

## Entrées-sorties d'un microcontrôleur

Les fabricants de microcontrôleurs, Microchip, AVR, etc, documentent les interfaces de sortie avec des figures comme ci-contre (figure 1). On trouve ailleurs de compteurs, des registres à décalage.

Ce document introduit les notions importantes concernant l'électronique, les circuits logiques des interfaces microprocesseurs, sans aller dans les détails que l'on trouve dans des livres comme J.D. Nicoud "Circuits numériques pour interfaces microprocesseur" (Masson). Ce livre n'est plus disponible en librairie. La plupart des figures sont reprises de ce livre.

### Electricité – ce que tout le monde sait (?)

Mesure		Unité		
Tension	U	Volt	V	mV
Courant	I	Ampère	A	mA
Résistance	R	Ohm	$\Omega$	k $\Omega$ , M $\Omega$
Capacité	C	Farad	F	$\mu$ F, nF, pF
Puissance	P	Watt	W	mW
Travail/énergie	E	Joule	J	

Loi d'Ohm	$U = R * I$
Loi de Coulomb	$Q = C * U = 1 \text{ A s}$
Travail/énergie	$1 \text{ J} = 1 \text{ N m} = 1 \text{ A s}$
	Accu 1mAh à 3.6V = ~13J
Puissance	$1 \text{ W} = 1 \text{ V A} = 1 \text{ J / s}$

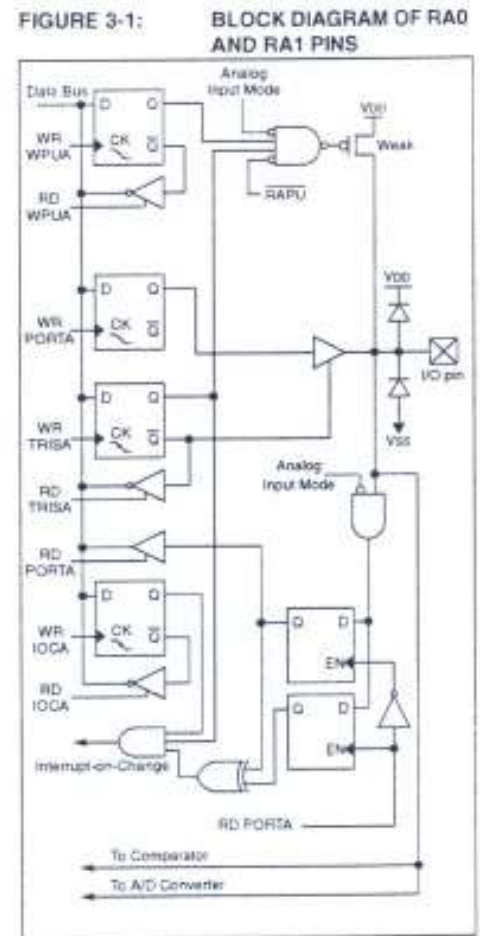


Figure 1

### Résistances

Une résistance Pull-down impose une tension proche du zéro et évite que la ligne soit flottante, donc avec une valeur logique mal définie.	Une résistance pull-up impose une tension proche de V+. Les microcontrôleurs ont souvent des pull-up programmables.	Un diviseur de tension définit une tension intermédiaire.	Une résistance série crée une chute de tension proportionnelle au courant, donc limite ce courant.

Figure 2 a b c d

A propos de pull-up programmables, on voit dans la figure 1 comment c'est fait dans un PIC. La résistance est un mauvais transistor (weak) activé si RAPU (RA Pull-Up) est à zéro (ainsi que d'autres conditions qui interviennent sur la porte ET). La commande par logiciel dépend du fabricant (voir plus loin)

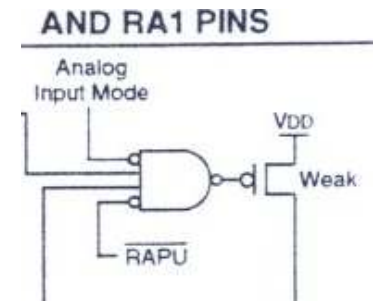


Figure 3

### Condensateurs

Une capacité de découplage réduit les pointes négatives de tension lors de la commutation des signaux. On met souvent 100 nF sur l'alimentation, à proximité du processeur. On évite d'utiliser des filtres passe-bas et passe-haut ; retarder ou dériver un signal se fait par programmation.

