas d'erreur

– Si le reste $\neq 0$, il y a erreur, donc on doit retransmettre

En choisissant judicieusement $G(x)$, on peut détecter toute erreur sur 1 bit, 2 bits consécutifs, une séquence de n bits et au-delà de n bits avec une très grande probabilité.

Exemple

Soit à transmettre le message 1011011 en utilisant le polynôme générateur $G(x) = x^4 + x + 1$. On procède comme suit pour calculer le message à transmettre

1. message original = 1011011 $\Rightarrow M(x) = x^6 + x^4 + x^3 + x^1 + 1$
2. $G(x) = x^4 + x + 1$
3. $M(x).x^4 = x^{10} + x^8 + x^7 + x^5 + x^4$
4. Calculer $R(x)$ par division polynomiale

x^{10}	+	x^8	+	x^7	+	x^5	+	x^4	x^4	+	x	+	1	
x^{10}	+	x^7	+	x^6					x^6	+	x^4	+	x^2	
x^8	+	x^6	+	x^5	+	x^4								
x^8	+	x^5	+	x^4										
x^6														
x^6	+	x^3	+	x^2										
		x^3	+	x^2										

5. $R(x) = x^3 + x^2 = (1100)_2$
6. Le message à envoyer $T(x) = M(x).x^r - R(x) = x^10 + x^8 + x^7 + x^5 + x^4 - x^3 - x^2 = (10110111100)_2$

À la réception, un calcul semblable s'effectue sur le mot reçu, mais il faut, ici, que le reste soit nul. Dans le cas contraire, c'est qu'une erreur est survenue en cours de route.

Codes polynomiaux utilisés

Les principaux polynômes générateurs (diviseurs) sont :

- LRCC-8 : $x^8 + 1$
- LRCC-16 : $x^{16} + 1$
- CRC 12 : $x^{12} + x^{11} + x^3 + x^2 + x + 1$
- CRC 16 Forward : $x^{16} + x^{15} + x^2 + 1$
- CRC 16 Backward : $x^{16} + x^{14} + x + 1$
- CRC CITT Forward : $x^{16} + x^{12} + x^5 + 1$
- CRC CITT Backward : $x^{16} + x^{11} + x^4 + 1$

Exemple: 6 bits de données: 110101 , Polynôme générateur 101 : $x^2 + 1$

```
1 1 0 1 0 1 0 0 | 1 0 1
1 0 1           |-------
-------         | 1 1 1 0 1 1
  0 1 1 1       |
  1 0 1         |
  -------       |
    0 1 0 0     |
    1 0 1       |
    -------     |
      0 0 1 1 0 |
        1 0 1   |
        -------   |
          0 1 1 0 |
            1 0 1 |
            -------
Reste= 0 1 1
```

Démonstration de technique de détection des erreurs de transmission CRC :

Exercice :

Soit le message $\mathbf{x^7 + x^5 + 1}$ que nous voulons transmettre à une autre station en utilisant le polynôme générateur $P(x) = G(x) = x^3 + 1$.

1. Quelle est la longueur du FCS (champ du contrôle d'erreur) ?
2. Quel est le message effectivement émis sur la ligne (DATA + FCS) ?
3. Montrer à l'aide d'exemples comment peut-on détecter une erreur à la réception. Autrement, donner deux cas ; cas 1 (réceptions message correct), et cas 2 (réception d'un message erroné).

Solution :

1. Puisque le polynôme générateur P prend la forme $P(x) = x^3 + 1$, donc, la longueur de FCS qui est égale au degré de polynôme générateur vaut 3. Egalement, la longueur de CRC est 4 (4 = n +1, étant donné que n est le degré du polynôme générateur). $C(x) = P(x) = 1 * x^3 + 0* x^2 + 0* x^1 + 1 * x^0$, d'où CRC = **1001**.

2. Le message effectivement émis sur la ligne :

Pour calculer le message émis sur la ligne, il faut passer par les étapes suivantes :

a) Soit M le message d'origine ou la séquence de bits à envoyer.
   Sachant que $M(x) = x^7 + x^5 + 1$, alors M = 10100001. CRC = 1001.

b) On multiplie M par 2^3 (ajouter 3 '0' à la suite de la séquence M. 3 est le degré du polynôme générateur. Soit M'= 10100001 000

c) On divise M'par C en utilisant le OU exclusif

d) Le reste obtenu correspond à FCS

e) Le message à envoyer est : **M + FCS**
   M = 10100001
   CRC = 1001

```
10100001000
1001
001100
 1001
 01010
  1001
  001110
    1001
    01110
     1001
     01110
      1001
      0111
FCS = 111
```

Le reste de la division de M' / CRC est 111. D'où, FCS = 111.

