

# LIAISON SERIE RS232

## 1 - Généralités

Une liaison série permet de faire communiquer deux appareils : En général un PC et un périphérique pour échanger des informations.

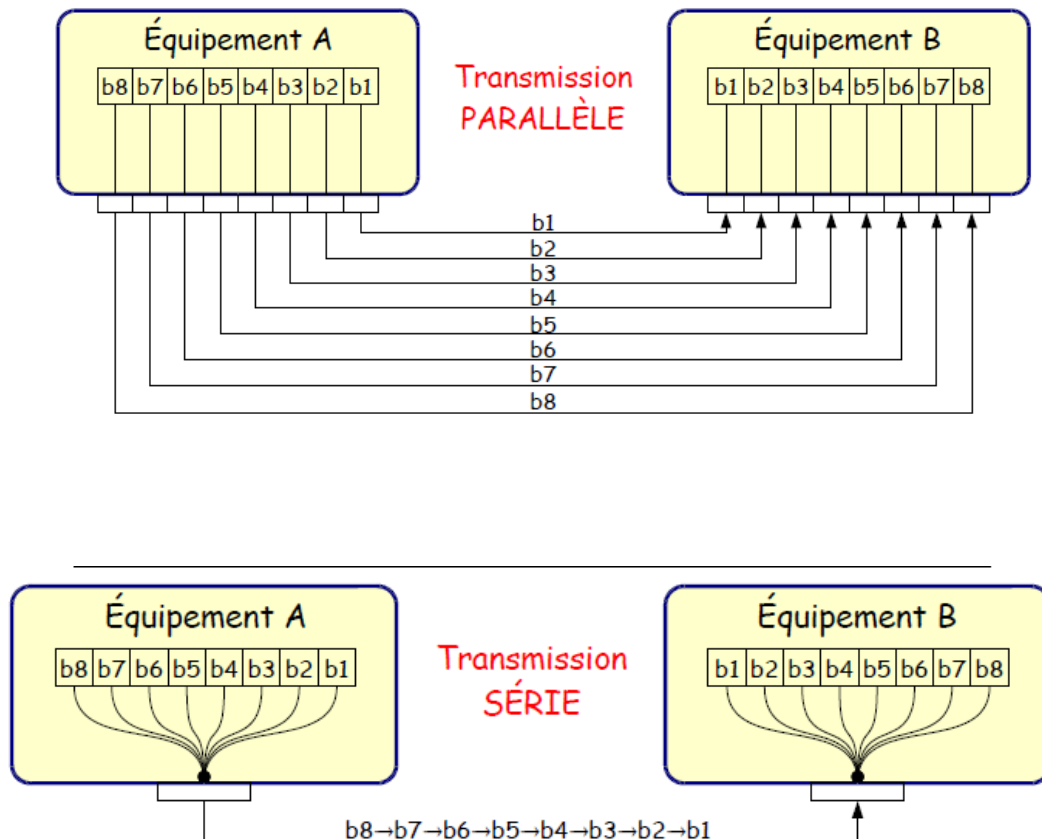
Exemples :

- Systèmes d'affichage
- Appareils de mesure
- GPS
- Scanners optiques
- ...

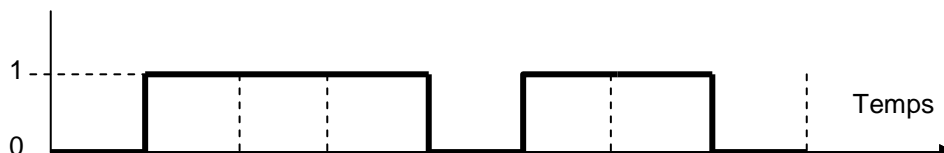
Le terme RS232 est le nom de la norme (Recommended standard 232) associée à ce mode de communication. Le mode de transmission série s'oppose au mode de transmission parallèle

Contrairement à une interface parallèle qui transmet les 8 bits d'un octet simultanément sur des fils séparés, l'interface série transmet les bits d'un octet les uns après les autres sur un même fil.

Cela amène une économie de câble, un fil au lieu de 8, mais nécessite un montage spécialisé pour sérialiser les bits à l'émission et les désérialiser à la réception. Ceci peut se faire avec des registres à décalage. Dans la pratique, des circuits spécialisés, nommés UART se chargent de la transmission, de la réception et du dialogue local avec le microprocesseur hôte. Les données sont alors envoyées bit par bit.



