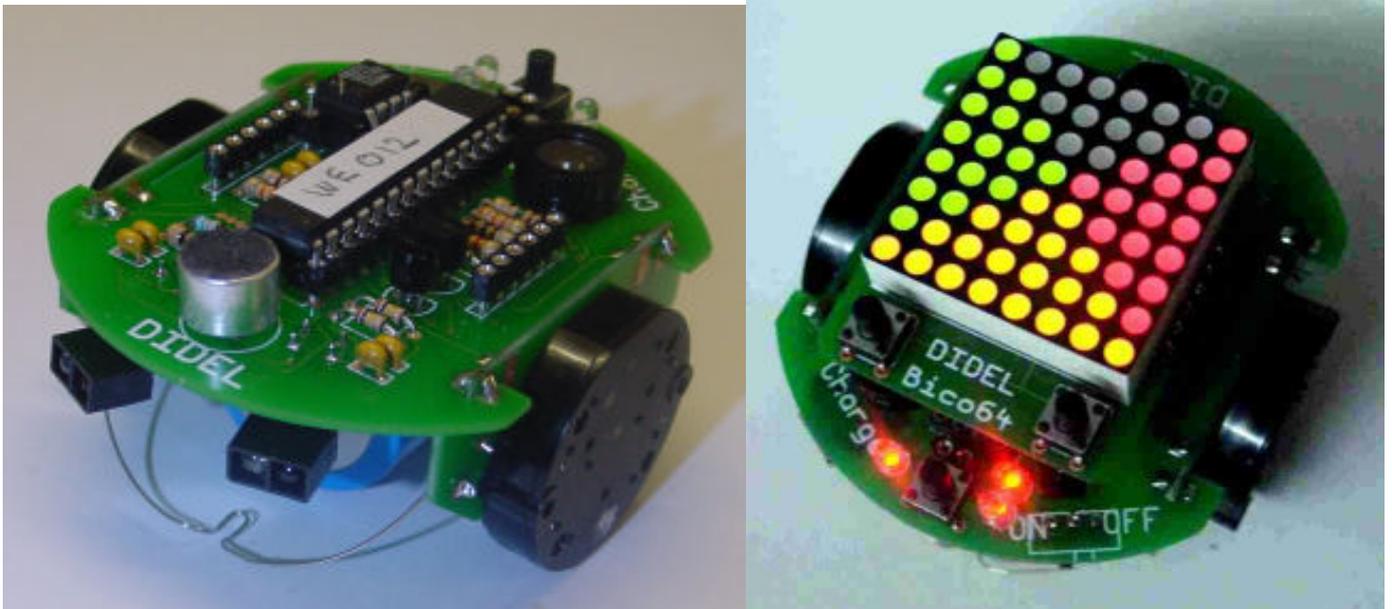
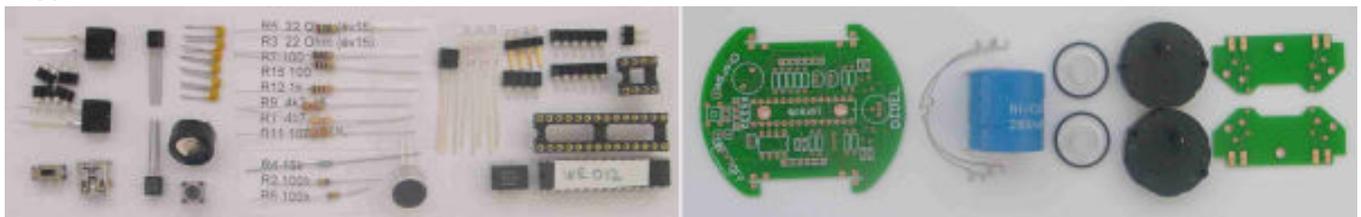


Wellbot-Pic avec 16F882



Le Wellbot est très intéressant pour son prix et sa fonctionnalité. Ses moteurs pas-à-pas permettent des déplacements précis. Deux capteurs de distance infrarouges permettent soit d'éviter les obstacles, soit de suivre une ligne. Il est complété d'un affichage 3 couleurs qui motive à écrire des programmes spectaculaires.

Le kit est facile à souder et contient 3 circuits imprimés, 2 moteurs et roues, 1 batterie NiCd de 3.6V 280 mAh, tous les composants électroniques et un processeur Pic 16F882 qui a remplacé le 16F870 initial.