Donc le message émis est M + FCS = 10100001 111

1. Montrer à l'aide d'exemples comment peut-on détecter une erreur à la réception.

   Cas 1 : Réception par la station destinataire d'un message correct composé de séquence de bits à envoyer (data + FCS). Pour s'assurer de l'intégrité du message reçu, elle divise la série binaire (M + FCS) par le code CRC.

```
10100001111
1001
001100
 1001
 01010
  1001
  001111
    1001
    01101
     1001
     01001
      1001
      0000
```

   Reste 0, donc **pas d'erreurs**.

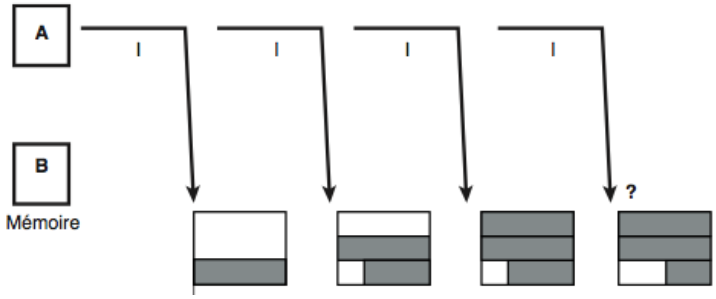
Cas 2 : Réception par la station destinataire d'un message erroné. Le message reçu est 10100001110.

```
1010000 1110
1001
0011 00
  1001
  0101 0
    1001
    0011 11
      1001
      0110 0
        1001
        0 101
```

Reste <> 0, donc **erreurs de transmission** > **retransmission après expiration de temporisateur**

5.3 Contrôle de flux

Dans une transmission d'information d'un émetteur A vers un récepteur B, si l'émetteur produit les données à une vitesse nettement supérieure à la vitesse de consommation du récepteur, ce dernier sera engorgé (saturé ou surchargé) et les informations émises seront perdues. Pour résoudre ce problème, on peut penser à doter le récepteur d'une mémoire tampon lui permettant de stoker les messages en attendant leur traitement. On peut constater rapidement que quelque soit la taille de la mémoire utilisée, elle peut être saturée.



Le contrôle de flux sert à mettre en place un mécanisme de contrôle du rythme d'envoi des informations vers le récepteur. Il peut être réalisé par plusieurs méthodes :

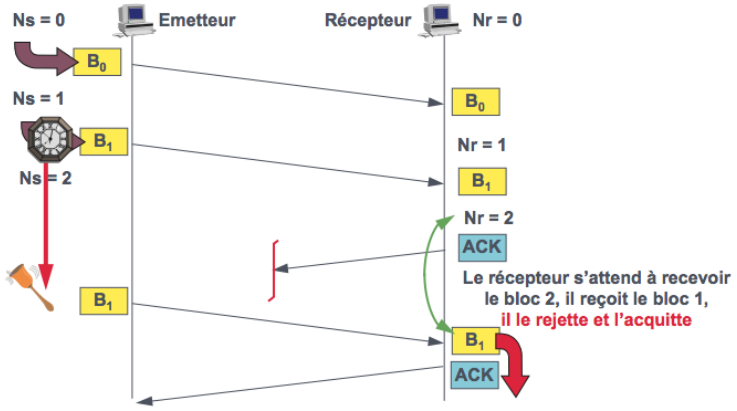
5.3.1 Envoyer et attendre (Sent and Wait)

Après l'envoi d'un bloc d'information, L'émetteur s'arrête dans l'attente d'un accusé de réception. À la réception de l'acquittement, noté ACK pour Acknowledge, l'émetteur envoie le bloc suivant.

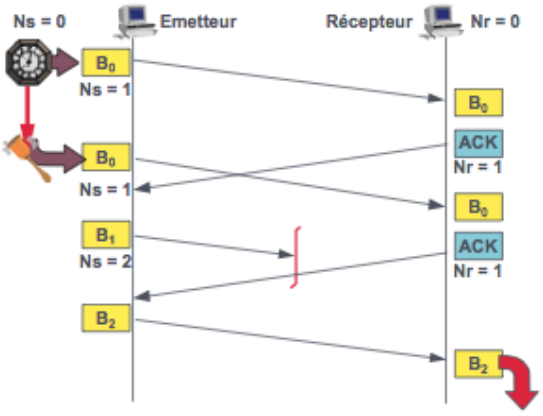
En cas d'erreur de transmission, le bloc reçu est rejeté. Le bloc est dit perdu, il n'est pas acquitté. L'émetteur reste alors en attente. Pour éviter un blocage de la transmission, à l'émission de chaque bloc de données, l'émetteur arme un temporisateur (Timer). À l'échéance du temps imparti (Time Out), si aucun accusé de réception (ACK) n'a été reçu, l'émetteur retransmet le bloc non acquitté.

