

[English version](#) here

## Modules flexibles pour le développement d'applications avec les microcontrôleurs PIC et AVR

Pour une doc complète, voir [www.didel.com/08micro/Microdules.pdf](http://www.didel.com/08micro/Microdules.pdf)

Liste de prix : <http://www.didel.com/08micro/Micro.xls>

Ensemble d'évaluation 16F628 offre 100 CHF : <http://www.didel.com/08micro/M18Eval.xls>

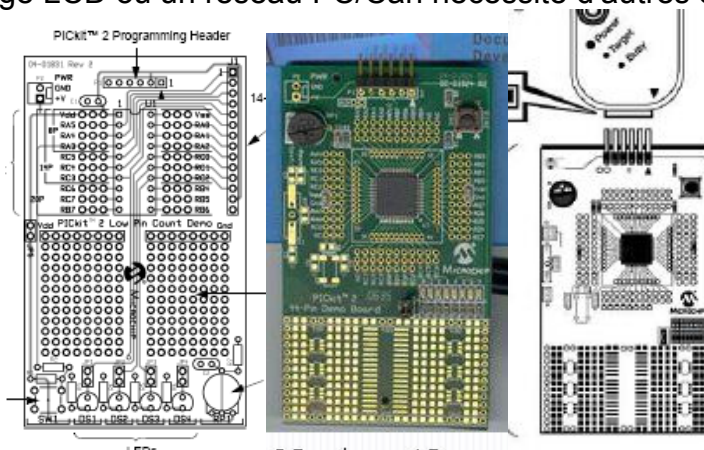
Nouveau – Mars 2010 –

[www.didel.com/pic/Deverminage.pdf](http://www.didel.com/pic/Deverminage.pdf)

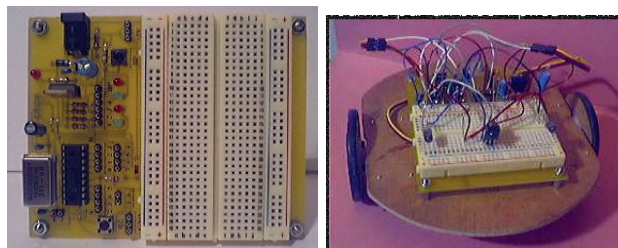
[www.didel.com/pic/DebugSer.pdf](http://www.didel.com/pic/DebugSer.pdf)

Quelles sont les solutions pour développer une application avec un microcontrôleur et des interfaces variées ? Seules les applications qui ont des contraintes de temps réel nous intéressent. Jouer avec un affichage LCD ou un réseau I<sup>2</sup>C/Can nécessite d'autres outils.

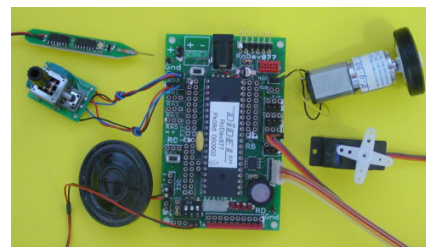
Microchip et d'autres proposent des cartes pour se familiariser avec la programmation. Elles comportent une zone universelle qui convient pour câbler les circuits annexes. Mais alors, la carte n'est plus utilisable pour le projet suivant.



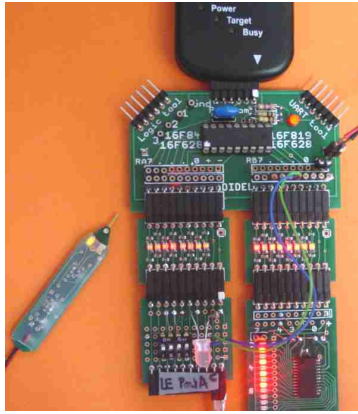
Une carte d'expérimentation a l'avantage d'accepter une grande variété de composants, mais la fiabilité est mauvaise et le montage doit être refait si on veut redévelopper du soft ou faire des modifications quelques mois plus tard.



La carte **Dev877** de Didel est optimisée pour un schéma avec des pot, moteurs, servos. Elle a des connecteurs pour câbler ce que l'on veut, mais elle est trop compacte pour expérimenter.