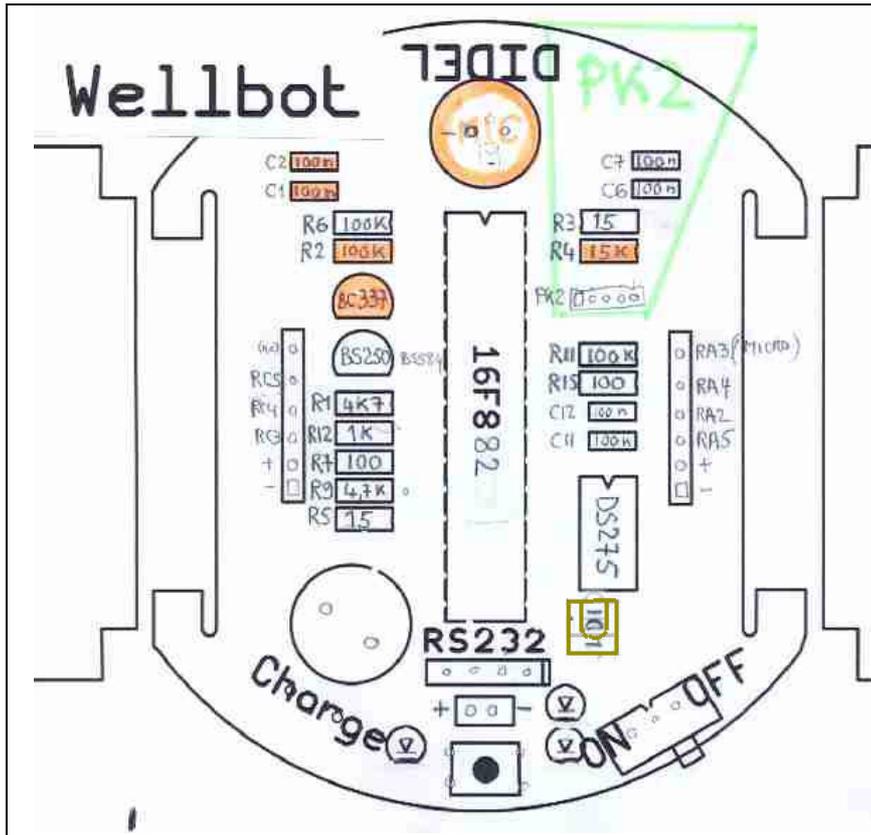
Liste des composants

3 x PCB (circuits imprimés)	2 x 15 Ohm brun-vert-noir	1 x on/off switch	2 x VID29 motors
3 x Leds	2 x 100 Ohm brun-noir-brun	1 x poussoir	1 x NiCd 280 mAh
1 x socket DIP8	1 x 1k brun-noir-rouge	1 x Electret micro	1 x USB conn
1 x socket DIP28	2 x 4k7 jaune-violet-rouge	1 x IR receiver	2 x roues
2 x strips 6c female 0.5	1 x 15k brun-vert-orange	1 x Pic 16F882	2 x o-rings
1 x strip 4c female 0.7	3 x 100k brun-noir-jaune	1 x DS275	2 x springs
4 x strips 2c female 0.5	6 x 100nF	4 x LIT301	3x 2m fil souple
1 x strip 4c mâle 0.7	1 transistor BS250 ou ZUP21	1 x buzzer	1 conn. RS232
	1 transistor BC337 ou 3904		1 kit PK2

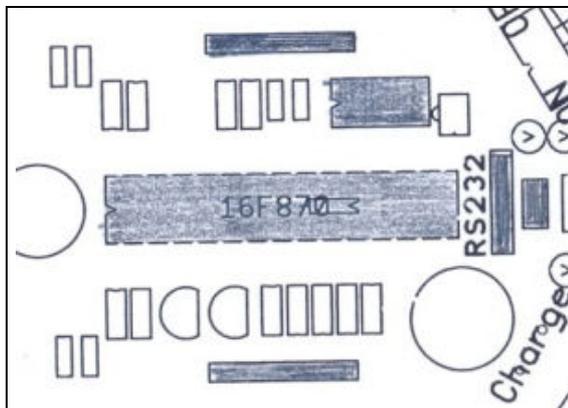
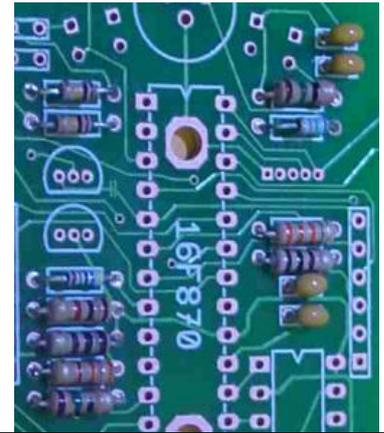
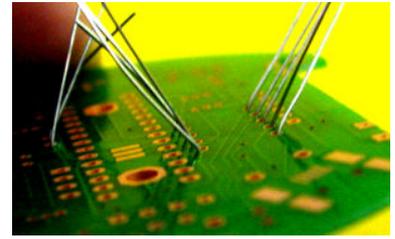
Soudure

