

Les communications série

cours n°5
LI326

Plan

- Principe de la communication série
- Le type de bus séries
- Le bus rs232
- Le cas du pic16f877

Principe de la communication série



Pour "transmettre" des données entre deux ordinateurs (ou périphérique) par des fils, on dispose de 2 modes:

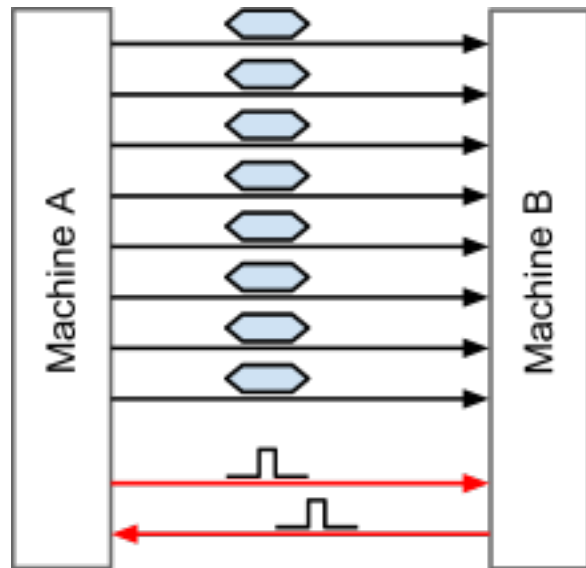
- mode parallèle
 - tous les bits de données d'un même caractère sont envoyés en même temps
- mode série
 - les bits de données sont envoyés l'un après l'autre

Actuellement :

le mode série est le plus utilisé pour la communication avec les E/S, le mode parallèle est confiné sur le circuit imprimé.

Différence : Série / Parallèle

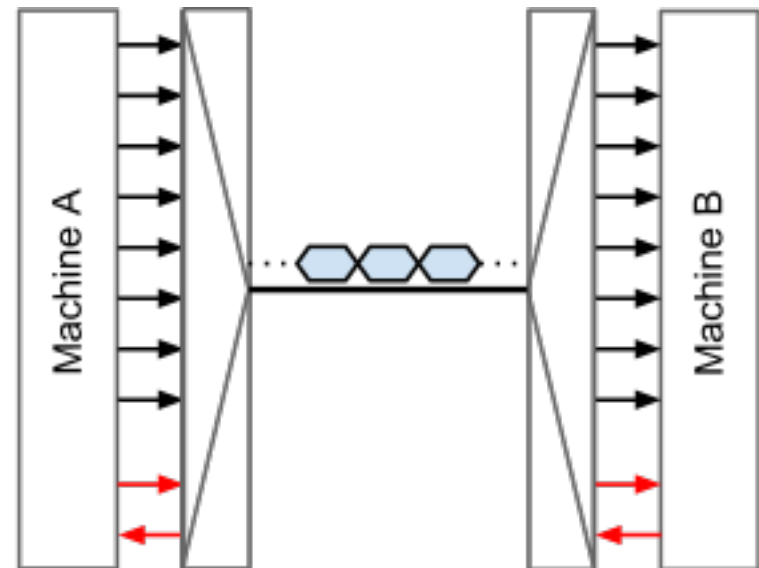
Transmission parallèle



Les bits sont envoyés en parallèle
les caractères sont envoyés en série.

a priori plus simple, mais tous les signaux doivent arriver en même temps, c'est donc cher et difficile pour les grandes distances.