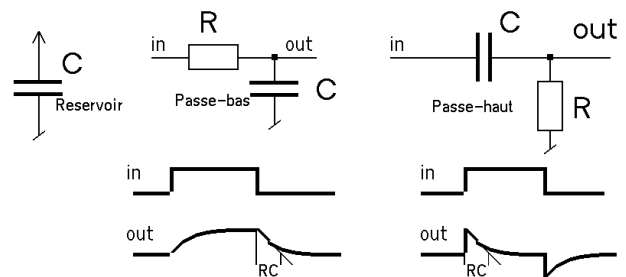


Figure 4

On met aussi un condensateur entre les bornes des moteurs et leur châssis, pour couper un peu les surtensions qu'ils génèrent.

### Diodes et diodes de protection

Une diode ne laisse passer le courant que dans un sens. Comme on le voit sur la figure 1, les sorties sont protégées par des diodes. Si la tension est de 6V et que l'on applique 6V sur une entrée, la diode du haut va conduire, chauffer et se détruire si le courant est important. Si l'alimentation du microcontrôleur est à l'envers, ces diodes conduisent et le circuit chauffe. Si on coupe le courant à temps, ce n'est pas destructeur.

### Niveaux des entrées et sorties

Le grand principe des circuits logiques est que les sorties ont pour les états 0 et 1 une tension proche du 0 et V+. Les entrées ont un domaine d'acceptation plus large, ce qui fait qu'il y a une marge appelée immunité au bruit, utile quand les conducteurs sont longs, passent à côté de signaux qui oscillent.

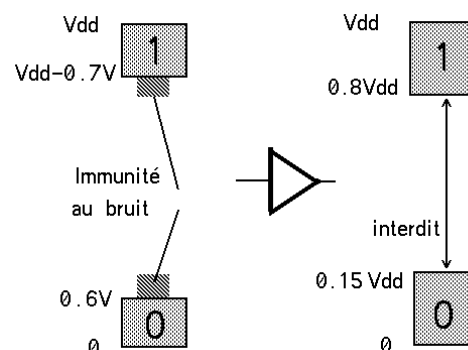


Figure 5

### Entrée en Bascule de Schmitt (Schmitt-trigger)

La bascule de Schmitt présente une hystérèse qui fait que les petites fluctuations du signal ne passent pas. C'est l'idéal pour toute ligne connectée à un signal extérieur. Sur un PIC, seuls les timers ont une entrée en Schmitt trigger.

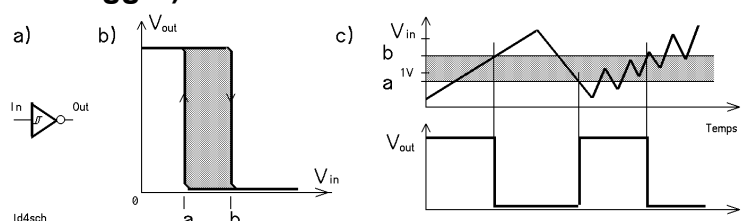


Figure 6

### Entrée non connectée

Une entrée "en l'air" (non connectée) a une tension mal définie. Il suffit d'approcher la main pour capter des effluves qui font changer l'état logique, et si la main ou le fer à souder a une tension électrostatique élevée, l'entrée peut être détruite. Les bracelets et fils à la masse sont indispensables dans les pays chauds et secs, moins en Europe.

