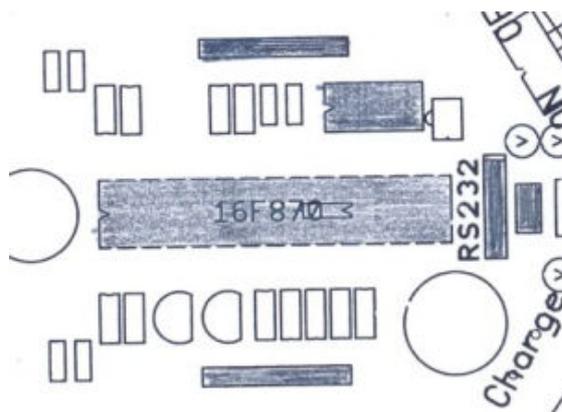
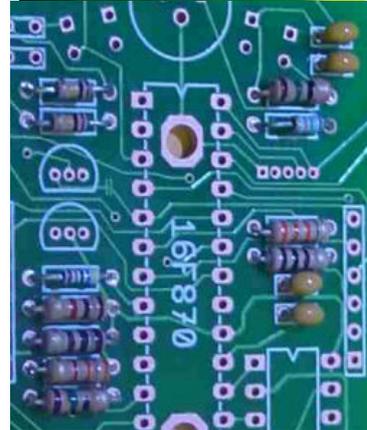
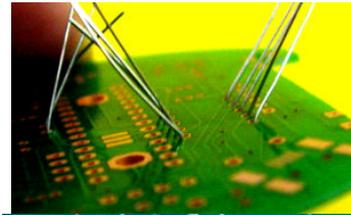
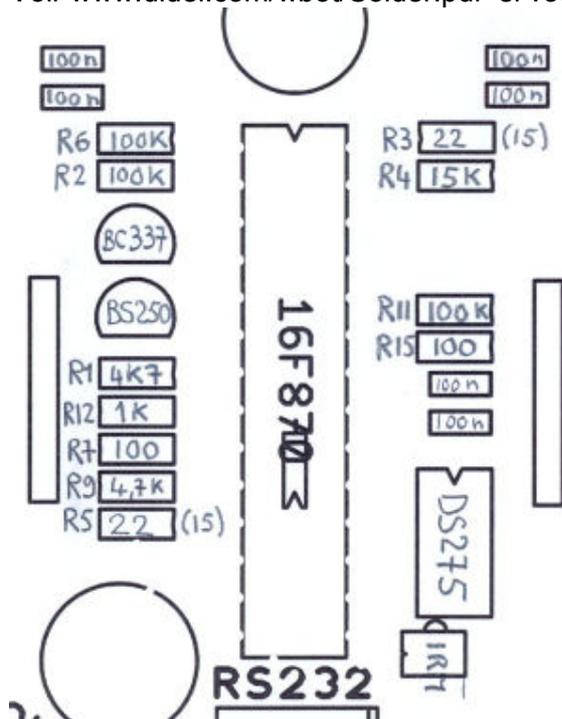




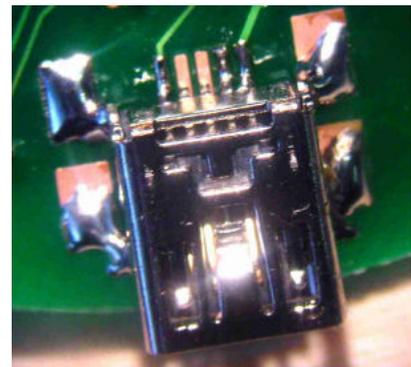
## Assemblage

Voir [www.didel.com/wbot/Solder.pdf](http://www.didel.com/wbot/Solder.pdf) si vous n'avez pas d'expérience pour souder  
Souder d'abord les résistances, condensateurs et transistors.

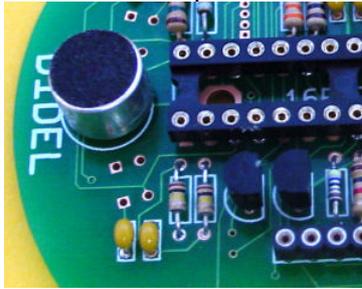


Souder les barrettes et socles. Mettre en place, retourner sur une plaquette de bois, souder une seule patte de chaque socle, vérifier que l'enfoncement est OK, repositionner si nécessaire. Souder toutes les pattes.

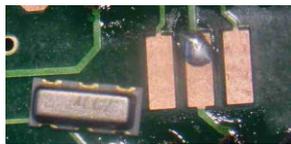
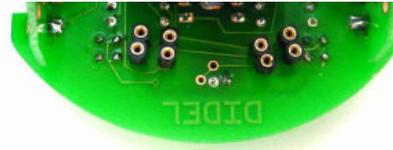
Souder le connecteur USB et l'interrupteur.  
Travail délicat, il faut une pointe fine !  
Ne pas avoir de court-circuit entre les pattes.



Souder les LEDs, le buzzer (orientation quelconque), le poussoir. Souder le microphone.



Souder dessous les connecteur pour les capteurs de distance (broche de 0.7mm). Aligner au mieux.



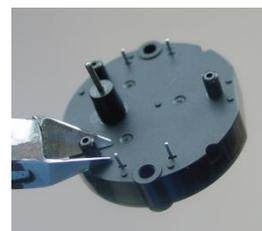
Le quartz est nécessaire seulement pour un 16F870/873 et 18F . Ne pas le mettre avec un 16F882.



Souder l'accu. Le plus est du côté de l'interrupteur. Activer l'interrupteur. Les deux leds proches doivent s'allumer (une seule si l'accu est déchargé).