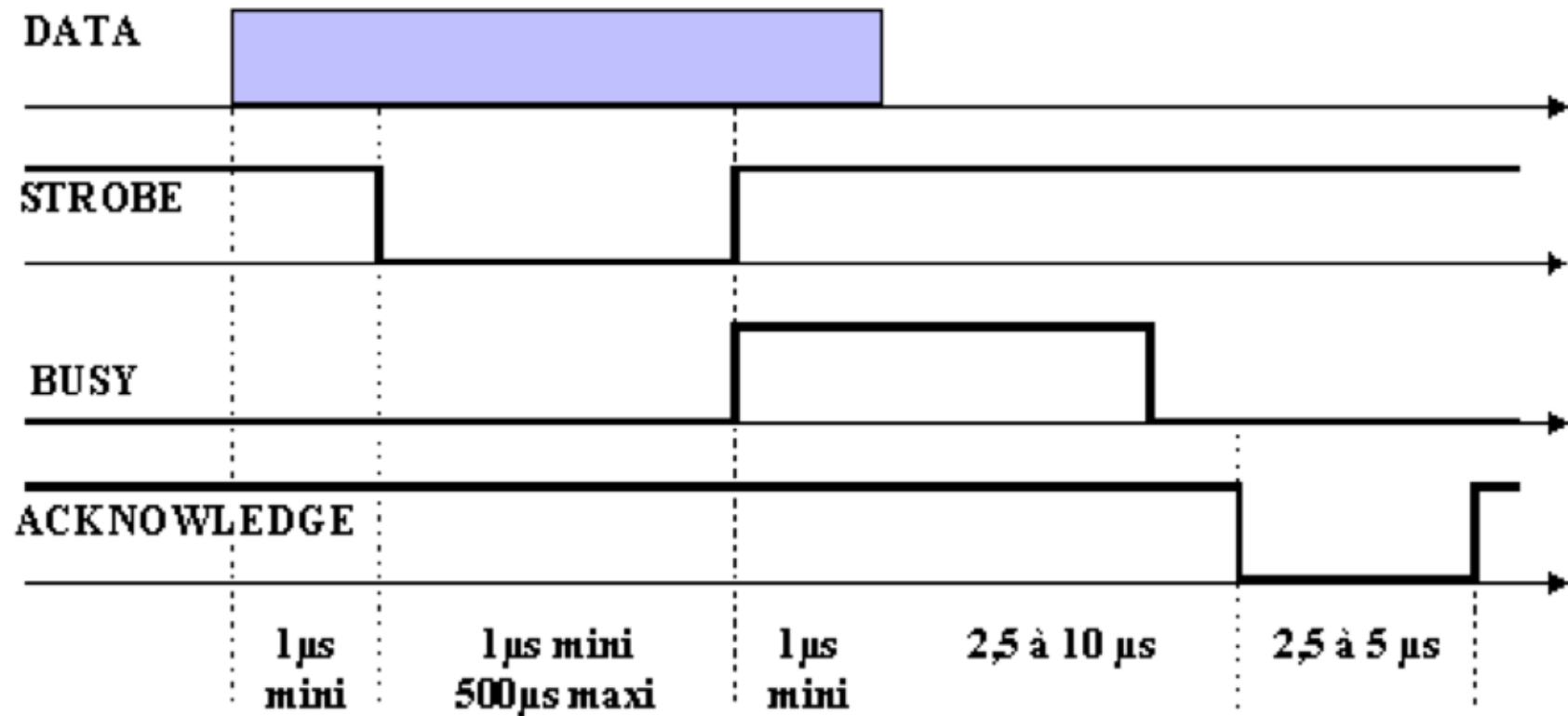
Transmission série



Les bits de chaque caractères sont envoyés en série.

nécessite un sérialiseur/désérialiseur, mais tous les bits arrivent dans l'ordre cela semble plus long, mais on peut augmenter la fréquence.

Port Parallèle EPP (chronogramme)



Les bus séries

- Il existe une très grande variété car aucun bus n'est universel.
- Il faut choisir en fonction :
 - de la distance entre les équipements (cm, m, km)
 - du nombre d'équipements à relier
 - du débit de données (contrôle ou data)
 - de la consommation autorisée (pile/secteur)
 - de la fiabilité nécessaire (bruit)
 - de la maintenabilité (hotplug)
 - des contraintes temporelle (QoS)
 - du catalogue disponible
 - du coût (en générale, la bonne solution est trop chère)
 - etc...
- Chaque bus existe en plusieurs versions, en général compatibles entre elles mais avec un rapport débit / distance différent.