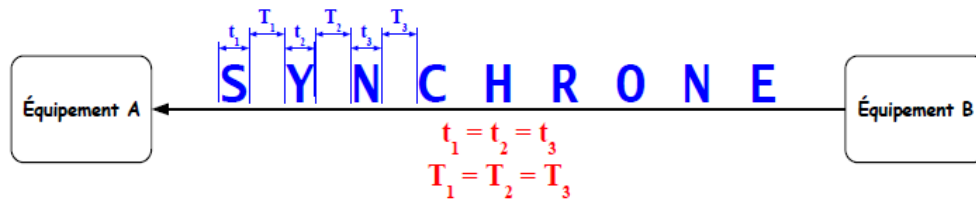
L'état de la ligne de transmission peut être représentée par le chronogramme ci-dessous :



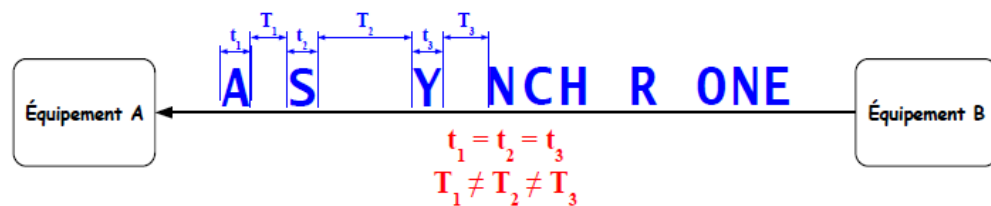
## 2 - Liaison Asynchrone

La transmission de données en série peut se concevoir de deux façons différentes :

► en mode synchrone, l'émetteur et le récepteur possède une horloge synchronisée qui cadence la transmission. Le flot de données peut être ininterrompu.



► en mode asynchrone, la transmission s'effectue au rythme de la présence des données. Les caractères envoyés sont encadrés par un signal *start* et un signal *stop*.



La liaison série RS232 est **une liaison asynchrone**, ce qui veut dire que les horloges des équipements émetteur et récepteur ne sont pas synchronisées, et que les envois de données sont fait à tout moment par l'émetteur.

## 3 - Protocole

Afin que les éléments communicants puissent se comprendre, il est nécessaire d'établir un protocole de transmission. Ce protocole devra être le même pour chaque élément.

Paramètres rentrant en jeu :

► **longueur des données** : 7 bits ou 8 bits

code ASCII ou code ASCII étendu. Les bits sont envoyés en commençant par le bit de poids faible (LSB)

► **vitesse de transmission** : les vitesses varient le plus souvent de 110 bit/s à 19200 bit/s (155200bit/s maxi)( à ne pas confondre avec les bauds qui correspondent au nombre de changement d'état de la ligne par seconde !!)

► **parité** : le mot transmis peut être suivis ou non d'un bit de parité qui sert à détecter les erreurs éventuelles de transmission. Le bit de parité est généré par l'émetteur et est contrôlé par le récepteur. Il existe deux types de parité : paire ou impaire.

Parité Paire : Le nombre de bits (de données et de parité) au niveau logique 1 est pair.

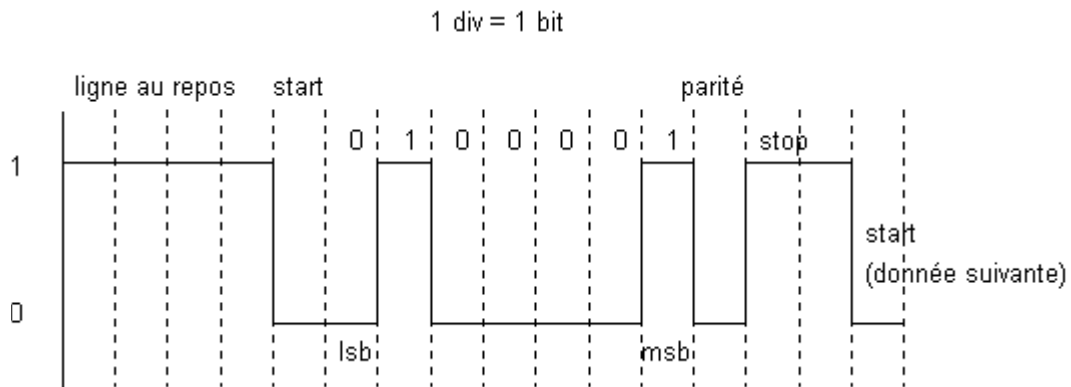
Parité Impaire : Le nombre de bits (de données et de parité) au niveau logique 1 est impair.

