

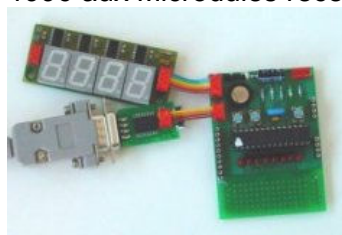
L'inventaire des outils que nous connaissons pour apprendre à programmer se trouve sous [www.bricobot.ch/programmer/](http://www.bricobot.ch/programmer/) (fichier implicite index.htm)

Si vous continuez la lecture, c'est que la programmation en assembleur vous intéresse. En assembleur CALM pour Microchip 10F 12F 16F pour être plus précis. Et avec l'éditeur-assembleur SmileNG et la documentation très complète résultant de plusieurs tentatives pour expliquer l'assembleur et ses applications à des étudiants EPFL d'abord, puis à travers internet.

Vous aimeriez bien maîtriser les PICs de Microchip. Nous avons fait beaucoup de documentation sur les PICs, mais c'est en CALM et ce n'est pas un produit "commercial". La doc de Microchip est en anglais et cela vous embête aussi. Et il y a encore tous ces gens qui ne pensent qu'à des programmes complexes et qui font croire que l'assembleur est difficile et limité. Que pensez-vous du motard qui critique le VTT ? Ou du VTTiste qui roule sur une grande route ? Assembleur et C ont chacun leurs processeurs, leurs domaines d'application et leurs spécialistes !

C'est extraordinaire ce que l'on peut faire avec ces petits processeurs de 6, 8, 14, 18, 20, 28 ou 40 pattes, en boîtier DIL, SO, SSOP.

Didel a fait des dizaines d'applications avec ces processeurs et a développé des environnements de développement adaptés, depuis la simple carte 16F84 « PicGénial » de 1996 aux Microduals récents.



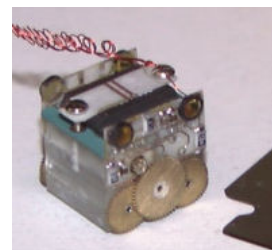
PicGénial de 1996



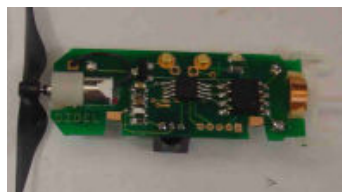
Commande des signaux Z-H-V d'un ancien moniteur video



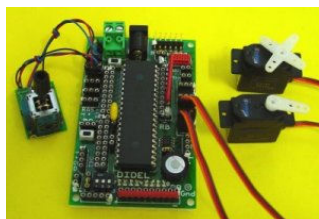
Les cartes Ir3 et UB4 décodent les commandes pour PicooZ



Robot 1cm3. Moteurs smooovy 3mm commandés par un F84



Minicarte pour Airboat



Carte Dev877



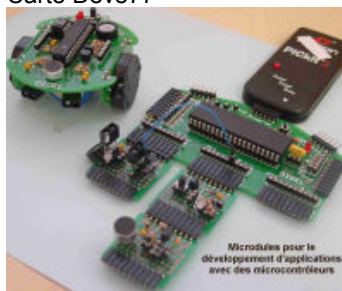
Le fameux Bimo



La souris Rhodo



Le robot DidelBot dessine



Mise au point du Wellbot avec les Microduals



Le Wellbot et son affichage



Le bâton magique

Mais revenons à ce qui peut vous intéresser et pour quoi nous avons de bonnes solutions : **Apprendre l'assembleur** avec le minimum d'effort pour mémoriser les instructions et savoir exactement ce qu'elles font

→ Calm a été développé à l'EPFL dans ce but

**Ecrire les programmes** avec un éditeur-assembleur facile à utiliser, insérant les erreurs dans le source, créant des listages compacts pour l'imprimante, avec la structure mise en évidence

→ SmileNG développé par S.Gerlach remplit complètement cet objectif

**Programmer tous les PICs** de Microchip

→ Le PicKit2 est vraiment bien, et il a des fonctions supplémentaires très intéressantes. Didel a l'expérience (et les accessoires) pour programmer "in situ" les processeurs les plus petits.

Si vous êtes très jeune ou intéressé par programmer des séquences avec nos macros en français, vous avez le choix entre

programmer les moteurs et Leds du Bimo

[www.bricobot.ch/docs/Abimo.pdf](http://www.bricobot.ch/docs/Abimo.pdf)

programmer un cœur avec 16 Leds de couleur

[www.bricobot.ch/docs/Coeur4Prog.pdf](http://www.bricobot.ch/docs/Coeur4Prog.pdf)

programmer des sculptures lumineuses autour des circuits Deneb ou Orion

[www.bricobot.ch/docs/Deneb.pdf](http://www.bricobot.ch/docs/Deneb.pdf) [www.bricobot.ch/docs/Orion.pdf](http://www.bricobot.ch/docs/Orion.pdf)

Si vous ne savez pas si la programmation en assembleur vous intéresse vraiment, jouez avec le simulateur de Dauphin <http://www.epsitec.ch/dauphin/>

Si vous voulez aller plus loin avec le Dauphin, étudiez ses instructions et les exemples de programme plus complexes dans

<http://www.epsitec.ch/downloads/documentation/dauphin/dauphin2.pdf>

Si vous voulez pouvoir faire votre propre système avec un microcontrôleur et activer des moteurs, des LEDs, un haut-parleur, etc, alors vous avez encore plusieurs choix :

Vous préférez apprendre en voyant des programmes simples qui expliquent petit à petit ce que peut faire un microcontrôleur, sans se perdre dans toutes ses possibilités, mais avec des liens si cela vous intéresse de creuser.

[www.didel.com/pic/Prog877.pdf](http://www.didel.com/pic/Prog877.pdf) Apprendre à programmer les PICs de Microchip avec l'environnement efficace du Microdude 16F877 et accessoires.

Utilise l'ensemble [www.didel.com/08micro/M2840Eval.pdf](http://www.didel.com/08micro/M2840Eval.pdf)

*Le cours assembleur PIC organisé à l'EPFL dès le 8 septembre 2009*

*utilise la première version de ce kit. Voir [www.didel.com/pic/CoursPic2009.pdf](http://www.didel.com/pic/CoursPic2009.pdf)*

Une approche moins complète a été écrite en 2000 pour le 16F84 et le 16F877

[www.didel.com/picg/picg84/](http://www.didel.com/picg/picg84/) [www.didel.com/picg/picg87x/](http://www.didel.com/picg/picg87x/)

Vous préférez que l'on explique l'architecture, les principes, les instructions, avant de voir les exemples

[www.didel.com/pic/Architecture.pdf](http://www.didel.com/pic/Architecture.pdf) (non terminé)

Vous connaissez d'autres assembleurs, et un document en anglais convient.

[www.didel.com/picg/doc/PicSoft.pdf](http://www.didel.com/picg/doc/PicSoft.pdf)

Sur internet, on trouve plusieurs cours plus ou moins développés qui utilisent les outils et notations proposées par Microchip. Citons en français :

Bigonof <http://www.abcelectronique.com/bigonoff/>

Oumnad <http://z.oumnad.123.fr/index.php>

et les livres de Claude Tavernier, paru chez Dunod <http://www.tavernier-c.com/>

Bonne chance pour trouver l'outil qui vous convient le mieux. Si nous ne le connaissons pas, dites lequel et nous compléterons notre liste. Merci.

jdn 090823/091207