Quelques bus série pour μ C

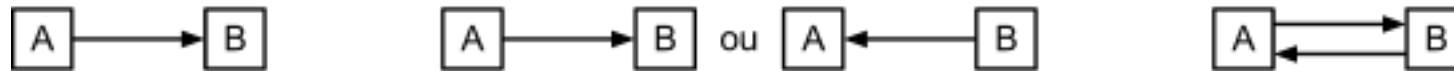
Bus	Distance (mètres)	Débit (bits/sec)	Avantage	Usage
CAN	100 - 10 000	33k - 5k	distance, sûr	domotique
USB	25	1.5 - 480M	débit	périphérique informatique
rs232	15	100k	simple	périphérique informatique
I2C ls	100	100k	coût, protocole	domotique
I2C hs	0.5	3.4M	”	périphérique électronique
SPI	0.1	10M	débit - coût	périphérique électronique

- Le pic16f877 propose RS232, I2C et SPI natifs
- La carte pic des TME propose RS232(via USB), I2C, SPI

Différences technologiques

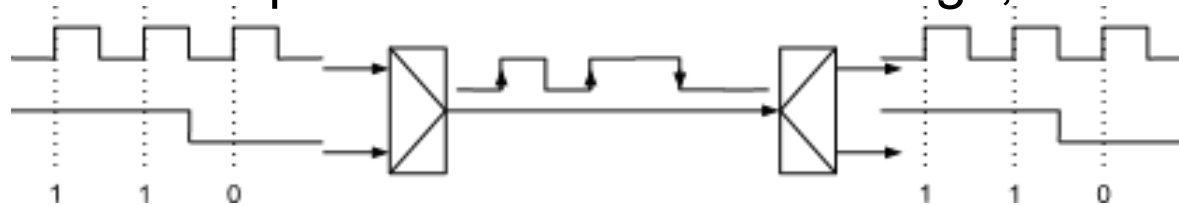
- Half duplex ou Full duplex

- transit dans un sens, les deux sens séparément ou en même temps.



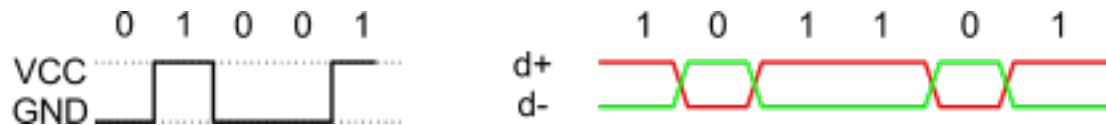
- Horloge et Data mélangés ou séparés

- 1 même fil pour les données et l'horloge, ou 2 fils.



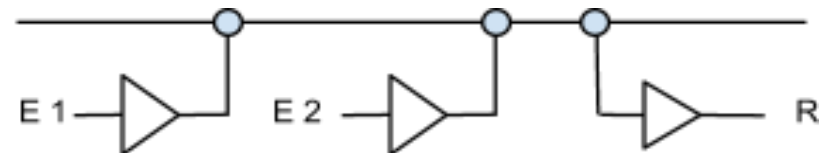
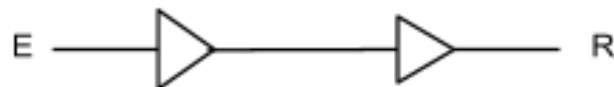
- Signal différentiel ou simple

- une donnée utilise 2 fils de valeurs opposées ou 1 seul valant 0 ou 1.



- Signal point-à-point ou bussé

- 1 seul pilote par fil ou plusieurs pilotes par fil



Différences technologiques

- RS232

- full duplex,
- pas d'horloge
- 2 data (3 fils min : RX, TX, GND),
- signal non différentiel
- point à point

- I2C

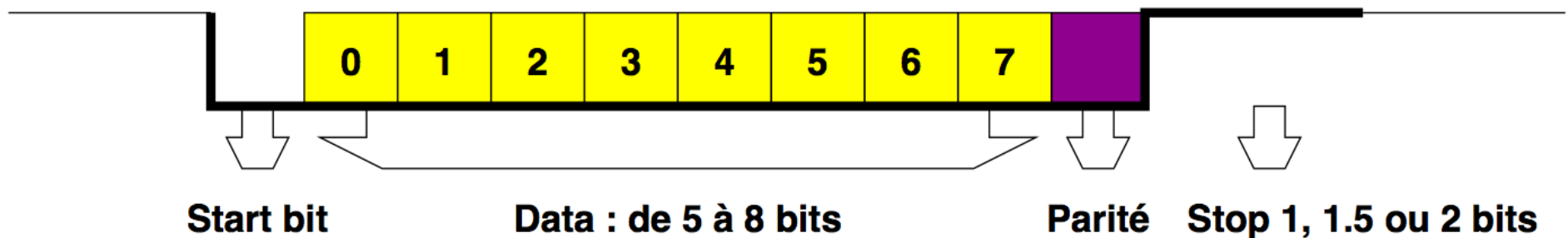
- half duplex
- horloge et data séparé (3 fils : SDA, SCL, GND),
- signal non différentiel,
- bussé