Voir www.didel.com/wbot/Solder.pdf si vous n'avez pas d'expérience pour souder

Montage

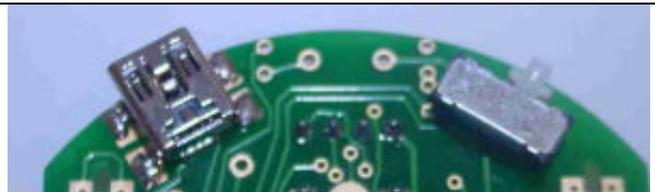


Souder d'abord les résistances, condensateurs et transistors.

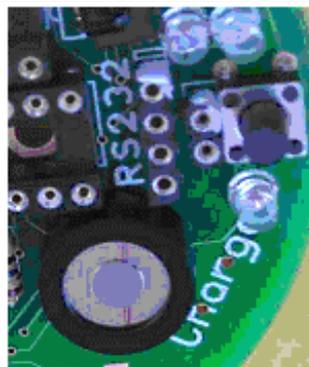
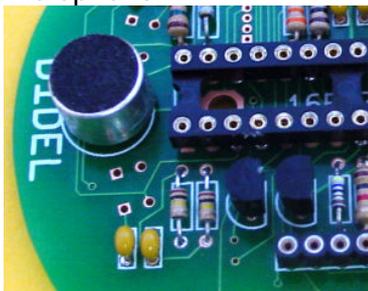


Souder les barrettes et socles.
Mettre en place, retourner sur une plaquette de bois, souder une seule patte de chaque socle, vérifier que l'enfoncement est OK, repositionner si nécessaire.
Souder toutes les pattes.

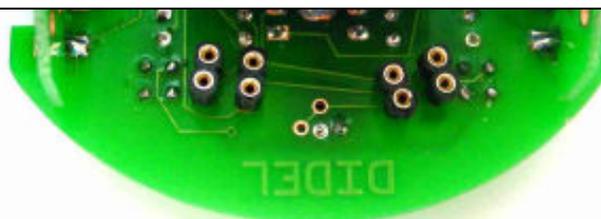
Souder l'interrupteur.
Le connecteur USB nécessite un bon fer et est déjà soudé.



Souder les LEDs (patte courte vers l'extérieur),
le buzzer (orientation quelconque), le poussoir.
Souder le microphone.

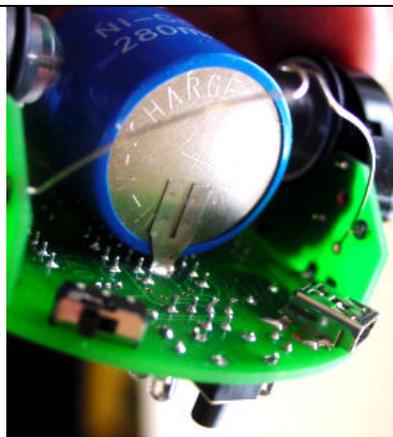


Souder dessous les connecteurs pour les
capteurs de distance (broches de 0.7mm).
Aligner au mieux.

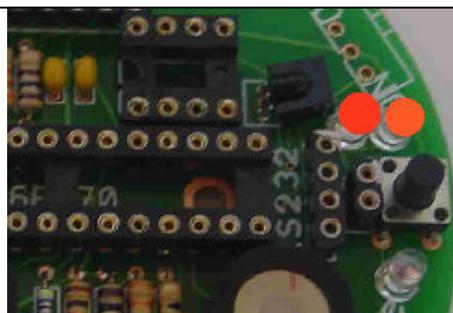


Souder l'accu.

