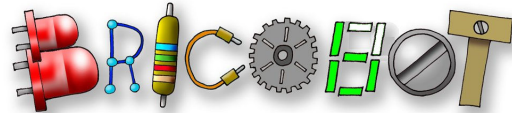




www.didel.com -- info@didel.com

www.didel.com/Petra.pdf lien provisoire

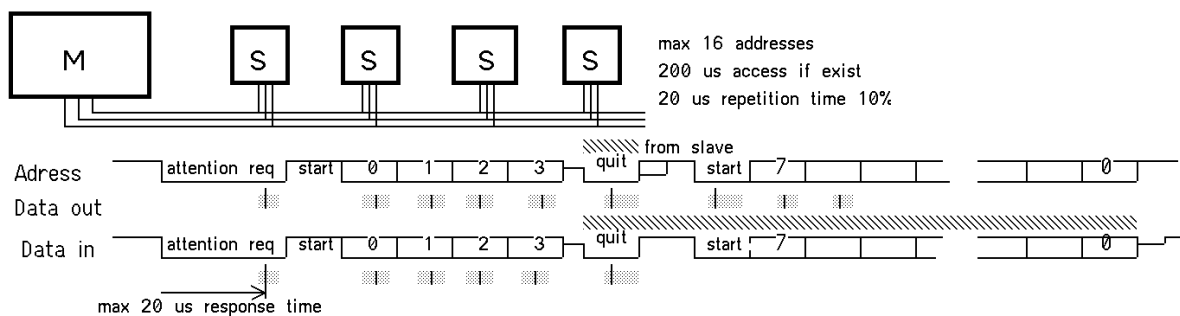


www.bricobot.ch -- info@bricobot.ch

Petra bus pour robots modulaires

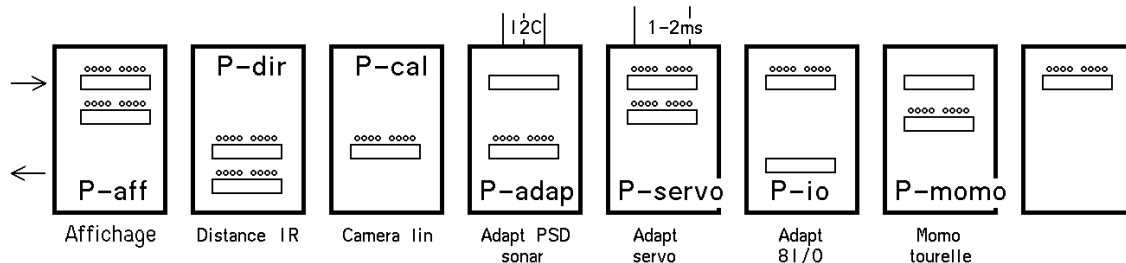
Le bus Petra a un concept original avec un maître qui interroge sans cesse des esclaves intelligents. Le maître a 16 registres qui sont l'image de 16 registres dans les esclaves, la remise à jour se fait toutes les 20 ms et utilise 15% du temps.

Les modules esclaves sont connectés par 3 fils et n'ont pas d'alimentation propre. Une caractéristique est qu'ils ont en principe une affichage qui montre la valeur des registres de communication. Cette contrainte de 3 fils éliminait les solutions traditionnelles, I2C, SPI, CAN, qui n'auraient de toute façon pas été implémentées efficacement sur les petits processeurs qui n'ont pas ces périphériques câblés.



Le maître envoie 16 adresses et écrit/lit le data, avorte si pas de réponse

Les esclaves sont réveillés à chaque adresse et répondent si concernés



Le principe du bus est le suivant : le maître envoie une impulsion et tous les esclaves doivent dans les 15 us entrer dans un programme qui se synchronise sur le start bit et lit l'adresse. L'esclave adressé quitte et lit ou envoie 8 bits en phase avec le maître (comme pour du RS232).

La programmation des esclaves, qui n'ont pas nécessairement des interruptions, se fait en appelant toutes les 12us une routine qui teste le bus. Si le bus est actif, la tâche en cours est en général abandonnée.

Les esclaves ont des adresses prédéfinies, mais ils sont tous facilement reprogrammables avec un Pickit2 ou autre.

Le logiciel au niveau du maître doit définir une table de 16 variables qui correspond aux adresses des modules. La routine d'interruption utilise le Timer0 et s'occupe de la mise à jour des registres en balayant toutes les adresses (accès toutes les 1.25 ms de durée 200 us si l'esclave répond). Toutes les 10ms, les registres de communication des modules et les registres image dans le maître sont mis à jour. Le processeur maître doit utiliser des routines de communications différentes à 4, 8, 16, 20 MHz.

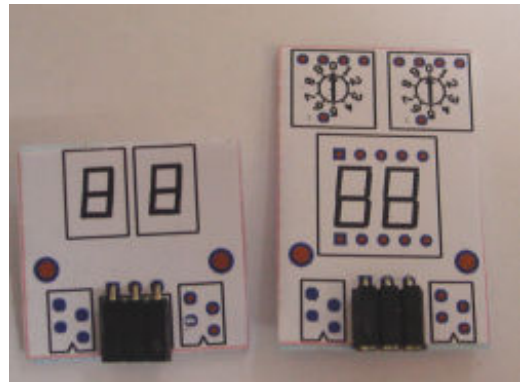
Le logiciel maître peut être écrit en C, mais il faut pouvoir insérer le module de communication par interruption optimisé en assembleur. Le maître peut être un AVR, mais il faut réécrire le logiciel de communication avec son timing précis. Avec un AVR à 8 MHz. il ne devrait pas avoir de problème.