- USB

- half duplex
- horloge et data mélangé (4 fils : VBUS, D+, D-, GND),
- signal différentiel
- point-à-point.

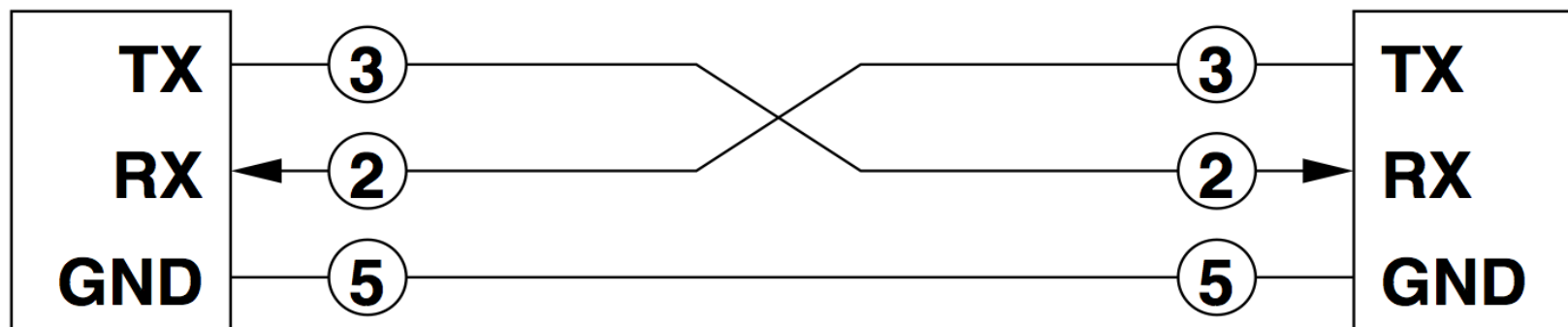
RS232 Principales caractéristiques

- Protocole des années 60.
- Protocole simple et très diffusé.
- Faible débit 10kbauds.
- Pas d'horloge: l'émetteur et le récepteur s'entendent avant.
- Protocole *handshake* optionnel : CTS, RTS, ...
- Liaison point-à-point, pas de notion d'adresse.
- Trame de données de 5 à 8 bits avec parité.



RS232 Cablage null-modem

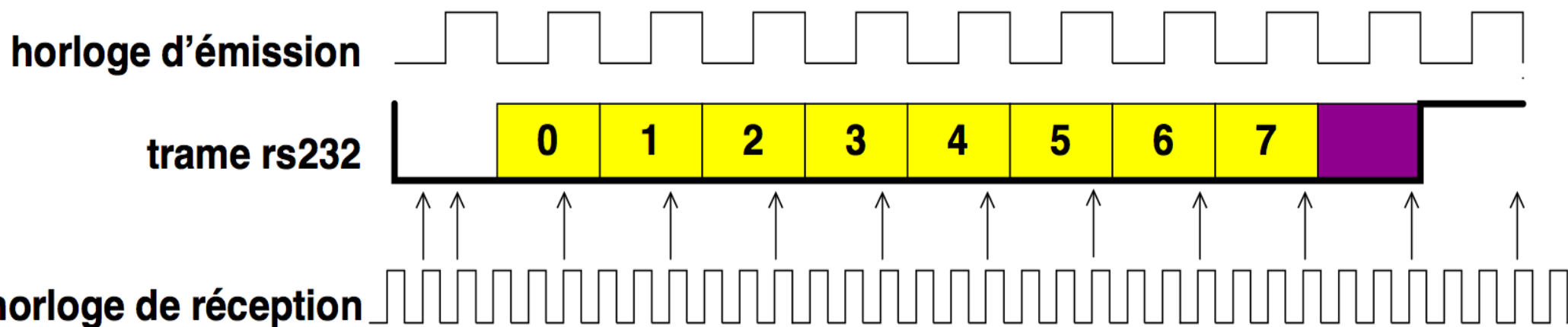
- RS232 définit deux types d'abonné : terminal et modem
- le cablage null-modem définit la communication entre 2 terminals
- au minimum 3 fils : TX, RX et GND
- on peut avoir besoin d'un convertisseur de niveaux électriques :
 - 0 logique : +8 à +12V
 - 1 logique : -8 à -12V