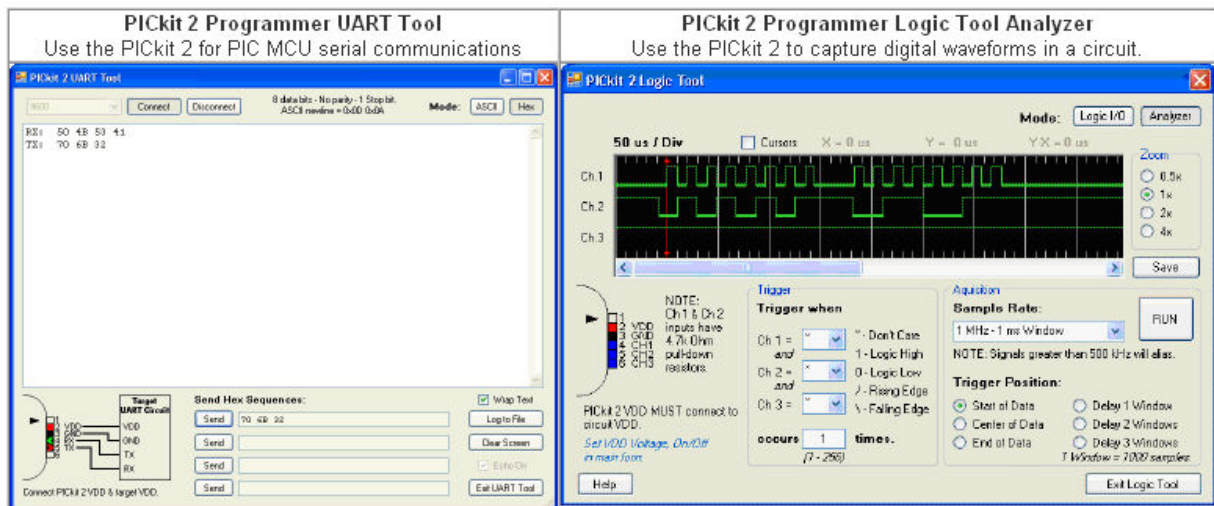
Les **Microdules** apportent une solution qui n'est pas centrée sur un groupe d'application. Ils visent le développement d'applications pour des processeur de 6 à 40 pattes, avec des contraintes temps réel et pour lesquelles on a un souci d'optimisation. L'assembleur CALM et l'éditeur-assembleur SmileNG, complété par le programmeur Picki2 forment un ensemble efficace et facile à maîtriser.



Les **Microdules** sont un ensemble de cartes pour faciliter le développement d'applications avec des microcontrôleurs de toutes tailles. Un choix de cartes compatibles avec les processeurs les plus courants est complété par des modules 8 bits se connectant sur les ports et adaptés pour mettre au point une application donnée. Quand le schéma et le logiciel ont été testés, le circuit imprimé définitif (ou Veroboard compact) est réalisé et connecté à l'application.

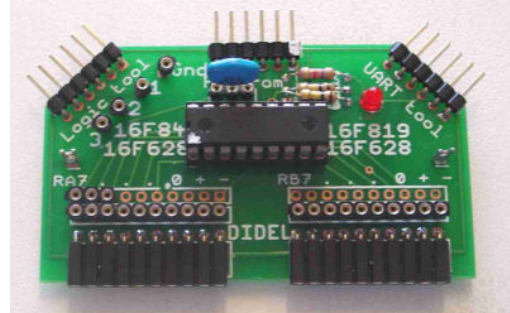
Les cartes processeurs n'ont que des connecteurs et sont prévues pour être connectées à un programmeur et à un ensemble de petites cartes interfaces, câblées en fonction de l'application.

Les cartes compatibles avec les processeurs de Microchip sont optimisées pour le programmeur PicKit2 et ses deux fonctionnalités complémentaires, uart et analyseur logique. On peut déplacer le PicKit2 ou en avoir plusieurs.



## Exemples

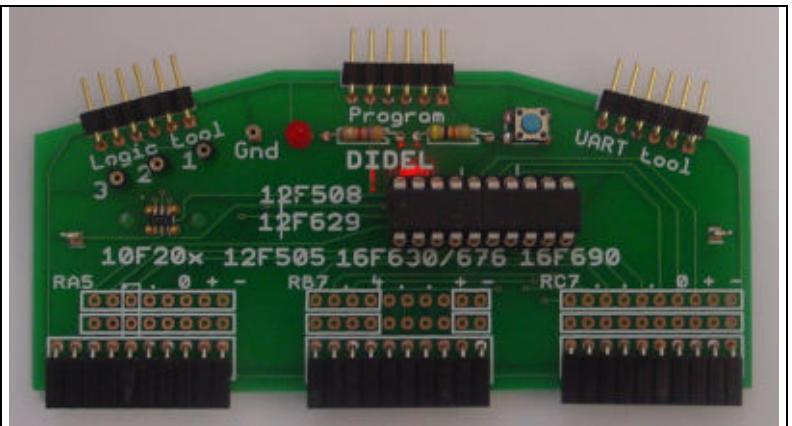
Carte **M18** pour  
F84/88/626/627/628/819 (Ports A  
et B).



