

Carte d'évaluation PICGénial 870 et extensions

La carte **PICGénial/870** utilise le processeur PIC 16F870 qui dispose d'une interface série et de 5 entrées analogiques. Un 16F872, 874 ou 876, avec plus de mémoire et plus de périphériques, peuvent aussi être utilisés. La carte **PICGénial** est optimisée pour apprendre à programmer et les exercices proposés sont identiques à ceux du **PICGénial** avec les 16F84, car le répertoire d'instruction est identique. Le **PICGénial/870** permet de résoudre plus d'applications, et les entrées/sorties supplémentaires ont été mises sur des connecteurs d'extension qui permettent de se relier à un PC (interface série) ou à des capteurs SPI ou I2C. Des modules moteurs sont à disposition.

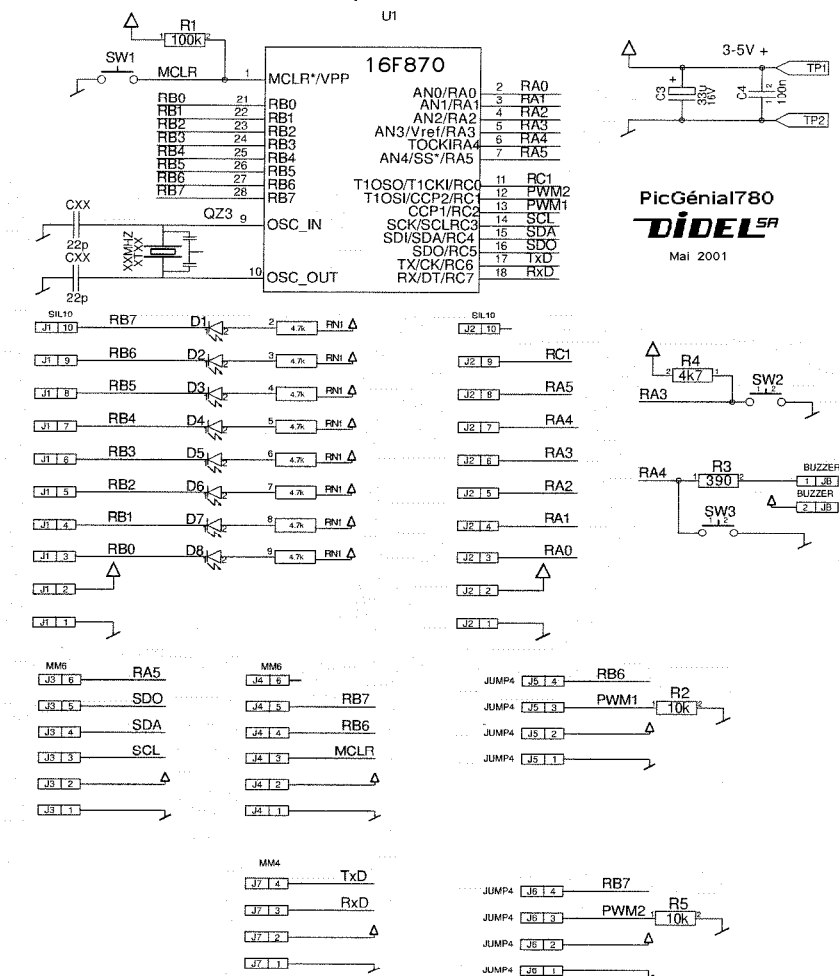


Fig 1: Schéma de la carte **PICGénial/870**

Pour l'exécution des programmes, la carte **PICGénial/870** doit être alimentée en 4 à 6V. Les programmes sont téléchargés depuis la carte PicgeDev (voir notice www.didel.com/picg/Picg.html).

Carte PicgExe de base

La carte se monte avec un 16F84 sur socle, un résonateur à 4 MHz, un poussoir de remise à zéro, 8 Leds sur le port B, un poussoir sur RA3, un autre poussoir en parallèle avec un buzzer sur RA4. Les programmes des brochures **PICGénial/1** à 9 et **PICGénial/AB** n'utilisent que ces éléments. Le montage du kit est décrit dans le document www.didel.com/picg/PicGenkit.doc "**PICGénial** Mise en oeuvre du kit". Deux pastilles (TP1 TP2) permettent de partir avec un serre-fil ou des douilles vers une alimentation extérieure, nécessaire pour se rendre indépendant du programmeur de PIC PicgeDev, ou pour disposer de plus de puissance. On peut aussi partir avec deux fils depuis un connecteur micromatch ou SIL, sur tous les connecteurs, la broche 1 est à la masse et la broche 2 au +5V.