Le bit de parité est donc positionné afin de satisfaire aux conditions ci-dessus.

► **bit de start** : la ligne au repos est à l'état 1. Le passage à l'état bas de la ligne va indiquer qu'un transfert va commencer. Cela permet de synchroniser l'horloge de réception.

► **bit de stop** : après la transmission, la ligne est positionnée à un niveau un certains nombre de bit (1 ou 2) afin de spécifier la fin du transfert.

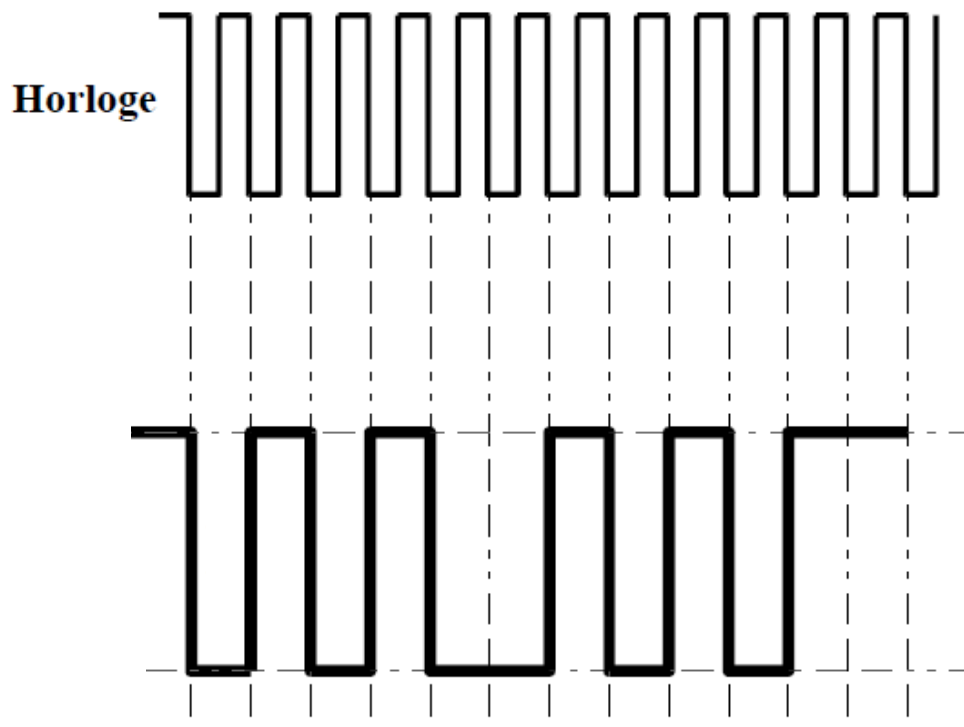
Example :