numéro des broches DB9

RS232 Synchro émetteur/récepteur

- L'émetteur transmet à une fréquence standardisée (1200, 2400, 4800,...)
- Le récepteur connaît cette fréquence, et sur-échantillonne pour repérer le bit start : deux échantillons à 0.
- Le récepteur a une petite marge d'erreur possible sur la fréquence.

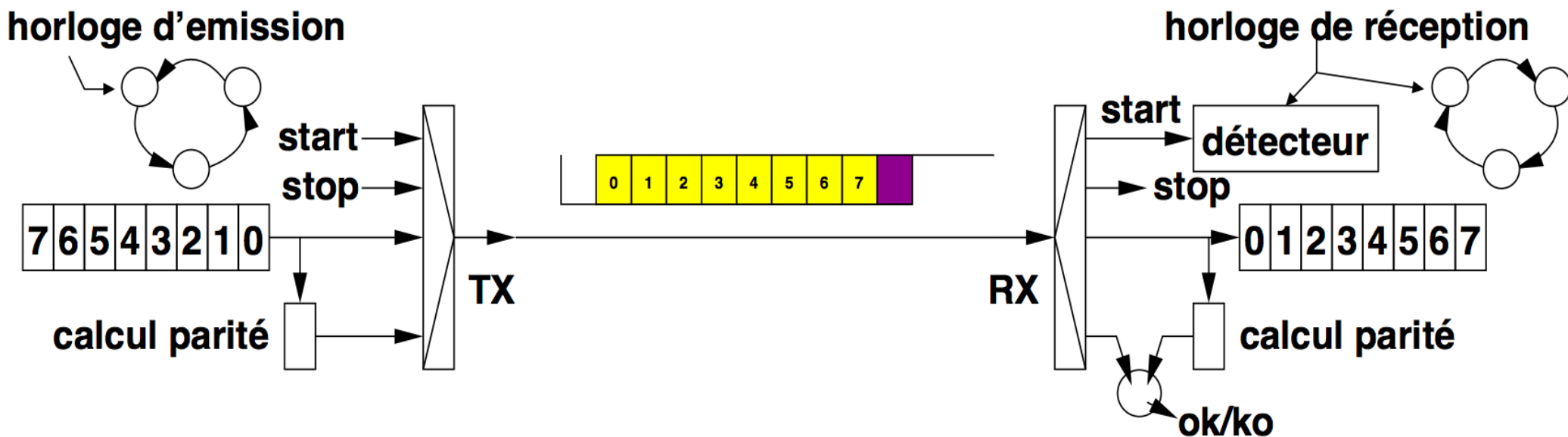


RS232 Schéma de principe

Émetteur La trame est produite par un automate qui vide un registre à décalage

Récepteur La trame est lue par un automate qui remplit un registre à décalage

Parité Un bit supplémentaire qui signe la donnée
parité paire : le nombre de bit à 1 de la donnée est rendu pair grâce au bit de parité.
parité impaire : c'est le contraire