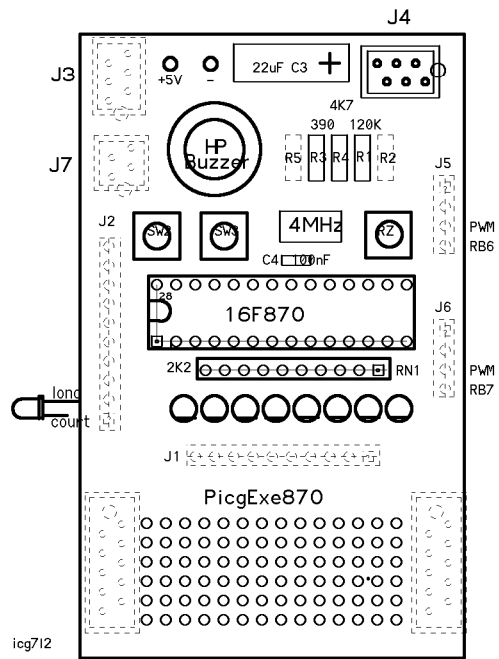
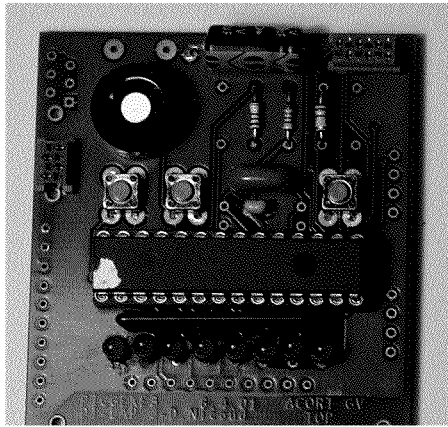


Fig 2: Plan de montage des composants de base et photo

Affichage sur les Leds et buzzer

Les Leds sont allumées lorsque les sorties correspondantes du processeur sont à zéro. L'instruction Not permet d'inverser en copiant la variable à afficher dans W, ensuite copié dans le port B (voir les exemples de programmes). De même, la lecture des poussoirs donne un zéro lorsque le poussoir est pressé. Le buzzer est un haut-parleur avec une membrane collée/relâchée selon l'état du port A bit4. Si ce port est en entrée, le 2e poussoir peut être lu; autrement, il faut éviter de le presser.

Le PIC est très robuste. Les sorties peuvent être court-circuitées et l'alimentation inversée un bref instant. Les surtensions sont plus dangereuses, et la programmation avec le bit de code protection actif bloque le circuit définitivement.

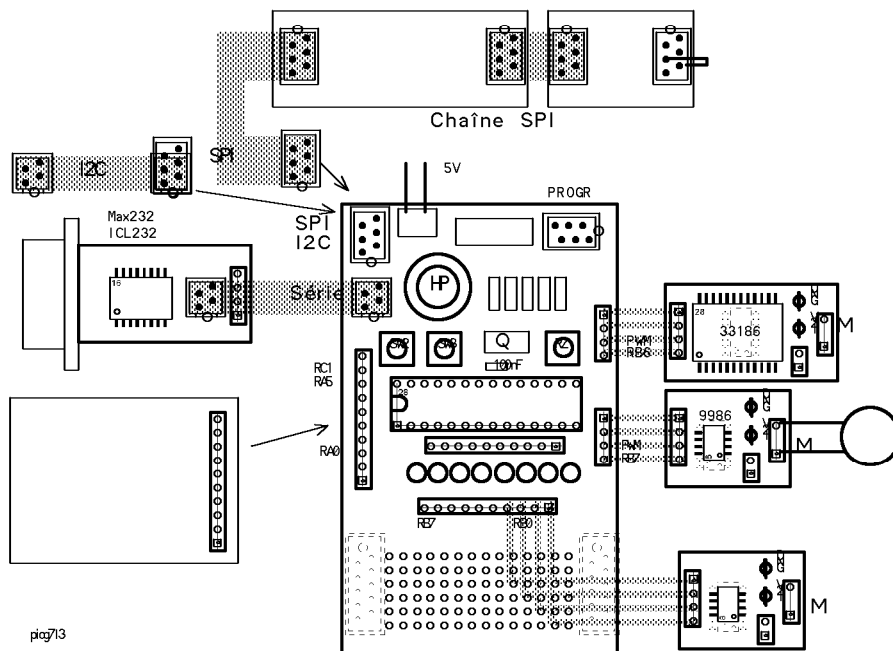


Fig. 1 Options autour de la carte PicgExe870

Connecteur sur le Port A

Le connecteur J2 est un SIL10 qui donne accès à toutes les lignes du port A et à RC0. Le port A peut être programmé pour avoir 5 entrées analogiques avec une résolution de 10 bits. On évitera de partir avec des longs fils vers les capteurs. Des soins particuliers doivent être pris pour éviter le bruit sur les masses et l'alimentation.