## Etage de sortie

Une sortie normale est en "push-pull" avec un transistor P actif si la sortie est à l'état 1, et un transistor N actif pour l'état 0. Si la sortie est surchargée, la résistance interne limite le courant et l'échauffement peut être destructeur. Cette résistance est assez importante sur une sortie de microcontrôleur. Elle est plus élevée vers le +. On préfère donc câbler une diode lumineuse pour qu'un état 0 soit actif, il y a moins de perte. Les valeurs données dans la figure 6 dépendent de la tension d'alimentation.

Pour augmenter le courant, on peut connecter plusieurs sorties en parallèle, et leur assigner naturellement le même état logique, Un moteur miniature de 30 Ohm peut ainsi être commandé par un port 8 bits sans amplis supplémentaires. Les 4 bits de poids faible et les 4 bits de poids fort sont utilisés pour une commande bidirectionnelle.

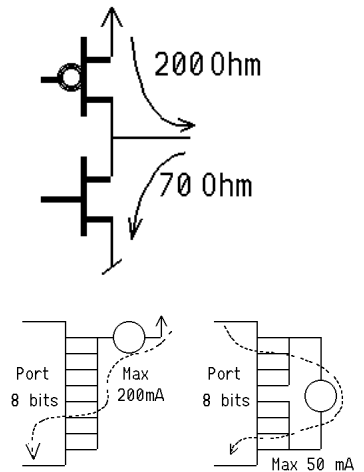


Figure 7

## Collecteur ouvert

Le transistor P manque en général sur une des lignes de programmation des microcontrôleurs. Dans ce cas, on peut écrire un 0, mais écrire un 1 laisse la ligne dans un état flottant, comme pour une entrée. Pour imposer l'état 1, il faut ajouter une résistance pull-up.

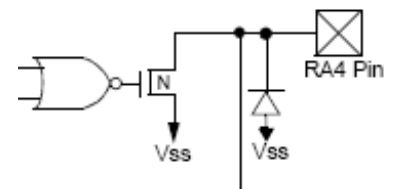


Figure 8

## Alimentation des microcontrôleurs

Les microcontrôleurs acceptent le plus souvent une tension d'alimentation de 2 à 6V, sauf les nouvelles générations très performantes qui travaillent en 3 ou 3.3V seulement. Les régulateurs de tension comme le 7405 sont encore beaucoup utilisés pour alimenter la carte ; leur chute de tension est de 1.5V, ce qui veut dire que si l'on veut un 5V bien régulé, la tension d'entrée ne doit pas être inférieure à 6.5V. Les circuits LDO (Low DropOut) ont de meilleures caractéristiques et sont disponibles, par exemple pour le MCP1700, dans des tensions de 1.2, 1.8, 2.5, 3.0, 3.5 et 5V (boîtier SOT23, 200mA max).

Il arrive de plus en plus fréquemment qu'un processeur 5V doive piloter un circuit 3.3V et inversement. Un signal 3V ne commande pas de façon fiable une entrée d'un circuit 5V. Un signal 5V peut générer un courant destructeur sur l'entrée d'un circuit 3V.

## Processeur 5V et interface 3V

L'amplitude du signal 5V en sortie est réduite par un diviseur de tension. En sens inverse, la solution n'est pas plus simple que pour un signal bidirectionnel qui utilise un transistor BSS138 orienté correctement. Des circuits intégrés existent, par exemple le MAX3013 contient 8 ampli bidirectionnels dans un boîtier TSSOP20 ou QFN20.

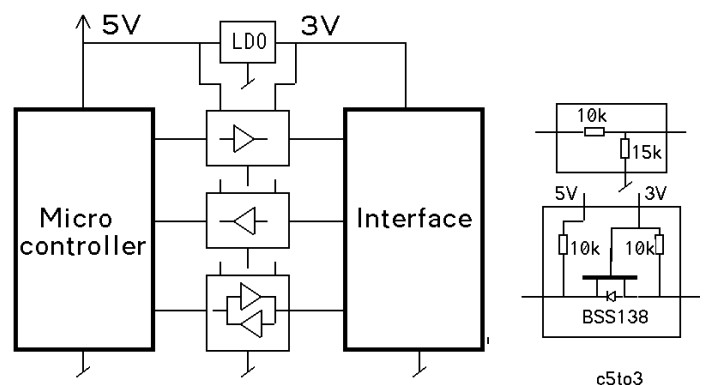


Figure 9

## Processeur 3V et interface 5V

On retrouve les mêmes schémas, Le signal 5V de l'interface peut être réduit par un diviseur de tension, et pour un signal bidirectionnel, la schéma avec le BSS138 convient.

