

IO_PAR13.DLL

*Accès générique au port parallèle de
votre PC.*

Release V1.3 01/03/2000

Christophe Courché
Release V1.3 du 01/03/2000



cc@doguet.com

Sommaire :

INTRODUCTION.....	3
LE PORT PARALLÈLE DE VOTRE PC	3
COMMENT ACCÉDER AU PORT	4
LA DLL IO_PAR13.DLL.....	4
QUESQU'UNE DLL	5
FONCTIONS DISPONIBLES	5
COMMENT L'UTILISER.....	8
<i>En C</i>	8
<i>En VB</i>	8
LA PLATINE DE TEST.....	10
SHÉMA DE FONCTIONNEMENT	10
RÉALISATION.....	10
A VOUS DE JOUER !	10

ATTENTION : je ne pourrais être tenu pour responsable des dommages que pourrait occasionner l'usage de cette DLL. Cette DLL est réservée à un usage privé. Toute utilisation commerciale est soumise à autorisation de l'auteur.

Cette DLL est “ Freeware ” : si vous l'utilisez, merci de me tenir au courant !

Introduction

Les projets d'interface entre capteurs, moteurs, relais, ... sont de plus en plus nombreux et réalisables par tout possesseur de " Compatibles PC ". Ces montages utilisent soit le port " série ", soit le port " parallèle " pour dialoguer avec le monde extérieur. Le port parallèle du PC est intéressant car il fournit des entrées/sorties numériques bidirectionnelles (**) permettant de piloter directement des capteurs, relais et autres sans l'aide d'un microcontrôleur interface.

(** uniquement si port EPP, voir chapitre " Le port parallèle de votre PC ").

Le port parallèle de votre PC

Le port parallèle est originellement destiné à dialoguer avec une imprimante. Un bus de données 8 bits permet de piloter (D0/D7) un bon nombre de périphériques de construction amateur. Pour des montages nécessitant un plus grand nombre d'entrées et de sorties, toutes les autres " broches " du port peuvent être utilisées sans aucun problème.

Il est important de noter que le bus D0/D7 est normalement accessible en sortie, c'est-à-dire que vous pouvez changer la valeur du port mais pas la lire, sauf si votre PC possède une interface bidirectionnelle de type EPP. Le port EPP est maintenant implémenté dans tous les PC depuis les 486 (environ). Cependant, si votre PC n'est pas pourvu d'un port EPP, des cartes ISA économiques (100 Frs) sont disponibles sur le marché pour ajouter un nouveau port parallèle à votre PC, en mode EPP.

Le port parallèle se présente physiquement de la façon suivante :

Broche	Nom de la broche	Sortie	Entrée
1	STROBE	Oui	Oui
2	D0	Oui	Oui si EPP
3	D1	Oui	Oui si EPP
4	D2	Oui	Oui si EPP
5	D3	Oui	Oui si EPP
6	D4	Oui	Oui si EPP
7	D5	Oui	Oui si EPP
8	D6	Oui	Oui si EPP
9	D7	Oui	Oui si EPP
10	ACK		Oui
11	Busy		Oui
12	PE		Oui
13	SELECT		Oui
14	AUTOFEED	Oui	Oui
15	ERROR		Oui
16	INIT	Oui	Oui
17	SELECT IN	Oui	Oui
18 - 25	MASSE		

Note importante : n'utilisez pas votre port parallèle pour alimentation 5 volts stabilisée !. En effet, soyez très prudent lorsque vous brancherez un montage sur votre port.

Comment accéder au port

Le port est accessible via 3 registres. Ces registres permettent de contrôler toutes les “broches” de votre port parallèle. Ces registres ne seront pas décrits ici car une solution plus pratique vous est proposée dans le paragraphe suivant.

La dll io_par13.dll

La dll “io_par13” est une bibliothèque de fonctions simples vous permettant d'accéder au port parallèle de votre PC très simplement.

Bien entendu, il faudra utiliser un langage de programmation pour dialoguer avec cette bibliothèque de fonctions. Cependant un programme de test autonome est fourni et il est capable de piloter votre port à des fins de tests.

