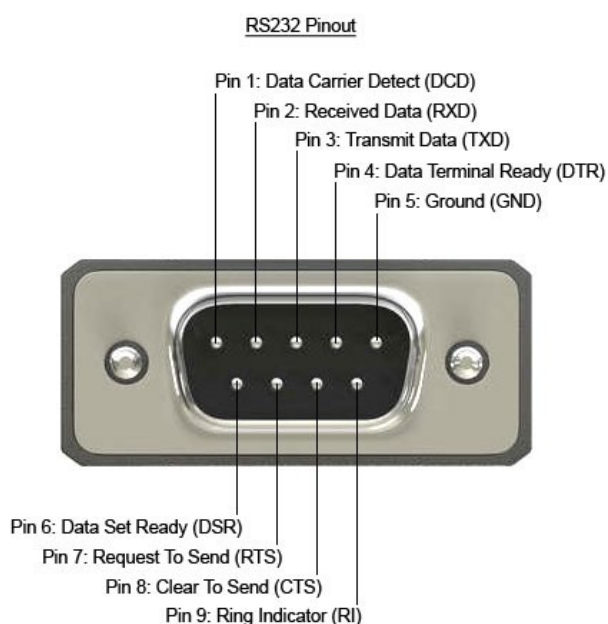


Programmation de la liaison série

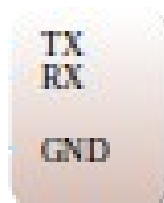
1 Mise en œuvre de la liaison série

Les ordinateurs types PC de bureau disposent généralement d'un port série. Le connecteur est un DB9 mâle comme le montre la figure ci-dessous. Un certain nombre de fils sont utilisés pour le contrôle de la communication, mais ne sont pas forcément nécessaire pour faire communiquer deux équipements.



1. Détection de porteuse (DCD);
2. Réception de donnée (RXD);
3. Transmission de donnée (TXD);
4. Terminal prêt (DTR);
5. Masse (GND);
6. Prêt à recevoir (DSR);
7. Demande d'émission (RTS);
8. Autorisation d'émettre (CTS);
9. Détection de sonnerie (RI).

1. On souhaite faire communiquer deux ordinateurs en mode full-duplex avec le minimum de fils. Proposez le câblage entre les ordinateurs, indiquez le sens de la communication et le numéro des broches utilisé



2. Comment se nomme ce type de câble ?

2 Test de la liaison série sous Linux

Pour avoir des informations sur la liaison série, il suffit de taper en ligne de commande :

```
stty -a < /dev/ttyS0
```

En vous aidant de la page de manuel de stty, donnez la signification des éléments suivants :

- speed
- parenb
- parodd
- csN
- cstopb
- crtscts
- ixon

Donnez la configuration actuelle de votre liaison série :

- Vitesse en baud :
- Nombre de bits par donnée :
- Nombre de bits de stop :
- Type de parité :
- Contrôle de flux :

Pour configurer les paramètres de la liaison série RS232, on peut utiliser également la commande stty :

```
stty -F /dev/ttyS0 vitesse "liste de drapeau"
```

Exemple :

```
stty -F /dev/ttyS0 115200 cs5 parenb parodd -ixon
```

La liaison série fonctionnera à 115200 bauds, chaque donnée sera codée sur 5 bits, avec activation de la parité, cette dernière étant impaire, il n'y aura pas de contrôle de flux xon/xoff (contrôle de flux logiciel).

Branchez un câble série entre votre PC et celui de votre voisin.

Configurer votre port série afin qu'il possède les caractéristiques suivantes :

- Vitesse en baud : 9600
- Nombre de bits par donnée : 8
- Nombre de bits de stop : 1
- Type de parité : aucune
- Contrôle de flux : aucun (ni matériel, ni logiciel)

PC1 :

```
cat /dev/ttyS0
```

PC2 :

```
écho "coucou" > /dev/ttyS0
```

A l'aide de puTTY sur un ordinateur et minicom sur l'autre, testez la communication entre deux ordinateurs.