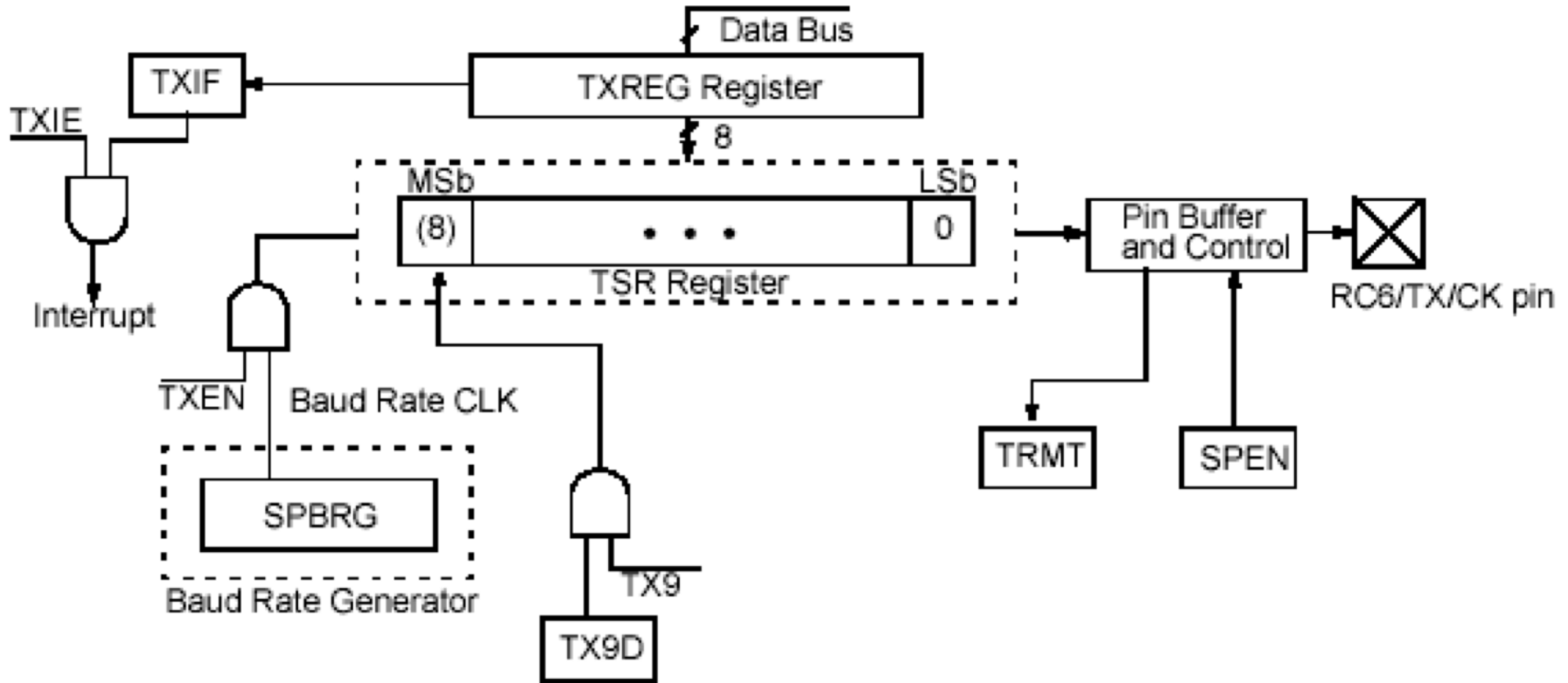


RS232 pic16f877 Comportement

- Le bit SYNC du registre TXSTAT à 0 et l'USART fonctionne dans le mode asynchrone standard, 10 (ou 11) bits sont transmis ou reçus:
 - 1 bit de START (toujours 0)
 - 8 ou 9 bits de donnée (LSB d'abord)
 - 1 bits de STOP (toujours 1)
- La transmission se fait sur la broche RC6/TX et la réception sur RC7/RX
- La configuration et le contrôle du port se fait par TXSTA et RCSTA
- La vitesse de comm est fixée par le registre SPBRG et le bit TXSTA.
BRGH
- La parité n'est pas gérée d'une façon matérielle, elle peut être gérée par soft si son utilisation est nécessaire.
- L'accès au port en lecture ou écriture se fait par les registres tampon RCREG et TXREG.
- La transmission et la réception se font par deux registres à décalage, un pour la transmission (TSR) et un pour la réception (RSR).
- L'accès au registres tampon peut se faire alors que les registres à décalage sont en train de transmettre/recevoir une donnée.
- Le drapeau RCIF est positionné quand le port a terminé une réception et
- TXIF est positionné quand le buffer de transmission TXREG est "vide".