## Amplificateur

Pour amplifier le courant de sortie, on utilise un transistor MOS. Vérifier que la tension d'entrée est compatible pour commander du 12V ou plus. Pour une commande bidirectionnelle par 2 bits du processeur, un grand choix de driver moteurs existe. Pour un courant de 100 mA, un boîtier d'inverseurs 74F245 est la solution la plus économique.

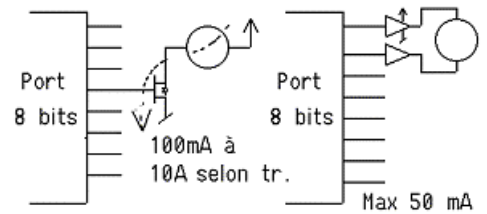


Figure 10

A noter qu'un circuit en pont comme le Si9986 a deux comportements différents: pour les états 00 et 11. Dans l'état 00, les transistors conduisent et le moteur est comme court-circuité, c'est-à-dire qu'il va s'arrêter rapidement. Dans l'état 11, le moteur est "ouvert", libre, et n'est freiné que par le frottement.

## Charge inductive

Si on commande un solénoïde, la surtension à la commutation est importante. Il faut utiliser des transistors avec une diode de protection et rajouter une diode en parallèle avec la charge inductive. Les ponts pour commander des moteurs bidirectionnels ont des diodes de protection. On ajoute souvent un condensateur pour éviter les fronts raides des signaux de puissance commutés, donc la transmission de pointes de tension par les lignes d'alimentation.

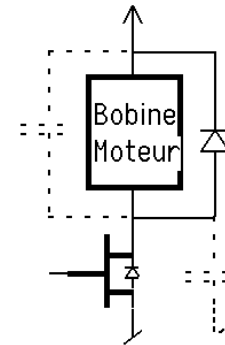


Figure 11

## Protection des sorties et inversion de l'alimentation

On voit dans la figure 1 deux diodes sur la broche d'entrée-sortie. Si la tension sur la broche est supérieure à 5.7V, la diode supérieure conduit (et chauffe). Si elle est inférieure à -0.7V, c'est la diode inférieure qui protège. Si on inverse la tension les deux diodes font court-circuit. Si on surveille la température en mettant la main sur les circuits, et que l'on coupe le courant avant de s'être brûlé les doigts, le processeur est le plus souvent encore opérationnel.

## Portes à 3 états et commande de direction

Un microcontrôleur a des registres de sortie dans lesquels on peut toujours écrire. Des amplificateurs à 3 états (que l'on appelle aussi "passeurs" relient ce registre aux sorties. La commande du 3<sup>e</sup> état (l'état flottant) se fait par un registre de direction, qui permet de commander chaque ligne individuellement.

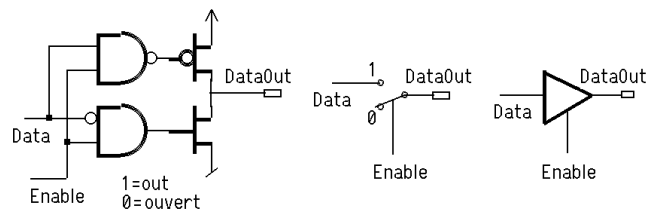


Figure 12

## Cas des PICs 10,12,16F

Dans la forme de base, les PICs ont deux registres appelés Port et Tris (tristate)  
Si un bit du registre Tris est à 1, le passeur n'est pas activé et un signal appliqué sur la broche ne sera pas en concurrence avec le passeur. Si le processeur lit l'adresse du port, il verra ce signal.  
Si le bit du registre Tris est à 0, le passeur essaie d'imposer son état, et il faut éviter de court-circuiter la sorties avec un signal actif qui chercherait à imposer son état.