Connecteur sur le Port B

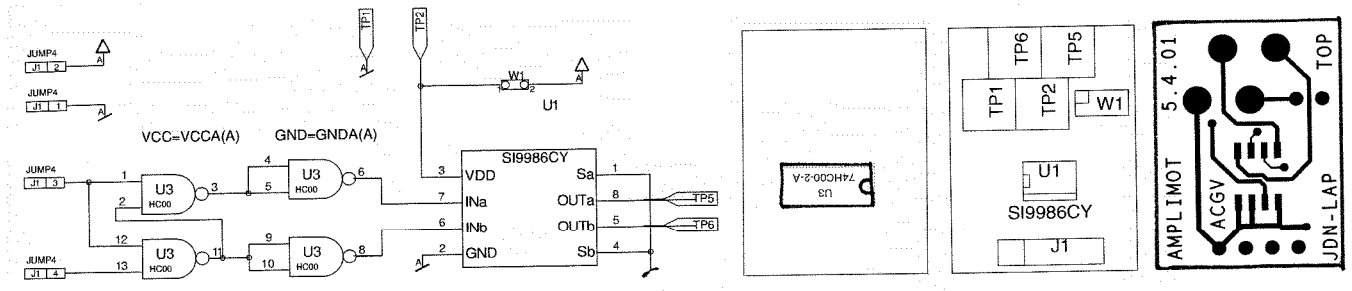
Le connecteur J7 est un SIL10 qui peut être ajouté pour prélever tous les signaux du port B. L'état des lignes du port B, en entrée ou en sortie, sont affichés sur les 8 LEDs si le réseau de résistances est en place. Il faut enlever ces résistances si on veut lire un signal qui a une résistance interne supérieure à 1 kOhm environ.

Connecteurs sur le port C

Les signaux du port C qui peuvent être programmés en PWM sont disponibles sur les connecteurs SIL4 J5 et J6. Les résistances pull-down sont alors ajoutées pour éviter que les moteurs soient activés (ligne PWM flottante) pendant le reset. Deux modules ampli moteurs utilisant des circuits modernes sont disponibles pour des applications robotiques. Des solutions bricolées avec des transistors sont documentées dans www.didel.com/doc/DocAmot.pdf.

Ampli de puissance pour moteurs

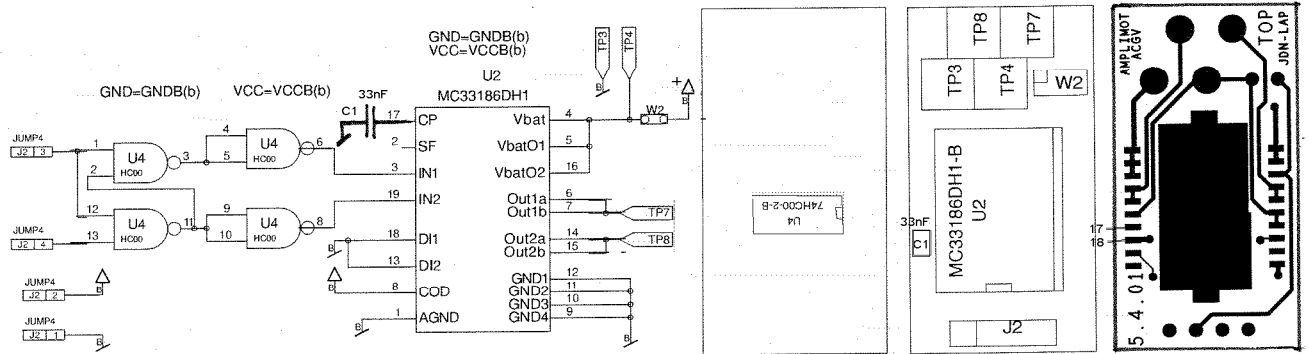
Le module Si8886 utilise le circuit Si9986 (www.vishay.com/docs/70007.pdf, qui est limité à 0.5A, 12V (2A en pointe). Pour être compatible avec les signaux PWM du 16F870/876, il a fallu ajouter un circuit d'encodage. Le plan de montage est donné dans la figure 4.



picg7LS *Fig. 4 Module moteur 0.5A Si9986*

Le moteur se branche sur TP5 TP6 (fiches Faston par exemple) et l'alimentation sur TP1 (Gnd) et TP2 (+12V max). Le jumper W1 permet d'alimenter pendant les tests avec le +5V du processeur, si les fils sont courts et l'alimentation adaptée au courant de démarrage du moteur.