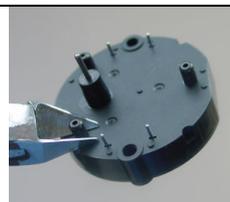
Le plus est du côté de l'interrupteur.



Activer l'interrupteur. Les deux leds proches
doivent s'allumer (une seule si l'accu est
déchargé).



Couper à ras les 3 picots plastique du moteur.



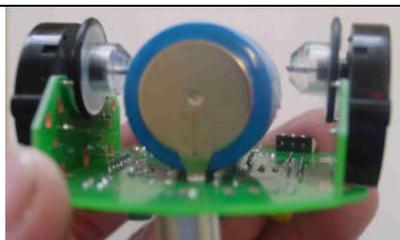
Souder les moteurs. Les PCB sont symétriques.
Pour insérer les roues, mettre une cale de 0.2 à
0.4mm pour garantir la distance au circuit
imprimé (papier épais, lame de rasoir).
Ajouter éventuellement une goutte de colle sur
l'extrémité de l'axe.



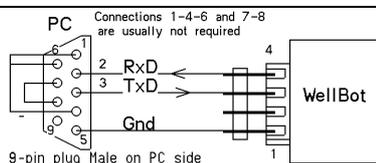
Souder les circuits moteurs sur le circuit principal.
 Souder en un point, vérifier la perpendicularité, souder partout, sur les 2 faces.

Souder les deux ressorts d'équilibrage.

Insérer le processeur et le DS275.



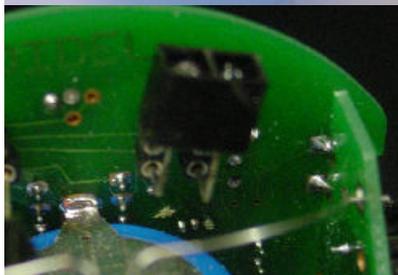
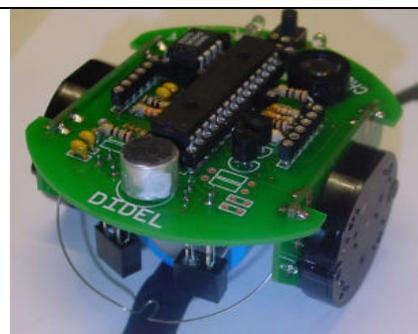
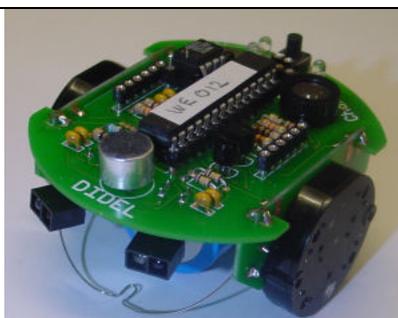
Souder le câble série.



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. La largeur d'une piste à suivre est de 15 à 18mm.



Test

Le processeur est livré avec le programme WE11 (ou version ultérieure).

A l'enclenchement, la LED clignote et le processeur attend des ordres venant par la ligne série. En pressant sur le poussoir, on lance quelques démos qu'il faut interrompre en coupant l'alimentation.

Schéma

Le schéma est donné à la page suivante.

Le robot est alimenté par un accu NiMH 3.6V, 260 mAh., permettant un fonctionnement d'une demi-heure. La recharge de l'accu se fait simplement au moyen d'une résistance limitant le courant et dure 8-10 heures si l'accu est complètement déchargé.

Pour mesurer la charge de l'accu, on utilise deux LEDs connectées en série, avec une résistance en parallèle avec l'une des LEDs. Le résultat est que l'une des LEDs nécessite davantage de tension que l'autre. En dessous de 3.6V elle ne clignote pas. Le Wellbot peut encore fonctionner longtemps, mais avec moins de puissance sur les moteurs.

La partie émettrice du capteur de distance est formée de deux LEDs IR en série contrôlées par un transistor, afin que la consommation soit minimale.