RS232 pic16f877 Transmission

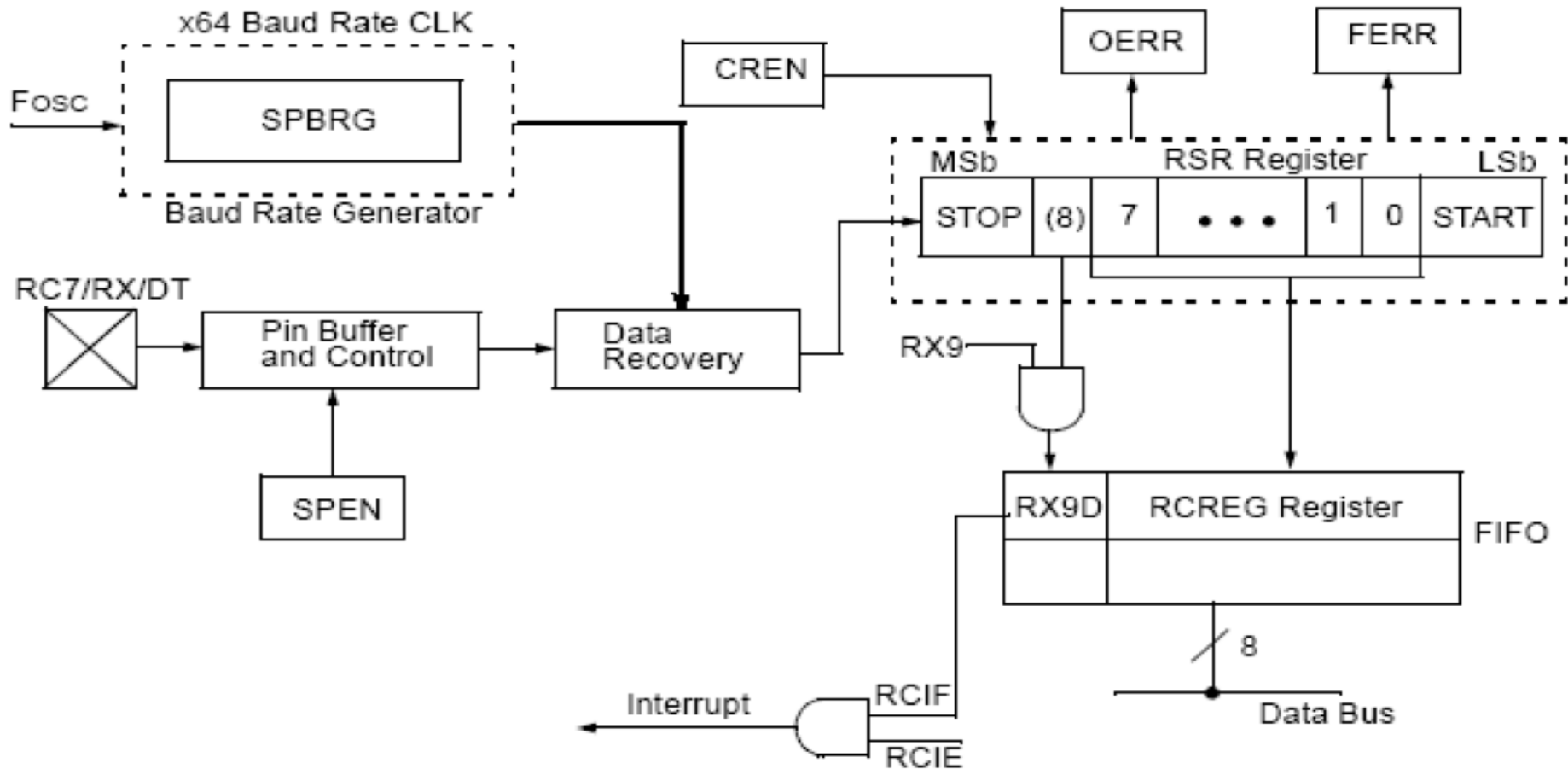
Le module USART en mode asynchrone :



Le contrôle de la transmission se fait par le registre TXSTA

CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D
------	-----	------	------	---	------	------	------

RS232 pic16f877 Réception



Le contrôle de la réception se fait par le registre RCSTA

SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D
------	-----	------	------	-------	------	------	------