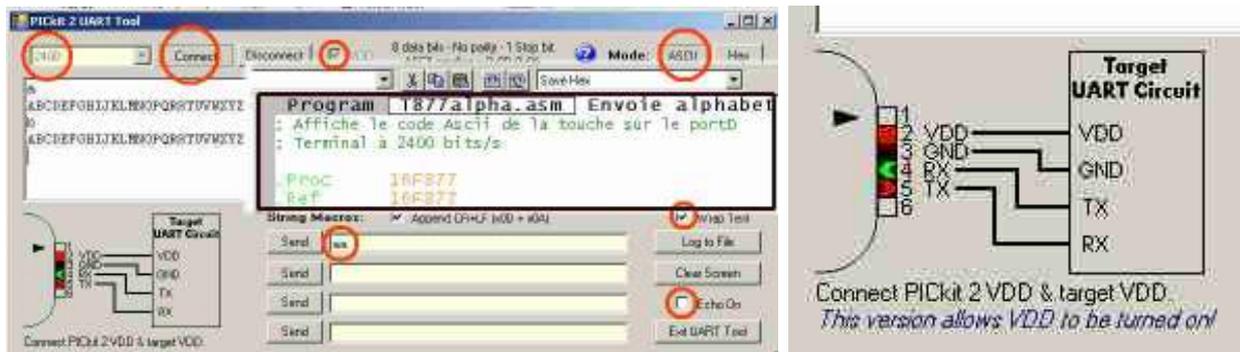


## Liaison série – UART soft et hard

Oublions le RS232, son +/-12Volts et sa prise 9 pôles qui n'existe plus sur les PC. Pour une interface simple, le Swibot se passait des circuits RS232. Le Swibot et Didelbot utilisent le DS275 [www.didel.com/doc/interf/Doc232.pdf](http://www.didel.com/doc/interf/Doc232.pdf)

Les Pics à plus de 14 pattes ont une interface série UART (Universal Asynchronous Receiver and Transmitter) dite aussi "série 5V". Le Pickit2 a un mode série et un programme d'affichage des caractères reçus intéressant pour la mise au point (voir [www.didel.com/pic/DebugSer.pdf](http://www.didel.com/pic/DebugSer.pdf) ).



Attention, les signaux Tx et Rx changent de nom. Sur le processeur maître, Tx est une sortie, Rx une entrée. Sur l'esclave aussi ! Le dessin d'un connecteur doit clairement indiquer si ce sont les signaux du processeur ou de l'interface. Pour un câble rallonge, on croise les fils, oui ou non ?

<p>La ligne est à "1" au repos. Chaque byte est précédé d'un start bit. Un silence suit le message. Oublions les options parité. La durée des bits est fixe, avec tout un choix. Par exemples 9600 bits/sec (ne parlez pas de "Baud rate" devant un spécialiste)</p>	<p>The diagram shows a signal line with a high level labeled '1' and a low level labeled '0'. A 'Start bit' is shown as a low pulse. This is followed by data bits labeled 'LSB' and 'MSB'. A 'Stop bit' is shown as a high pulse. The signal is labeled 'uart'.</p>	<p><b>UART</b></p> <p>uP board</p> <p>The diagram shows a 5-pin connector with the following labels: 'i' (top), '0' (second), 'RxD' (third), 'TxD' (fourth), '+V' (fifth), and 'Gnd' (bottom).</p>
--	--	--

Le récepteur doit naturellement travailler avec le même bit rate. Le start bit démarre un compteur qui échantillonne l'information au milieu de chaque bit reçu. Une différence de 3% sur le bit rate est acceptable.

Entre deux microcontrôleurs qui ont l'interface câblé, il suffit de préparer les bons registres et de tester des sémaphores pour savoir quand on peut placer une nouvelle information dans le registre TxReg, ou quand on a une nouvelle valeur à lire dans le registre RxReg.

Les routines d'initialisation, de lecture et d'écriture se trouvent dans la librairie X et dans tous les fichiers de test TxxAlpha.asm.

Les routines de base dans le fichier XSer.asi sont les suivantes :

IniSer PutChar GetChar PutCR PutSpace PutErr

[www.didel.com/pic/DebugSer.pdf](http://www.didel.com/pic/DebugSer.pdf) donne plus de détails

### Routines logicielles

Pour les PICs qui n'ont pas l'interface série câblés, il est facile d'écrire les routines de transfert, avec l'avantage que le câblage peut se faire sur n'importe quelle broche.

Les routines se trouvent dans les exemples [www.didel.com/pic/PicTests.pdf](http://www.didel.com/pic/PicTests.pdf) en particulier pour le 16F877 (T877Alpha.asm, T877AlphaU.asm). [www.didel.com/pic/PicT877.pdf](http://www.didel.com/pic/PicT877.pdf)