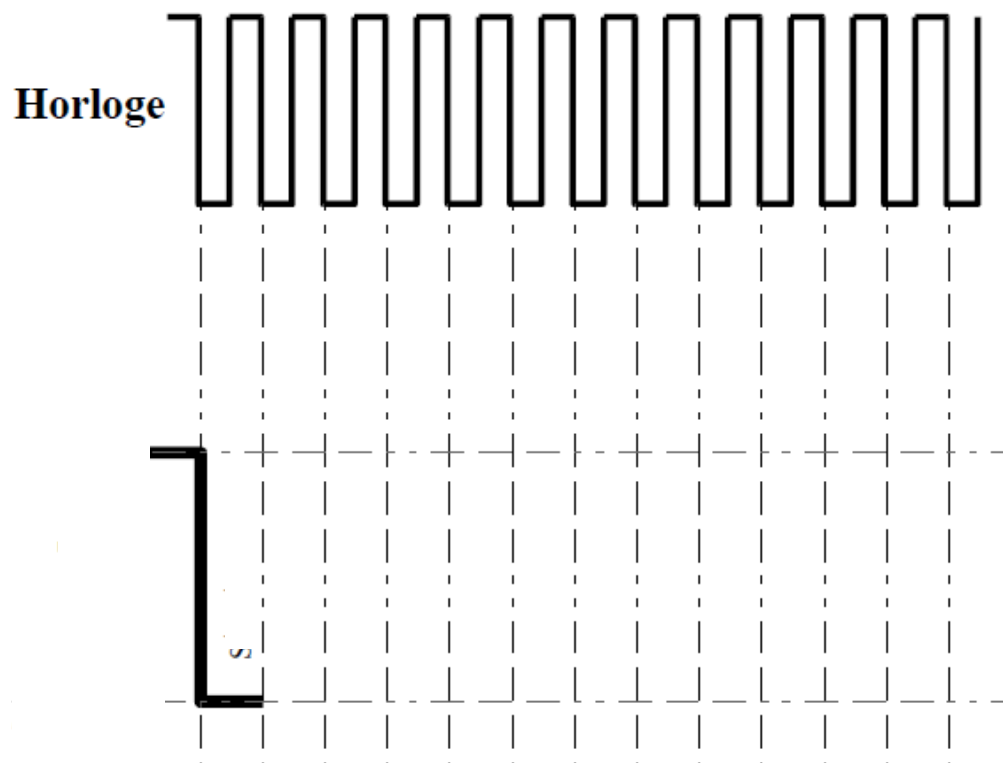
#### **4 - Exercices :**

Quelle vitesse a été sachant qu'une division a une durée de  $104 \mu\text{s}$ .

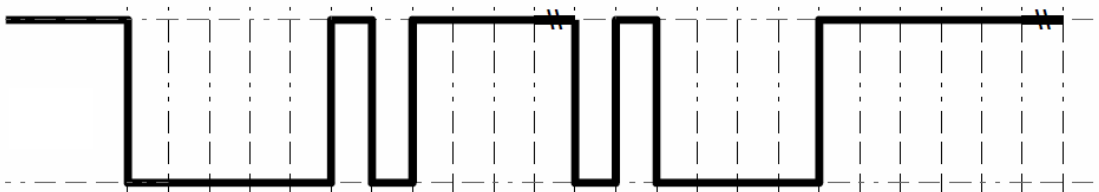
Quelle est la valeur hexadécimale transmise. Les paramètres étant 8 bits de données, 1 bit de stop parité paire.



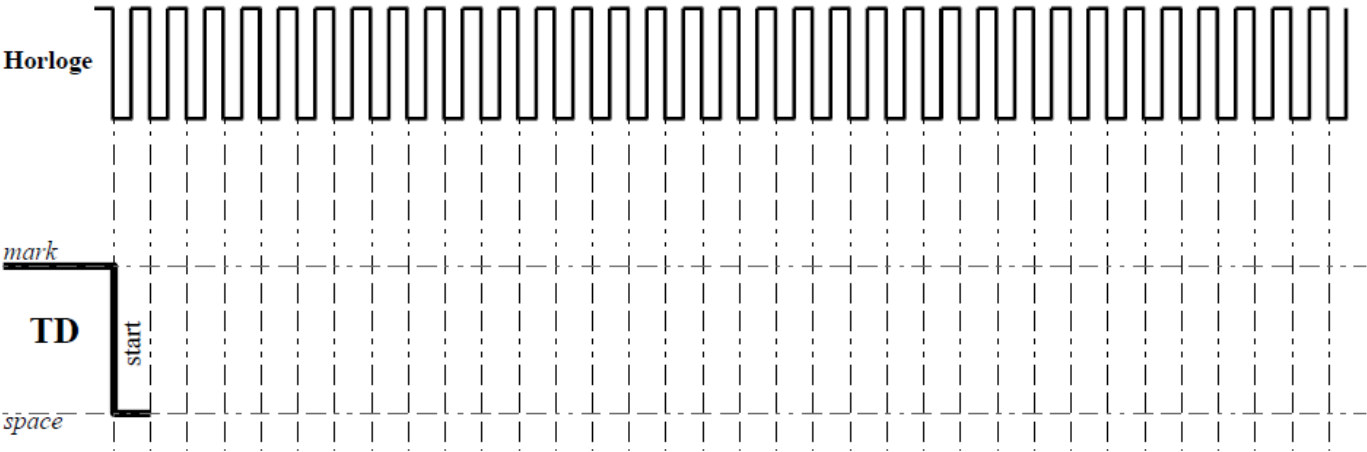
Le format de la transmission est 8 bits de données, 2 bits de stop, parité impaire  
Compléter le chronogramme correspondant à l'émission du code 0x48.