P-aff Module affichage

Le module n'est pas encore défini. Il est prévu d'afficher 2 paramètres en alternance sur le même affichage si les deux paramètres sont mis à jour. L'affichage en binaire ou 7-segments est à décider.

Le lecture de 8 switch ou 2 commutateurs hexa pourrait être ajoutée, ou prévue dans un module de lecture seul.

16F630+HC164+HC165 si entrées

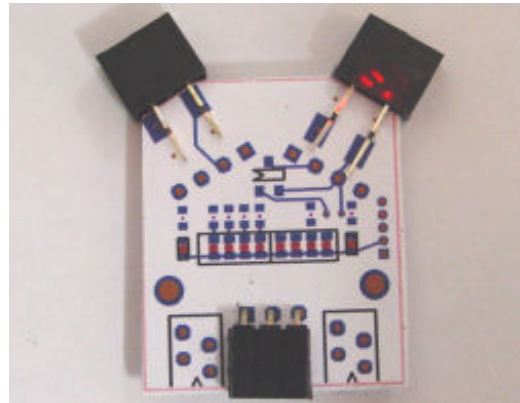


P-dir Module mesure de distance par IR

Le module est prévu pour 2 capteurs LIT301 qui ont une diode IR et un phototransistor efficace. Des TCRT1100 sont possibles.

La valeur de distance rendue est logarithmique et couvre 2 décades. La lumière ambiante est compensée en soustrayant la lumière non éclairée, mais le fonctionnement n'est acceptable qu'avec des néons.

16F630+HC164

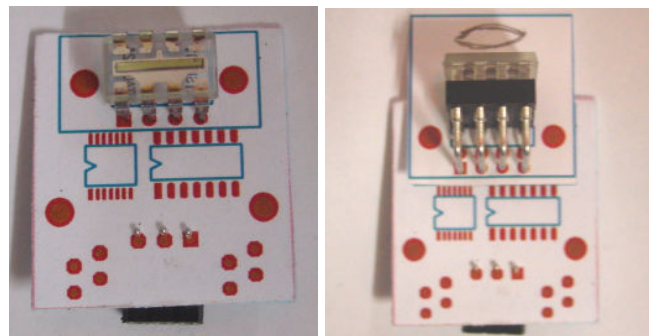


P-cal Module caméra linéaire

Le TSL3001 (ou dernière génération) est utilisé avec une lentille probablement cylindrique. Le profil lumineux est affiché sur les 8 (10 ?) leds. Deux paramètres sont envoyés : centre de gravité (donc position à viser) et intensité moyenne (donc distance).

L'orientation est soit perpendiculaire, soit parallèle au PCB.

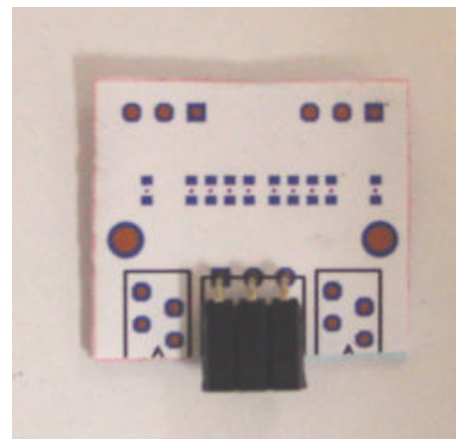
16F630+HC164



P-servo Module comande de servos

Le module reçoit deux valeurs 8 bits converties en impulsions de 1-2ms toutes les 20ms. (en déphasage avec les accès bus)

16F630+HC164



P-adap Module adaptateur

Le module est prévu pour communiquer avec un sonar standard ou PSD. Variantes connecteurs/soft à définir.

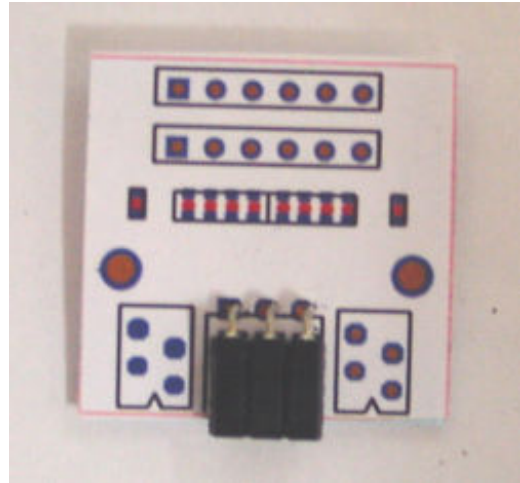
Comme module IO avec connecteurs et fixation adaptée au capteur-
16F676+HC164

P-io Module 8I/O

Le module a 8 I/O. Par défaut 4In 4 Out. Le 16F676 a des canaux analogiques peut être reprogrammé pour d'autres configurations, gérer des capteurs spéciaux, etc.

Directement utile pour ajouter des palpeurs et Leds. Deux ponts en H pourraient être ajoutés pour des moteurs, électro-aimants, etc.

16F676+HC164



P-Momo

Le module moteur Momo est équivalent à un servo, mais peut faire plusieurs tours. Il reçoit un ordre de position et vitesse (à définir)

16F676 Pas d'affichage dans le proto actuel.



Autre

Que faudrait-il encore ?

Un premier bus Petra a été développé avec Roman Schmidt en 1999, avec un protocole un peu différent. Le Swibot (16F84) avait deux connecteurs d'extension, utilisé ici pour une caméra linéaire (12F508), deux afficheurs 8 bits et une entrée 8 bits (12F508, 2x TB62075, HC165).

Le nom de Petra vient d'un endroit aux paysages merveilleux et temples creusés dans la roche, ou JDN a imaginé ce bus en soirées.