Une difficulté survient si la perte concerne l'ACK. En effet, bien que les données aient été correctement reçues, l'émetteur les retransmet sur temporisation. Les informations sont ainsi reçues 2 fois. Pour éviter la duplication des données, il est nécessaire d'identifier les blocs. À cet effet, l'émetteur et le récepteur entretiennent des compteurs Ns (Ns, Numéro émis, s pour send) et Nr (Numéro du bloc à recevoir, r pour receive).Les deux compteurs sont initialisés à zéro. Le contenu du compteur Ns est transmis avec le bloc, le récepteur compare ce numéro avec le contenu de son compteur Nr. Si les deux valeurs sont identiques

le bloc est réputé valide et accepté. Si les valeurs diffèrent, le bloc reçu n'est pas celui attendu. Il est rejeté et acquitté s'il correspond à un bloc déjà reçu.



Dans les cas où les délais de consommation sont plus importants, les données peuvent ne pas être acquittées à temps. Par exemple, si A transmet un bloc B0 et B se tarde dans son traitement, A va retransmettre B0 avant de recevoir l'acquittement. Si A transmet un nouveau bloc B1 et se perd, il va considérer l'acquittement du deuxième B0 comme un acquittement de B1.



Pour éviter cette confusion d'interprétation, il est aussi nécessaire de numéroter les ACK.

Le temps d'attente des acquittements rend la méthode send and wait peu efficace. En plus, il est unidirectionnel.

5.3.2 Fenêtre d'anticipation

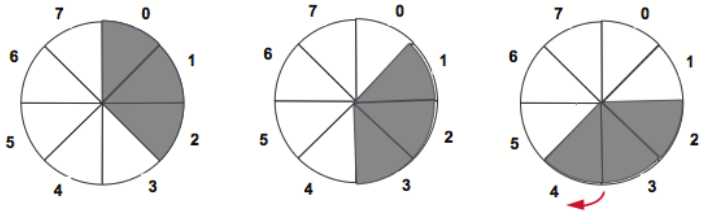
Pour améliorer le protocole précédent et réduire le temps d'attente des acquittements, on émet plusieurs blocs sans attendre les ACK, ce processus se nomme anticipation. Ainsi,

un acquittement 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 W, appelée fenêtre (Window). Le principe est d'autoriser l'émetteur à envoyer les trames de numéro de séquence compris entre le numéro r de la prochaine trame attendue (communiqué par le récepteur) et $r + W - 1$:

$$r \leq Ns \leq r + W - 1$$

Remarque : $W = 1$ dans le cas d'une procédure Send-and-Wait.

Pour pouvoir réenvoyer les trames en cas d'erreur, l'émetteur met les blocs non acquittés dans W mémoires tampons. À la réception d'un acquittement d'une trame, l'émetteur libère la mémoire correspondante et envoie une nouvelle trame.



Problème : que se passe-t-il si, de façon temporaire, le récepteur n'est pas prêt à recevoir les W trames d'information de la fenêtre ? L'utilisation d'une fenêtre d'anticipation peut nécessiter la mise en œuvre d'un mécanisme de régulation supplémentaire de type tout-ou-rien. L'idée est d'utiliser une trame de contrôle particulière est utilisée pour indiquer que le récepteur est momentanément dans l'incapacité de continuer à recevoir. L'émetteur recevant cette trame de contrôle cesse immédiatement toute émission (même s'il n'avait pas utilisé toute sa "fenêtre" d'émission). Une autre trame de contrôle est alors nécessaire pour indiquer à l'émetteur que le récepteur est revenu dans un état normal et qu'il est donc prêt à recevoir de nouvelles trames.