Le format de la transmission est 7 bits de données, 2 bits de stop, pas de parité  
 Quelles sont les valeurs hexadécimales des deux codes transmis ?  
 Quels sont les caractères ASCII correspondants?



Le format de la transmission est 7 bits de données, 1 bit de stop, parité impaire  
Compléter le chronogramme correspondant à l'émission de la chaîne « V24 ».



## 5 La Norme RS232 :

RS signifie 'Recommended Standard' . Dans les années 1960, un comité, actuellement nommé 'Electronic Industries Association' (d'où le EIA232 des années 1990), a développé un standard d'interface de transmission de données en série entre équipements. A l'époque, il était prévu essentiellement pour des communications entre ordinateur et modem. Par la suite, il a été utilisé à d'autres fins comme la transmission de donnée entre des ordinateurs, entre un ordinateur et ces périphériques (imprimante, table traçante, ...), entre un ordinateur et d'autres systèmes équipés de processeur tel les commandes numériques de machines outils, ...

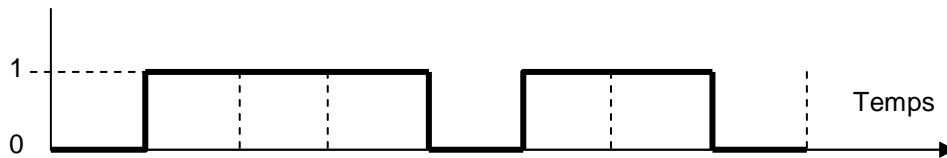
Ce standard définit les niveaux de tensions correspondant aux 1 et aux 0, le brochage des connecteurs, la fonction de chacun des signaux et un protocole d'échange des informations.

La norme a été reprise en Europe sous l'appellation V24 pour les caractéristiques électriques ou V28 pour les caractéristiques mécaniques (connecteurs brochage)

### ► Caractéristiques électriques

Le chronogramme représente les niveaux logiques 1 ou 0 de l'information, ces valeurs sont associées à des niveaux de tension électrique

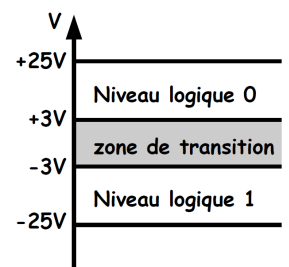
#### Exemple :



#### Tensions :

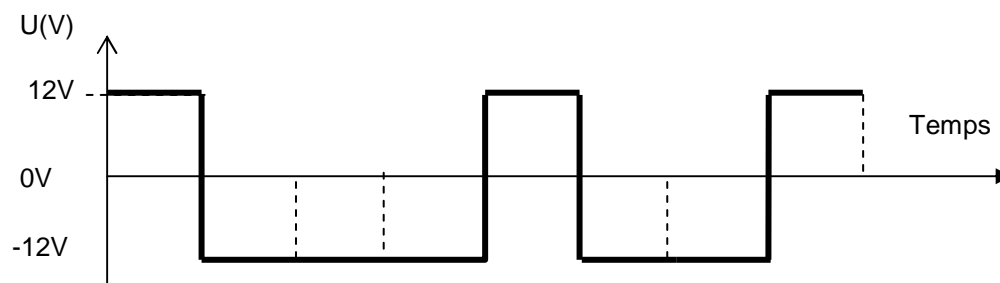
Les tensions représentant les 1 et 0 sont relatives à la masse (0V) commune qu'il y a entre les 2 équipements.

Tension	Etat
-15V à -3V	1 logique,
-3V à 3V	zone interdite, afin éliminer les problèmes dus aux bruits sur la ligne
3V à 15V	0 logique



En principe on utilise les niveaux +12V pour un 0 logique et -12V pour un 1 logique.