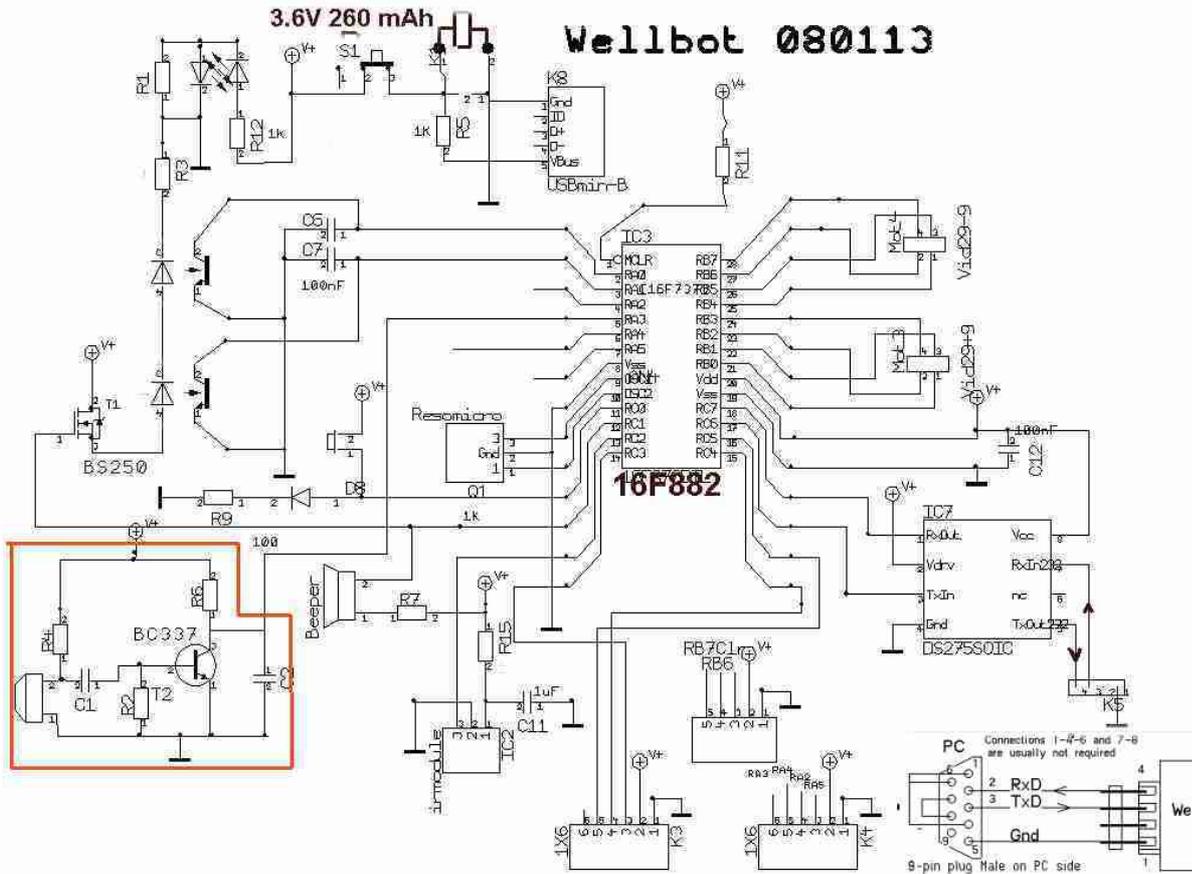
Le bouton-poussoir est aussi partagé avec les LEDs, sous contrôle du processeur. Quand le poussoir est pressé, la LED est ON.

Les lignes de sortie de poids forts sont utilisées pour commander les moteurs pas à pas. Voir

www.didel.com/bot/Step.pdf

Cadré en orange dans le schéma, le détecteur de sons : en claquant des mains, une impulsion arrive sur le processeur, qui doit l'intercepter.

Le processeur 16F882 est placé sur un socle et sa mémoire est remplie à moitié seulement. Il peut être remplacé par un des contrôleurs suivants : 16F870, 16F871, 16F873 (avec un quartz) 16F737, 16F84x ou 18F2220. Le 18F est recommandé si l'application est écrite en C.