Kit complet (10 PCB) avec un 16F628 seulement 100 CHF :  
<http://www.didel.com/08micro/M18Eval.xls>

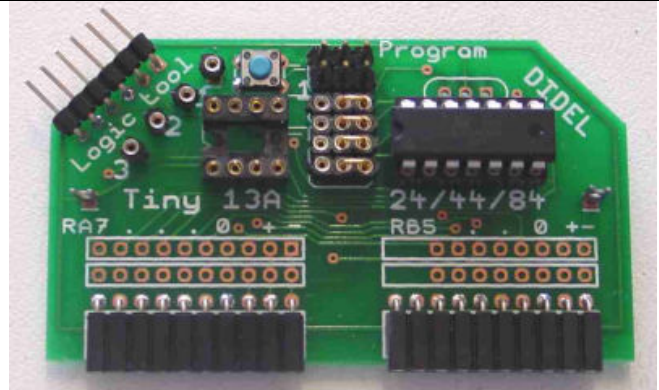
Carte **M20** pour 10F20x (portB 4 bits),  
 12F508, 12F505, 12F629/675/etc  
 (PortA 6 bits),  
 16F630/676/etc  
 (Ports A et C 6 bits),  
 16F690/etc  
 (Ports A et B et C 5-8 bits).

La carte **M2840** pour F87x 88x 18F

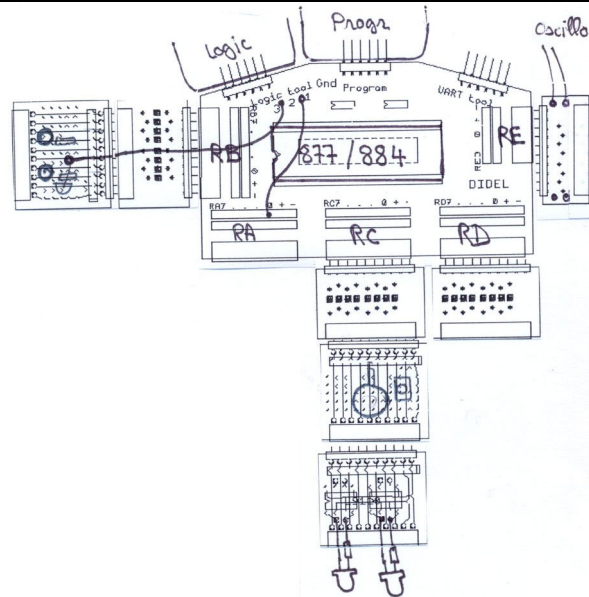


Carte **A814** pour Tiny13A, Tiny24,  
 44, 84

Carte **A2840** pour la famille AtMega



Mise au point avec un 16F877  
 d'une application qui sera réalisée  
 avec un 16F630 SSOP sur un PCB  
 de quelques cm<sup>2</sup>.  
 Voir  
[www.didel.com/08micro/WbTest.pdf](http://www.didel.com/08micro/WbTest.pdf)



**Nouveau**

Pour  
 dépanner  
 des  
 programmes  
 temps réel

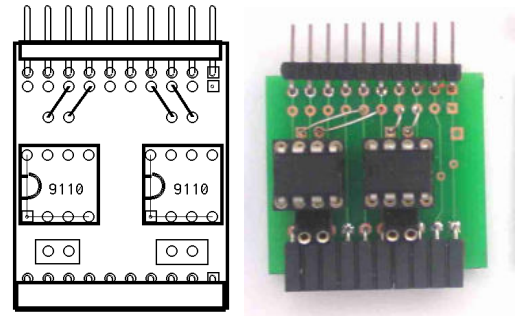


16F946 avec 7 ports.  
 1 ou 2 pour  
 l'application. 5 pour  
 l'observer.

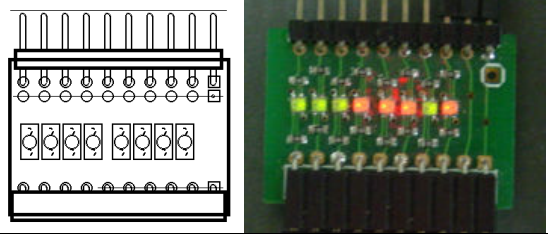
Ce n'est pas un outil  
 de développement  
 pour le 16F946, mais  
 pour un 10F202,  
 16F630, etc.