Qu'est-ce qu'une DLL ?

Une DLL est une bibliothèque de fonctions accessible par plusieurs programmes de nature différente en même temps.

Il suffit d'appeler dans son programme les fonctions présentes dans la DLL pour qu'elles s'exécutent.

L'avantage c'est que vous pouvez ignorer complètement ce qu'il se passe dans cette DLL ! la DLL est à considérer comme une boîte noire, on demande une fonction et on attends le résultat.

Fonctions disponibles

Ci-dessous la liste des fonctions disponibles dans la version 1.3 de la DLL :

Description des paramètres d'entrés

- Adresse du port // : 378 pour le port 1 (standard) et 279 pour le port 2
- Valeur a écrite : valeur 8 bits en décimal, hexadécimal ou binaire représentant la valeur à écrire sur le port D0/D7.

Description des paramètres de sorties

- Code retour : si tout c'est bien passé le code retour sera toujours 0
- Valeur lue : 0 = entrée en “ état bas ”, 1 = entrée en “ état haut ”

Remarque très importante : toutes les entrées mêmes inversées matériellement renverront 1 si la broche est connectée à un potentiel “ haut ”, 0 si la broche est connectée à la masse.

Write_Port_Bit

Paramètre(s) d'entrée(s) Adresse du port //, valeur à écrire.

Paramètre(s) de sortie(s) : Code de retour.

Description : Permet d'écrire un octet sur le port //.

Read_Port_Bit

Paramètre(s) d'entrée(s) Adresse du port //.

Paramètre(s) de sortie(s) : Valeur lue

Description : Permet de lire un octet sur le port //, uniquement si port EPP.

Modify_Port_Bit_On

Paramètre(s) d'entrée(s) Adresse du port //, valeur à écrire.

Paramètre(s) de sortie(s) : Code de retour

Description : Permet de passer en " état haut " des " broches " du port D0/D7, sans modifier l'état des broches non désirés.

Modify_Port_Bit_Off

Paramètre(s) d'entrée(s) Adresse du port //, valeur à écrire.

Paramètre(s) de sortie(s) : Code de retour

Description : Permet de passer en " état bas " des " broches " du port D0/D7, sans modifier l'état des broches non désirés.

Strobe_On

Paramètre(s) d'entrée(s) Adresse du port //

Paramètre(s) de sortie(s) : Code de retour

Description : Permet de passer en " état haut " la broche Strobe.

Strobe_Off

Paramètre(s) d'entrée(s) Adresse du port //

Paramètre(s) de sortie(s) : Code de retour

Description : Permet de passer en " état bas " la broche Strobe.

Read_Strobe

Paramètre(s) d'entrée(s) Adresse du port //.

Paramètre(s) de sortie(s) : Valeur lue

Description : Permet de lire l'état de la broche Strobe.

Idem pour les Broches Init (Init_On, Init_Off, Read_Init), Select_In (Select_In_On, Select_In_Off, Read_Select_In), Autofeed (Autofeed_On, Autofeed_Off, Read_Autofeed)

Les broches Ack, Busy, Pe, Select, Error, peuvent être lues uniquement

Exemple pour Ack:

Read_Ack

Paramètre(s) d'entrée(s) Adresse du port //.

Paramètre(s) de sortie(s) : Valeur lue

Description : Permet de lire l'état de la broche Ack.

Pour le bus D0/D7, 8 fonctions pour passer en “ état haut ” les broches D0/D7, , 8 fonctions pour passer en “ état bas ” les broches D0/D7 et , 8 fonctions pour lire l'état des broches D0/D7

Exemple pour D0 :

D0_On

Paramètre(s) d'entrée(s) Adresse du port //

Paramètre(s) de sortie(s) : Code de retour

Description : Permet de passer en “ état haut ” la broche D0.

D0_Off

Paramètre(s) d'entrée(s) Adresse du port //

Paramètre(s) de sortie(s) : Code de retour

Description : Permet de passer en “ état bas ” la broche D0.