5.4 Procédures de gestion des données

Les procédures de gestion de données sont des protocoles de la couche liaison qui mettent en œuvre les techniques précédentes (délimitation des trames, correction des erreurs et contrôle de flux). Elles sont de deux familles :

1. Les procédures orientées caractère : qui fonctionnent généralement à l'alternat (de type send and wait).

2. Les procédures orientées bit : conçues pour les transmissions bidirectionnelles simultanées à hauts débits.

5.4.1 Procédure HDLC

La procédure HDLC (High-level Data Link Control) est une procédure orientée bit, développée par IBM et normalisée par l'UIT en 1976. HDLC est une procédure point à point et multipoints en full duplex, utilisant des trames séparées par des fanions de valeur 01111110 (7E). Trois modes peuvent être exploités par HDLC :

1. le mode de réponse normal (Normal Response Mode ou NRM) : la station secondaire doit attendre un ordre explicite du primaire avant de pouvoir émettre.
2. le mode de réponse asynchrone (Asynchronous Response Mode ou ARM) :la station secondaire a le droit d'émettre des données sans attendre l'invitation du primaire. Ce mode de fonctionnement est également connu sous le nom protocole LAP (Link Access Protocol). Il suppose que les deux stations possèdent à la fois le statut primaire et le statut secondaire.
3. le mode de réponse asynchrone équilibré ou symétrique (Asynchronous Balanced Mode ou ABM) : la liaison est obligatoirement point à point ; comme pour LAP, les deux stations possèdent à la fois le statut primaire et le statut secondaire. Ce mode de fonctionnement est connu aussi sous le nom de protocole LAP-B (Link Access Protocol-Balanced). De nos jours, c'est le seul mode utilisé.

5.4.1.1 Types de trames HDLC

HDLC utilise trois types de trames :

1. les trames d'information ou trames **I** : assurent le transfert de données ;
2. les trames de supervision ou trames **S** (Supervisor) : assurent la transmission des commandes de supervision (accusé de réception...),
3. les trames non numérotées ou trames **U** (Unnumbered) : supervisent la liaison (connexion, déconnexion).

5.4.1.2 Structure de la trame HDLC

La trame HDLC est organisée comme suit :

8 bits	8 bits	8 bits	n bits	16 bits	8 bits
fanion	adresse	contrôle	information	FCS	fanion

- Fanion (flag) : 01111110
  . délimite les trames : toutes les trames doivent commencer et finir par un fanion.
  . permet la synchronisation des trames : toutes les stations rattachées à la liaison doivent rechercher en permanence cette séquence ;
  . un même fanion peut servir de fanion de fermeture pour une trame et de fanion d'ouverture pour la trame suivante ;
  . mécanisme de transparence au fanion par bits de bourrage : en émission, un 0 est inséré dès que cinq 1 consécutifs apparaissent en dehors des champs F ; ces 0 sont enlevés en réception. Si sept 1 apparaissent n'importe où dans une trame, elle est déclarée en erreur.
- Champ d'adresse : permet d'identifier la trame comme étant une commande ou une réponse. En mode ABM, les valeurs que peut prendre ce champ sont prédéfinies. Quatre valeurs sont suffisantes pour distinguer les commandes et les réponses dans les deux sens de transmission (ex : 11000000, 10000000, 11110000 et 1110000).
- Champ de contrôle : il indique le type de trame avec les paramètres nécessaires.
- FCS (Frame Check Sequence) : calculé sur les champs d'adresse, de commande et d'information, à partir du code polynômial V.41 ($x^{16} + x^{12} + x^5 + 1$).

5.4.1.3 Champ de contrôle et formats de trame

Il existe trois formats de trame qui correspondent à des codages différents du champ de contrôle :

	1	2	3	4	5	6	7	8
Information	0	Ns			P/F	Nr		
Supervision	1	0	S	S	P/F	Nr		
Non numérotée	1	1	M	M	P/F	M	M	M

- Ns : numéro de séquence de la trame émise,
- Nr : numéro de la prochaine trame attendue (acquittement dans les données),
- P/F (Poll/Final) : Ce bit est positionné à 1 par le primaire lorsque celui-ci sollicite une réponse immédiate du secondaire.

La signification des trames de type S dépend des deux bits SS selon le tableau suivant :

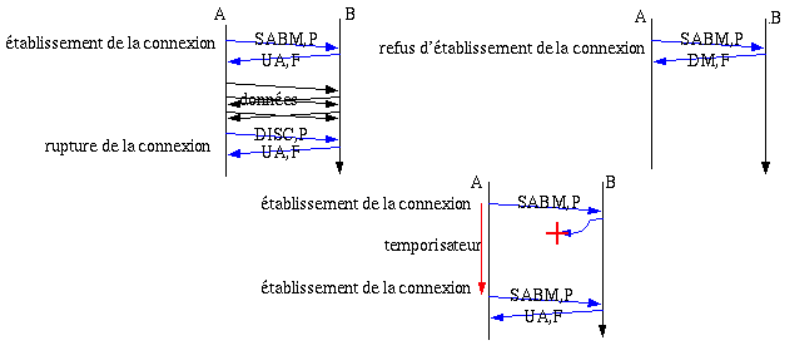
S	S	Commande	Signification
0	0	RR (Receiver Ready)	La station est prête à recevoir la trame numéro Nr et accuse positivement la réception des trames jusqu'à (Nr - 1)
0	1	RNR (Receiver not Ready)	La station n'est pas prête à recevoir des trames mais et accuse positivement la réception des trames jusqu'à (Nr - 1)
1	0	REJ (Reject)	La station rejette les trames à partir du numéro Nr. L'émetteur est obligé de retransmettre (P/F = 1)
1	1	SREJ (Reject)	= REJ mais uniquement pour la trame numéro Nr.