La transmission représentée par le chronogramme peut être représentée par le diagramme suivant :



## ► Caractéristiques électriques

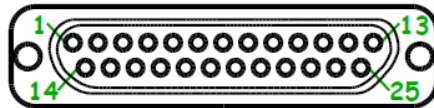
### - Longueur de câbles :

La longueur théorique du câble est de 15 mètres.

### - Brochages :

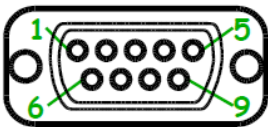
La norme est définie par 21 signaux et représentée par un connecteur 25 broches (type DB25) mais on ne les utilise tous pratiquement jamais.

Dans les cas les plus fréquents, on n'utilise que 7 de ces lignes ( 1 à 8 ). Ce qui conduit à employer un connecteur 9 broches (type DB9).



**Connecteur DB 25 mâle**

N° de broche	Description RS232	Type	Origine
1	PG : Protective Ground		
2	TD : Transmitted Data	donnée	ETTD
3	RD : Received Data	donnée	ETCD
4	RTS : Request To Send	contrôle	ETTD
5	CTS : Clear To Send	contrôle	ETCD
6	DSR : Data Set Ready	contrôle	ETCD
7	SG : Signal Ground	contrôle	
8	DCD : Data Carrier Detect	contrôle	ETCD
15	TC : Transmit Clock (DCE)	synchro	ETCD
17	RC : Received Clock (DCE)	synchro	ETCD
20	DTR : Data Terminal Ready	contrôle	ETTD
22	RI : Ring Indicator	contrôle	ETCD
24	TC : Transmit Clock (DTE)	synchro	ETTD



**Connecteur DB 9 mâle**

N° de broche DB9	N° de broche DB25	Signal
3	2	TD
2	3	RD
7	4	RTS
8	5	CTS
6	6	DSR
5	7	SG(masse)
1	8	DCD
4	20	DTR
9	22	RI

## **Connecteur 9 broches males (ordinateur)**

Broche	Signal	Type	Utilisation
1	CD	Entrée	Carrier Detect: le passage au 0 logique annonce que l'autre équipement reçoit une réponse.
2	RD	Entrée	Received Data: donnée reçue
3	TD	Sortie	Transmitted Data: donnée émise.
4	DTR	Sortie	Data Terminal Ready: le passage au 0 logique indique que le PC souhaite communiquer.
5	SG		Signal Ground: masse de référence des signaux (0V)
6	DSR	Entrée	Data Set Ready: le passage au 0 logique annonce que l'équipement opposé est prêt.
7	RTS	Sortie	Request To Send: le passage au 0 logique demande à l'équipement opposé de se tenir prêt à recevoir.
8	CTS	Entrée	Clear To Send: le passage au 0 logique annonce que l'équipement opposé est prêt à recevoir.
9	RI	Entrée	Ring Indicator: le passage au 0 logique annonce que le modem reçoit un appel.

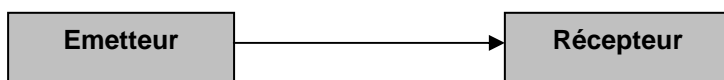
## **Connecteur 9 broches femelles (périphérique)**

Broche	Signal	Type	Utilisation
1	CD	Sortie	Carrier Detect: le passage au 0 logique indique à l'ordinateur que l'on reçoit une réponse.
2	TD	Sortie	Transmitted Data: donnée émise. Lorsque l'équipement est en attente, la sortie est au 1 logique.
3	RD	Entrée	Received Data: donnée reçue
4	DTR	Entrée	Data Terminal Ready: le passage au 0 logique annonce que l'ordinateur souhaite communiquer
5	SG		Signal Ground: masse de référence des signaux (0V)
6	DSR	Sortie	Data Set Ready: le passage au 0 logique indique que l'on est prêt.
7	CTS	Entrée	Clear To Send: le passage au 0 logique annonce que l'ordinateur nous demande d'être prêt recevoir.
8	RTS	Sortie	Request To Send: le passage au 0 logique indique que l'on est prêt à recevoir.
9	RI	Sortie	Ring Indicator: le passage au 0 logique indique à l'ordinateur que l'on reçoit un appel.