Read_D0 (Uniquement si port EPP)

Paramètre(s) d'entrée(s) Adresse du port //.

Paramètre(s) de sortie(s) : Valeur lue

Description : Permet de lire l'état de la broche D0.

Etc ... pour D1, D2, ... D7

Comment l'utiliser

En C

Pour les développeurs C, il existe plusieurs méthodes, soit vous appeler la DLL ou vous utiliser la LIB fournie (pour le “ linkage ”) et le fichier io_par13.h pour les déclaration.

Le fichier io_par13.h

Le fichier io_par13.h contient tous les prototypes des fonctions disponibles dans la DLL.

En VB

L'utilisation d'un langage de “ haut niveau ” comme visual basic permet à tous les utilisateurs de développer des petites (ou grosses) applications sans apprendre toute la subtilité d'un langage comme le C ou l'assembleur. Visual basic permet aussi de créer des interfaces graphiques très simples à gérer.

Les déclarations (modules)

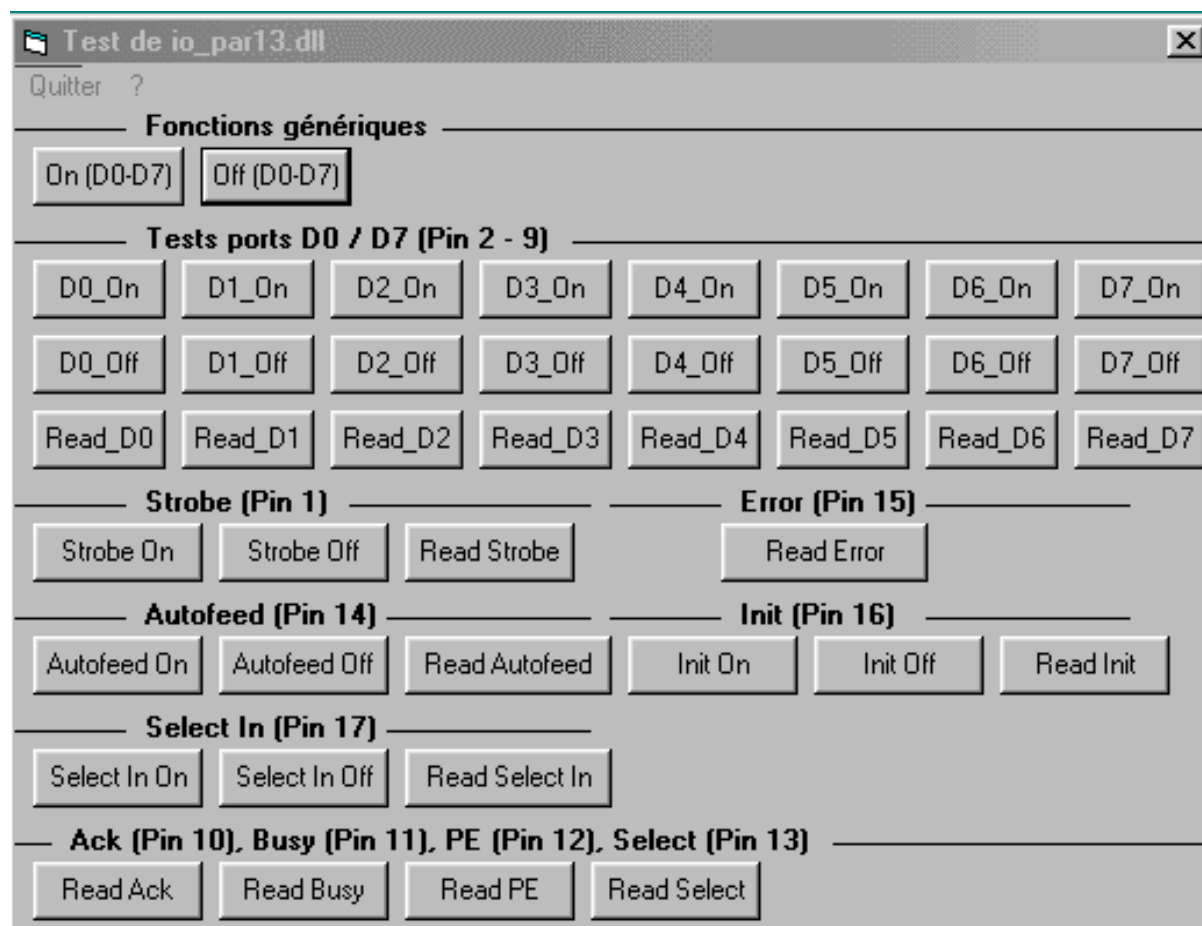
Pour utiliser la DLL io_par13, il suffit d'insérer un “ module ” à votre projet. C'est tout. Un module type est fourni avec la distribution standard.