La signification des trames de type U dépend des deux bits M selon le tableau suivant :

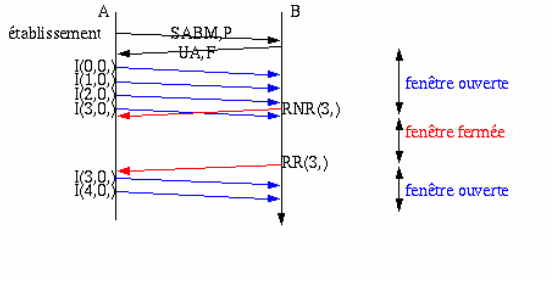
Trame	Commande	Signification
11111100	SABM	Set ABM demande l'établissement en mode ABM
11110000	DM	Disconnect Mode indique que la station se trouve en mode déconnecté
11001010	DISC	Disconnect libère la liaison
11000110	UA	Unnumbered Acknowledge indique la réception et l'acceptation d'une commande non numérotée

5.4.1.4 Exemples des échange

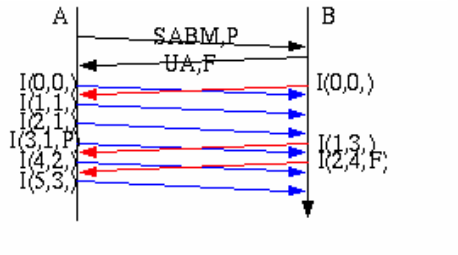
La figure suivante illustre des exemples des échanges entre deux stations utilisant la procédure HDLC pour établir la connexion.



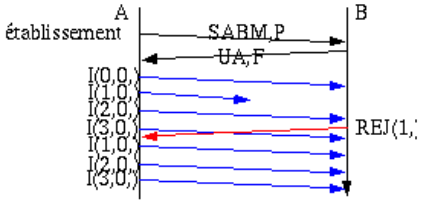
L'exemple suivant, illustre un cas d'échange unidirectionnel.



L'exemple suivant, illustre un cas d'échange bidirectionnel.



L'exemple suivant, illustre un cas d'échange unidirectionnel avec perte de trames.



5.5 Protocole PPP

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 (la liaison concerne typiquement un ordinateur personnel et le fournisseur d'accès à Internet). Il s'agit d'une version très simplifiée d'HDLC qui ne comprend ni contrôle de flux, ni mécanisme de reprise sur erreurs. La structure d'une trame PPP est donnée dans la figure suivante.

Flag	Address	Control	Data PPP	Flag

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 commence par deux octets (le champ protocole), qui identifient le protocole

de niveau supérieur auquel est destinée la trame ; il se termine par un champ FCS dont le mode de calcul est identique à celui d'une trame HDLC. La seule trame transportant des données sur une liaison fiable est une trame U de type UI (Unnumbered Information). Cette trame contient un champ d'informations mais n'est pas numérotée (car il n'y a pas de contrôle de flux). L'absence de mécanisme de reprise sur erreur ne signifie pas que le circuit n'est pas fiable : le champ FCS sert à valider les trames reçues.

PPP comprend trois composants principaux :

- Une méthode pour encapsuler les datagrammes de plusieurs protocoles.
- Un protocole de contrôle du lien "Link Control Protocol" (LCP) destiné à établir, configurer, et tester la liaison de données.
- Une famille de protocoles de contrôle de réseau "Network Control Protocols" (NCPs) pour l'établissement et la configuration de plusieurs protocoles de la couche réseau.

Deux variantes très connues du protocole PPP :

- **PPPoA** (en anglais point-to-point protocol over ATM) est un protocole utilisé par les connexions haut débit ADSL et câble destinées aux particuliers.
- **PPPoE** (en anglais point-to-point protocol over Ethernet) est un protocole d'encapsulation de PPP sur Ethernet. Il permet de bénéficier des avantages de PPP, notamment sa compatibilité avec les protocoles d'authentification (PAP, CHAP, etc.) et le contrôle de la connexion (débit, etc.), sur un réseau Ethernet. Il est beaucoup employé par les connexions haut débit à Internet par ADSL et câble destinées aux particuliers.