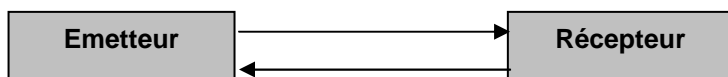
## **6 – Modes de Communication :**

Il existe plusieurs mode de communications :

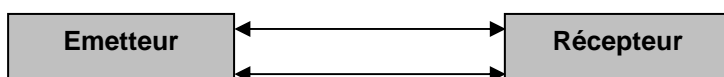
liaison simplex : la transmission ne se fait que dans un seul sens. ( ex : radio )



liaison half duplex : la transmission peut se faire dans les deux sens mais pas simultanément. ( ex : talkie walkie )



liaison full duplex : la liaison peut se faire simultanément dans les deux sens. ( ex : téléphone )





## 7 – Contrôle de FLux :

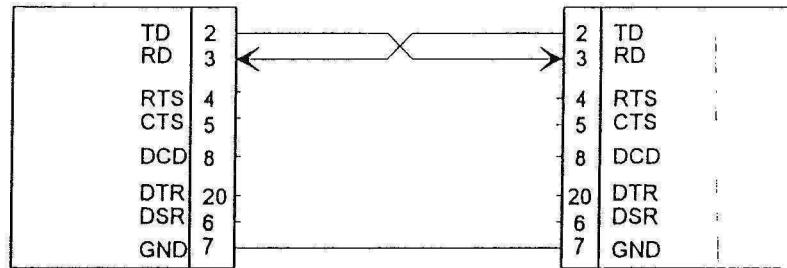
La mise en place d'un contrôle de flux a pour objectif d'éviter des pertes d'information dans les cas où la vitesse de traitement des informations côté récepteur n'est pas suffisante.

Lorsque le tampon du récepteur est plein, la transmission s'interrompt et reprend plus tard.

### Pas de Contrôle de Flux :

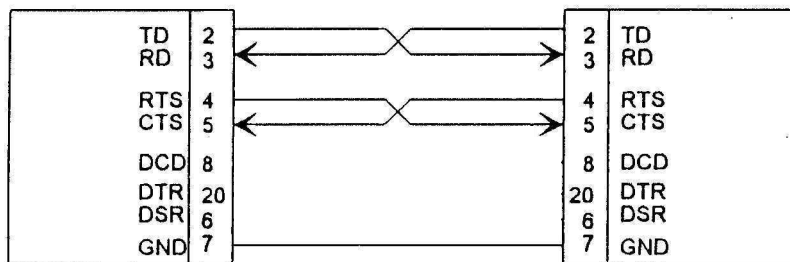
Un câble trois conducteur (type croisé) suffit.

Si le tampon du récepteur est plein, des données sont perdues. C'est une solution peu sûre mais qui peut être suffisante lorsque les volumes échangés sont faibles.



### Contrôle de Flux Matériel (CTS/RTS)

L'émetteur envoie des données. Le récepteur les stocke dans une mémoire tampon. Lorsque cette mémoire atteint un seuil de remplissage défini, le récepteur supprime son signal CTS (passage au 1 logique). L'émetteur arrête immédiatement d'envoyer des données. Le récepteur continue de traiter les données qu'il a dans sa mémoire tampon. Lorsque sa mémoire tampon arrive au seuil espace libre suffisant, il remet le signal CTS (passage au 0 logique). L'émetteur se remet à envoyer des données. Le cycle recommence jusqu'à ce que toutes les données aient été envoyées.



### Contrôle de Flux Logiciel (XON/XOFF)

L'émetteur envoie des données. Le récepteur les stocke dans une mémoire tampon. Lorsque cette mémoire atteint un seuil de remplissage défini, le récepteur envoie le code XOFF (caractère de code décimal 17) à l'émetteur. L'émetteur arrête immédiatement d'envoyer des données. Le récepteur continue de traiter les données qu'il a dans sa mémoire tampon. Lorsque sa mémoire tampon arrive au seuil espace libre suffisant, il envoie le code XON (caractère de code décimal 19) à l'émetteur. L'émetteur se remet à envoyer des données. Le cycle recommence jusqu'à ce que toutes les données aient été envoyées.

Le logiciel de communication doit alors pendant la transmission vérifier en permanence la présence ou non de l'envoi du caractère XOFF. Un câble 3 conducteurs est suffisant.