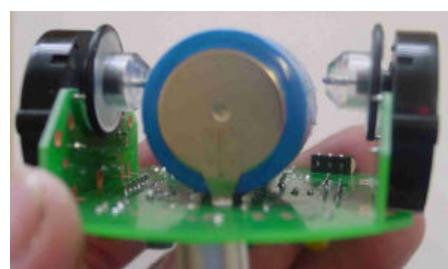


Couper à ras les picots du moteur.



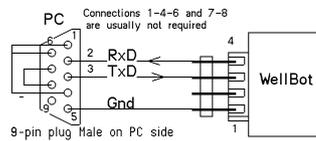
Souder les moteurs. Les PCB sont symétriques. Pour insérer les roues, mettre une cale de 0.2 à 0.4mm pour garantir la distance au circuit imprimé (papier épais, lame de rasoir). Ajouter éventuellement une goutte de colle sur l'extrémité de l'axe.

Souder les circuit moteur sur le circuit principal. Souder en un point, vérifier la perpendicularité, souder partout, sur les 2 faces.



Insérer le processeur et le DS275.

Souder le câble série.  
photo

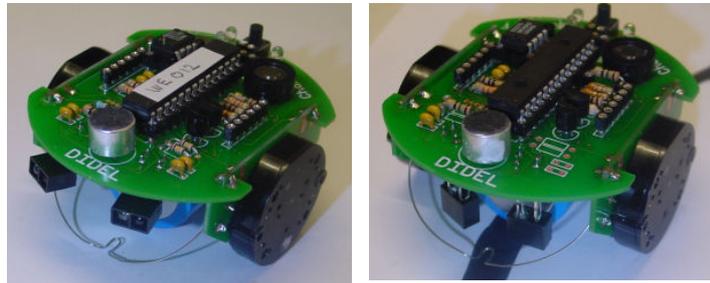


Signal	DB09 pin	Wbot pin
Gnd	5	1
RxD	3	3
TxD	2	4

### Capteur de distance

Le capteur LIT301 est une paire LED infrarouge/phototransistor. Des distances jusqu'à 40 mm peuvent être mesurées avec un éclairage normal.

Deux paires de capteurs sont livrées, droite pour le suivi de piste, coudée pour détecter les obstacles.



### Test

Le processeur est livré avec le programme WE11 (ou version ultérieure) décrit plus loin. A l'enclenchement, la LED clignote et le processeur attend pour des ordres venant par la ligne série. En pressant sur le poussoir, on lance quelques démos qu'il faut interrompre en coupant l'alimentation.

### Explication du schéma

The robot is powered by a 3.6V NiMh, 260 mAh. That capacity allows for more than 30 minutes moving and avoiding obstacles. Loading the accumulator is made with a simple resistor limiting the current and lasts 58-10 hours if the accu is completely empty.

For « measuring » the accu voltage, two LEDs are connected in serial, with a resistor in parallel with one of the LEDs. The result is that LED need a higher voltage than the other. Below 3.6V it does not blink. The robot can still work for a long time, with less power for the motors.

The distance sensor has an emitting part, with the two IR LEDs in serie and controlled by a transistor, so the power consumption is minimum. The loudspeaker is in parallel, it was necessary to share.

The push-button is also shared with the LED under control of the processor. When depressed, the LED is on. Height outputs are used to control the stepping motor. See [www.didel.com/bot/Step.pdf](http://www.didel.com/bot/Step.pdf) (in french)

The electret microphone uses a simple schematic. The transistor discharge the capacitor when there is a strong noise (hand clac) and the zero-level pulse is detected by the processor, if it checks that input.

Finally, the RS232 interface just uses a DS275 circuit that do all the job.

The processor on the socket, a 16F870 has only half of it's memory filled. It can be replaces by a 16F871, 16F873, 16F737, 16F84x or 18F2220. The 18F is recommended if the application in completely rewritten in C.

### Ports assignment

PortA	PortB	PortC
RA0 sensor i/o	RB0 motor out	RC0 LED/sw
RA1 sensor i/o	RB1 motor out	RC1 HP
RA2 bico data to bico	RB2 motor out	RC2 IR
RA3 in micro	RB3 motor out	RC3 ext I2C ck
RA4 bico data from bico	RB4 motor out	RC4 ext I2C data
RA5 bico Ck	RB5 motor out	RC5 ext
	RB6 motor out	RC6 Tx
	RB7 motor out	RC7 Rx

### Reprogramming the WellBot870

Pic microcontrollers can be programmed with many programmers. We use the PicKit2

and provide an adapter toward the 5-pin 1.27mm pitch connector on the WellBot. The HEX files the PicKit2 can program may come from different sources :

- SmileNG editor and assembler. The source code of the Wellbot programs, using CALM notations, are available.
- Microchip assembler and C compiler
- Mikro C compiler
- PicBasicPro compiler



## Options

[www.didel.com/wbot/WeBico.pdf](http://www.didel.com/wbot/WeBico.pdf)

telecommande

## Logiciels Wellbot

Le premier but du WellBot est d'apprendre à programmer et différentes options sont possibles.

Le programme We11 permet quelques démonstrations et la télécommande avec la manette Emir/Bimotel. L'interface RS232 (et un adaptateur USB si nécessaire) dialogue avec les ressources via un programme terminal, écrire des programme en VisualBasic, etc ou utiliser CeeBot.

Une série de programmes permet de comprendre comment les fonctions du WellBot sont programmées en assembleur, et comment on peut commander l'affichage couleur.

La documentation logicielle se trouve sous WbSoft.pdf et WbSoft.html pour les liens aux fichiers source.