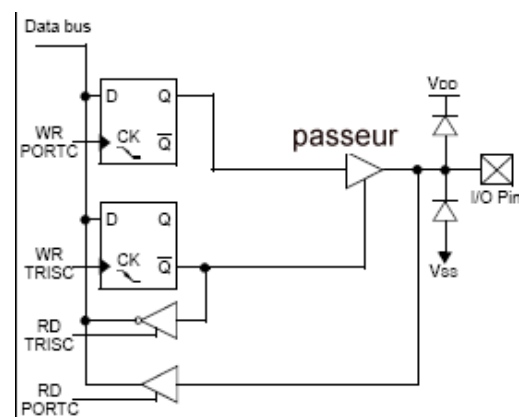


Figure 13

A noter que si on écrit un 1 et que l'on relit un 0, le logiciel peut en déduire que la pin est court-circuitée à la masse. Mais il faut se souvenir que l'on a écrit un 1; on ne peut pas relire directement le registre data.

Avec ce schéma, il faut faire attention avec toutes les instructions à un opérande qui font une lecture-modification-écriture. Si un bit est en entrée, il va être lu, modifié et recopié dans le registre de sortie, ce qui peut être gênant si on repasse en sortie : la valeur préparée auparavant peut avoir été perdue.

### Cas des AVR et Pic 18F

Les entrées-sorties des AVR sont plus riches que celles des PICs. Les PIC 18F et plus récents ont une structure similaire. Un port en entrée permet de relier le registre de sortie directement. Le port de lecture de la pin s'appelle Pin, ce qui est clair.

De plus, un bit dans un registre décide s'il y a ou non des pull-up sur toutes les lignes en entrée et une astuce avec l'état du Port permet de décider individuellement si la pull-up est active.

Le problème du read/modify/write n'existe pas, puisque la modification du port ne concerne pas les lignes extérieures. Une différence importante entre AVR et PIC est que sur AVR pour les bits de direction, un 0 signifie entrée et un 1 sortie.

La famille Texas MSP a une même disposition de registres avec des noms différents.

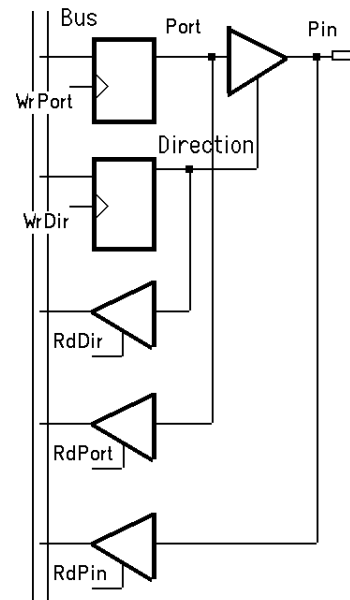


Figure 14

### Pull-up

Si une ligne est en entrée, une pull-up est nécessaire pour éviter un état flottant si elle est "en l'air". Un transistor interrupteur commandé par une porte ET (il faut que l'on soit en entrée et que l'on veuille une pull-up). Sur les PICs on voit un registre appelé WPU. Sur les AVR, il n'y a pas de registre supplémentaire. Si on lit la pin, le registre de sortie est utilisé pour mettre en service les pull-up (l'état "1" active la pull-up).

### Conditions initiales

Au reset, la direction est en entrée et l'état des registres ports n'est pas modifié. Donc à l'enclenchement l'état est imprévisible et il est recommandé de mettre à 0 tous les registres et variables (ce que le compilateur insère parfois systématiquement).