## Exemple d'application

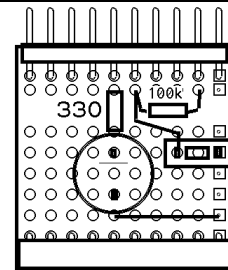
Pour tester le Bimo on doit câbler  
RC0 RC1 moteur gauche  
RC4 RC5 moteur droit  
Le module L9110 est câblé pour que les entrées soient sur les bons bits. Les moteurs ou des leds bicolors se branchent sur les connecteurs de sortie



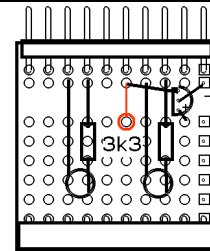
Un affichage peut toujours être ajouté pour visualise l'état du registre.



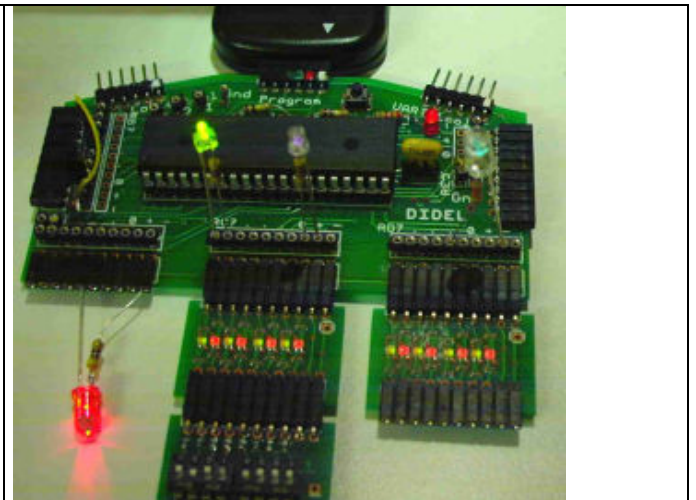
RC2 poussoir  
RC3 haut-parleur  
Un module vero est facile à câbler et reste clair  
Il sera cascadié avec le module moteur puisque c'est sur le même port.



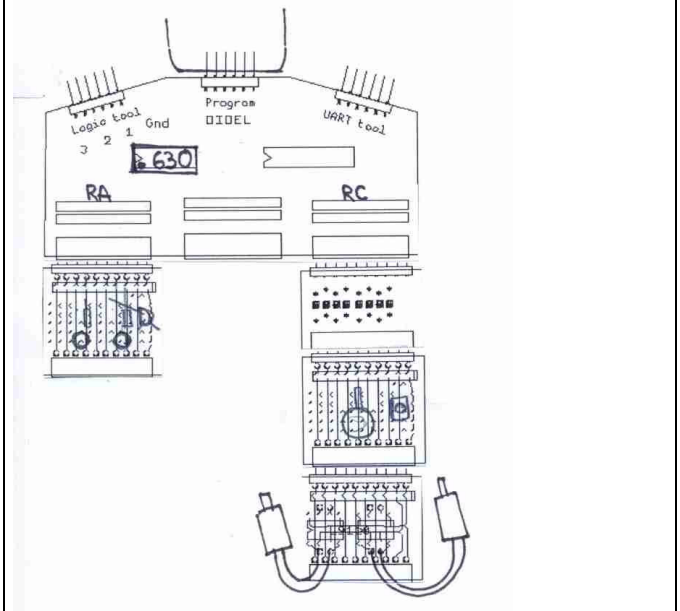
RA2 entrée module IR (récepteur infrarouge)  
RA0 RA1 LED bicolor  
RA4 RA5 LED bicolor  
(RA3 programmation)  
On câblera une douille 0,5mm pour aller sur l'analyseur logique du Picket2, ou un oscilloscope, au moment de la mise au point du décodage de la réception infrarouge.



Pour le développement, on utilise un 16F877 ou 16F884 avec le PortB à la place du portA du 630 du schéma (sur le 877 RA4 est open collector). Les Ports A, D et E sont libres pour la mise au point. On utilise RA0, RA1 en sortie comme signaux de synchronisation pour l'analyseur logique ou l'oscillo. On utilise le PortE avec un convertisseur D/A pour le suivi des machines d'état, le portD avec ses LEDs ou un convertisseur D/A pour observer des données, des compteurs.



La compatibilité avec le 630 (.If dans le programme) est testée en déplaçant les modules sur la carte M20 et en réassemblant. La mise au point est terminée, mais on peut devoir régler des paramètres et l'analyseur logique peut encore être utile.



Le transport du logiciel sur le Bimo n'a pas de raison de poser problème. Pas de fil temporaire à souder pour aider à la mise au point !

Un peu plus d'explications sous [www.didel.com/08micro/BimoTest.pdf](http://www.didel.com/08micro/BimoTest.pdf)

Autre exemple :



La documentation détaillée présente 7 cartes processeur et 10 modules d'entrée-sortie dans [www.didel.com/08micro/Microdules.pdf](http://www.didel.com/08micro/Microdules.pdf)

jd n 081029/090928