Utilisation des fonctions

Un exemple est plus rapide qu'un long discours, regarder le projet “ test_io ” fourni, ce petit programme permet de tester toutes les fonctions de la DLL.

Le programme de tests

Si vous ne possédez pas Visual Basic, une version autonome du programme est fourni pour commencer à tester votre port parallèle (port LPT1).

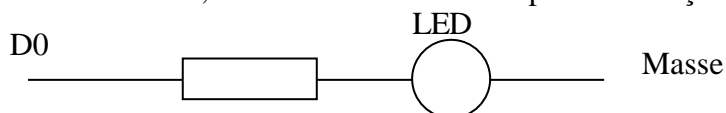


La platine de test

Il est possible de réaliser très facilement une petite platine de test, qui illustre parfaitement toutes les fonctions fournies avec la DLL io_par13.

Shéma de fonctionnement

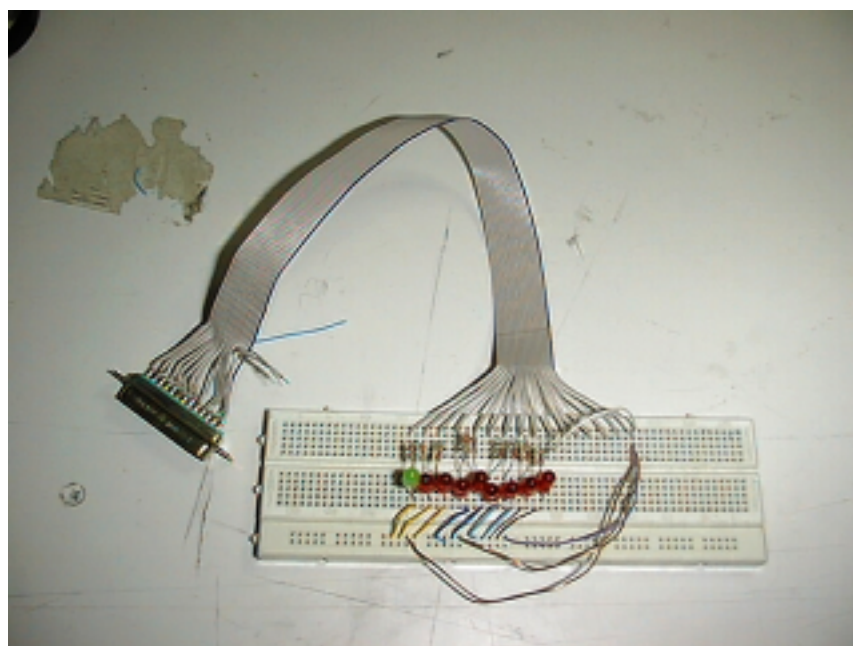
Pour tester une sortie, il suffit de connecter un port de la façon suivante :



Pour tester une entrée, il suffit de connecter l'entrée à un potentiel haut (5v) ou à un potentiel bas (masse).

Réalisation

Sur une platine “ Lab ” par exemple :



A vous de jouer !

Pour tous renseignements complémentaire, idées etc...

Christophe Courché : Email : cc@doguet.com