Assignment des Ports

PortA	PortB	PortC	Signal serie	DB09 pin	Wbot pin
RA0 sensor i/o	RB0 motor out	RC0 LED/sw	<u>Gnd</u>	<u>5</u>	<u>1</u>
RA1 sensor i/o	RB1 motor out	RC1 HP	<u>RxD</u>	<u>3</u>	<u>3</u>
RA2 bico data to bico	RB2 motor out	RC2 IR	<u>TxD</u>	<u>2</u>	<u>4</u>
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			

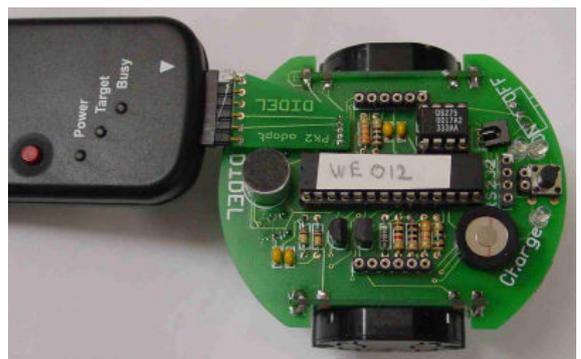
Reprogrammer le WellBot882

Pour reprogrammer le WellBot, nous utilisons le Pickit2 et un adaptateur PK2 vers le connecteur 5-pin 1.27mm du WellBot.

www.didel.com/wellbot/Pickit2.pdf

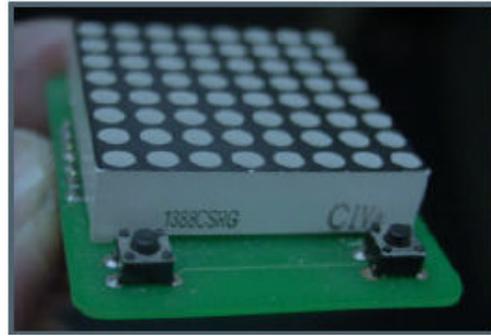
Le Pickit2 peut programmer des fichiers HEX provenant de diverses sources:

- De l'éditeur-assembleur SmileNG. Tous les sources des programmes du Wellbot écrits avec les notations CALM sont disponibles.
- Du compilateur C et de l'assembleur Microchip
- Du compilateur Mikro C
- Du compilateur PicBasicPro.



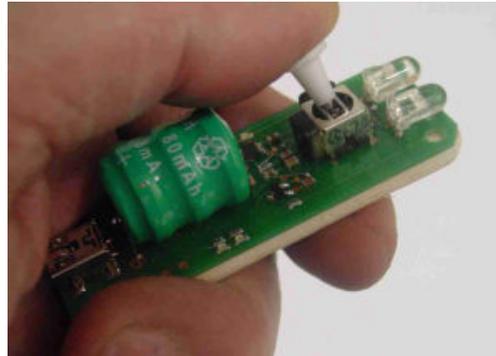
Affichage BICO64

Le réseau de 64 LEDs est décrit ici www.didel.com/wellbot/WeBico.pdf
Le logiciel Wellbot a les primitives pour communiquer en série avec l'affichage.



Télécommande

Le Wellbot dispose d'un module IR intégré. Les utilisateurs peuvent programmer leurs propres routines afin d'être compatibles avec n'importe quel émetteur IR. Didel supporte la télécommande Emir2 du robot Bimo avec contrôle proportionnel.



Logiciels Wellbot

Le Wellbot est livré avec un programme de démonstration qui illustre quelques comportements choisis par un certain nombre d'actions sur le poussoir.

Le Wellbot est fait pour être programmé par l'utilisateur. Il faut pour cela un Pickit2. Voir www.didel.com/wellbot/Pickit2.pdf.

Les programmes sont écrits en CALM sur l'environnement gratuit SmileNG.

Une documentation didactique abondante existe pour apprendre à programmer les PICs en CALM, voir <http://www.didel.com/pic/Programmer.pdf>

La documentation logicielle pour le Wellbot suppose des connaissances de base. Des programmes de test et de démo sont proposés, pour mettre en œuvre les différentes fonctionnalités et aider à les comprendre.

Le programme We11 déjà chargé dans le processeur permet quelques démonstrations et l'utilisation de la télécommande Emir2/Bimotel pour contrôler les déplacements. L'interface RS232 (et un adaptateur USB si nécessaire) permet de dialoguer avec les ressources via un programme terminal, écrire des programmes 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 est livrée sur CD avec le Wellbot. La version la plus récente se trouve sous www.didel.com/wellbot/WellbotSoft.pdf et www.didel.com/wellbot/WellbotSoft.zip pour les fichiers sources.

A noter encore que la programmation peut se faire avec d'autres environnements. Pour programmer en C, le 16F882 et le compilateur de Microchip est adéquat. Un 18F2450 semble préférable pour du C.