Pour minicom, en ligne commande tapez :

```
minicom -D /dev/ttyS0
```

Configurer votre port série (via minicom) avec les mêmes caractéristiques qu'avec la ligne de commande. Configurez minicom afin d'avoir un écho local de ce que vous tapez.

Testez avec votre voisin en tapant chacun du texte dans la console de minicom. Inversez les rôles puTTY et minicom. Changez la configuration de la liaison.

3 Programmation en langage C

1. En vous inspirant de l'exemple du cours, réalisez la fonction **OuvrirPort**, elle prend en paramètre d'entrée le nom du port série sous la forme d'une chaîne de caractères et retourne un descripteur de fichier (un entier).

La structure **termios** de la librairie <sys/termios.h> permet de configurer la liaison série.

Elle est composée des champs suivants :

```
tcflag_t c_iflag;      /* modes d'entrée */
tcflag_t c_oflag;      /* modes de sortie */
tcflag_t c_cflag;      /* modes de contrôle */
tcflag_t c_lflag;      /* modes locaux */
cc_t      c_cc[NCCS];  /* caractères spéciaux */
```

Chaque champ peut prendre une ou plusieurs valeurs.

Pour assigner plusieurs valeurs à un champ, il suffit de les séparer par | (barre verticale OU binaire)

Pour proposer l'inverse d'une valeur, il faut la faire précédé du symbole ~ et faire un ET binaire.

Le code ci-dessous permet de fixer la taille des données à 7 bits, avec une vitesse de 9600 bauds et pas de parité :

```
struct termios tty ;
tty.c_cflag = B9600 | CS7
tty.c_cflag &= ~PARENB
```

2. En utilisant la librairie **termios** réalisez le code de la fonction **ConfigurerPort**, elle prend en paramètre d'entrée la vitesse, le nombre de bits de données et la parité. Les autres valeurs de la structure termios sont données dans le tableau ci-dessous

Lien vers le manuel de termios : <http://manpagesfr.free.fr/man/man3/termios.3.html>

Champ de la structure termios	Signification	Valeurs à mettre
c_iflag	<ul style="list-style-type: none">Ignorer les signaux BREAK en entrée.Ignorer les erreurs de format et de parité.	IGNBRK IGNPAR
c_oflag	<ul style="list-style-type: none">Pas de mode particulier (mettre zéro pour ce champ)	0x00
c_cflag	<ul style="list-style-type: none">Ignorer les signaux de contrôle du modem.Valider la réception.Pas de contrôle de flux rts/cts	CREAD ~CRTSCTS CLOCAL
c_lflag	<ul style="list-style-type: none">Mode non canonique.Pas d'affichage des caractères saisis.	~ICANON ~ECHO
c_cc[NCCS]	<ul style="list-style-type: none">Un caractère minimum.Lecture immédiate.	c_cc[VMIN]=1 c_cc[VTIME]=0

3. Réalisez un programme principal qui ouvre le port série ttyS0 et le configure avec les paramètres suivant 9600 bauds, 7 bits de données, parité paire et 1 bit de stop. Ce programme envoie une chaîne de caractères composée de 13 chiffres terminés par les caractères **CR**. Après avoir reçu la chaîne, le programme ferme la liaison série et se termine en ayant remis la configuration du port série avec les paramètres d'origine.

4. Réalisez la fonction **LireChaine**, elle prend en paramètre d'entrée le descripteur de fichier, le caractère de fin de chaîne attendu et en paramètre de sortie un tableau suffisamment dimensionné pouvant contenir des caractères. Son rôle est de lire caractère par caractère le port série jusqu'à l'arrivée du caractère de fin. Les caractères ainsi reçus sont placés dans le tableau pour former une chaîne de caractères.
5. Réaliser un nouveau programme principal permettant la lecture d'une chaîne de caractères terminée par retour chariot. Cette chaîne est ensuite affichée sur l'écran. Testé dans un premier temps avec votre premier programme, puis avec le lecteur de code à barres.
6. Réaliser un nouveau programme permettant de mettre en place un chat entre deux ordinateurs par la liaison série. Définir le caractère permettant de mettre fin à la session.

Lien pour un exemple de programme :

http://ftp.lip6.fr/pub/linux/french/echo-linux/html/ports-series/ports_series.html