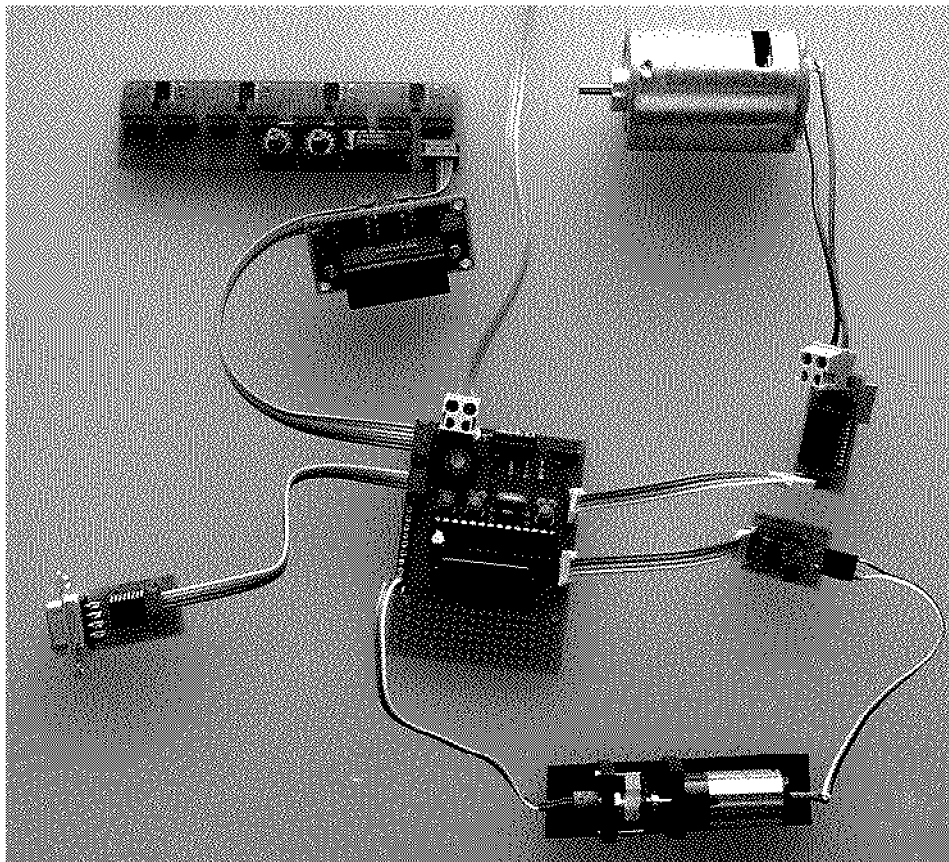
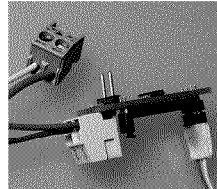
Le module MC33186 utilise le circuit Motorola MC33186 (e-www.motorola.com puis search 33186), qui est limité à 5A, 18V (10A en pointe). Pour être compatible avec les signaux PWM du 16F870/876, il a fallu ajouter un circuit d'encodage. Le plan de montage est donné dans la figure 5. Le condensateur C1 a été oublié sur le circuit imprimé. Il est patché entre les pins 17 et 18. Il peut être utile de rajouter un condensateur de 10-30 uF, 20V sur l'alimentation du moteur, et déparasiter les gros moteurs avec des condensateurs céramiques de 10-100 nF soudés directement entre les pins de sortie du moteur et la carcasse du moteur.



picg716 **Fig. 5 Module moteur 5A MC33186**

Le moteur se branche sur TP7 TP8 et l'alimentation sur TP3 (Gnd) et TP4 (+12V max). Le jumper W2 permet d'alimenter pendant les tests avec le +5V du processeur (robot sur des plots et moteur ne demandant pas d'à-coups de courant qui pourraient remettre à zéro le processeur).

Pour relier le moteur et son alimentation, il est conseillé d'utiliser des bornes à visser avec connecteurs (Distrelec 2001, page268, 120288 et 140292). Ces bornes à visser sont aussi conseillées pour l'alimentation du PicgExe870, mais il faut percer un trou de 1,3mm à 5mm du +5V.



Zone en câblage universel

La zone permet d'ajouter plusieurs circuits dans différents boîtiers, et des composants discrets (que l'on monte de préférence sur des socles de circuits intégrés). Deux connecteurs micromatch à 10 broches au plus peuvent être ajoutés pour partir vers l'application.