### Fonctionnalité multiple.

La plupart des pins sont programmables pour jouer des rôles différents, entrée analogique, liaison à un quartz, interfaces spéciaux, etc. Des registres commandent des aiguillages pour relier les pins aux circuiteries spécialisées du microcontrôleur. On le voit dans la figure 1. Dans le mode Analog, une porte ET inhibe une partie de la logique et l'information va seulement vers le convertisseur.

### Reset et brown-out

A la mise sous tension, un délai est programmé et le reset est activé. Si la tension baisse (brown-out) en dessous d'une valeur programmable dans un bit de configuration, il y a également reset, et maintien du reset jusqu'à ce que la tension soit remontée à une valeur supérieure, plus un temps d'attente. Dans les microcontrôleurs récents, les niveaux de tension et les délais sont programmables.

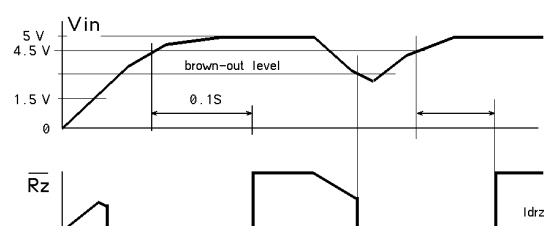


Figure 15

## Portes logiques

Les tables de vérité des portes logiques sont bien connues. Voir [www.didel.com/pic/Bases.pdf](http://www.didel.com/pic/Bases.pdf) si vous n'êtes pas à l'aise avec la logique et l'arithmétique binaire. Les symboles ci-contre suivent la norme US, plus lisible que les symboles avec rectangle. Remarquez les deux schémas équivalents suite à la loi de De Morgan. Les ronds d'inversion, qui signifient "inversion" ou "inhibition" sont très importants pour la lisibilité du schéma, on le voit sur la porte ET à 4 entrées de la figure 1.

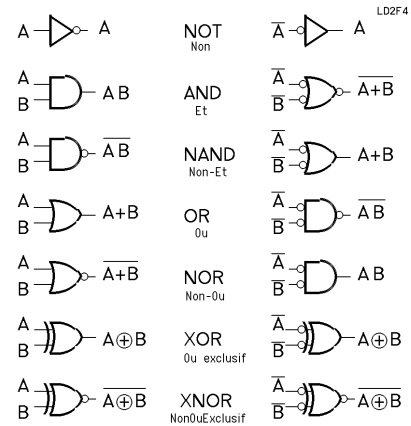


Figure 16

## Bascules

Les bascules, et ce que l'on construit avec, changent d'état au front montant ou descendant de l'horloge. La convention du rond d'inversion n'est pas toujours respectée. Si le chargement est statique (pendant la durée de l'impulsion), on a un verrou (latch).

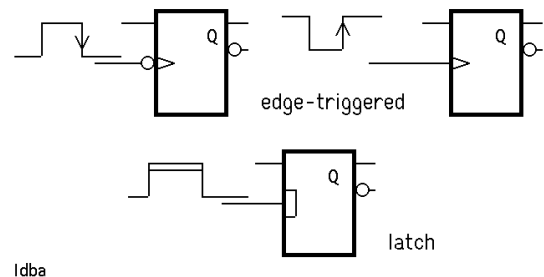


Figure 17

## Multiplexeurs

Les multiplexeurs sont utilisés pour sélectionner une entrée parmi plusieurs, comme pour le choix d'un canal analogique. On voit souvent dans les schémas équivalents le symbole de l'aiguillage à 2 ou plus de positions.

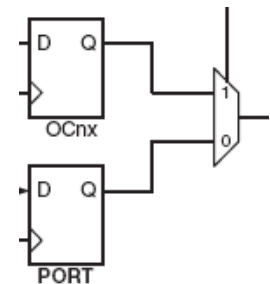


Figure 18

## Registres et compteurs

Les noms expriment bien la fonction réalisée.

**PIPO** Parallel In Parallel Out

**PISO** Parallel In Serial Out

**SIPO** Serial In Parallel Out

**SISO** Serial In Serial Out

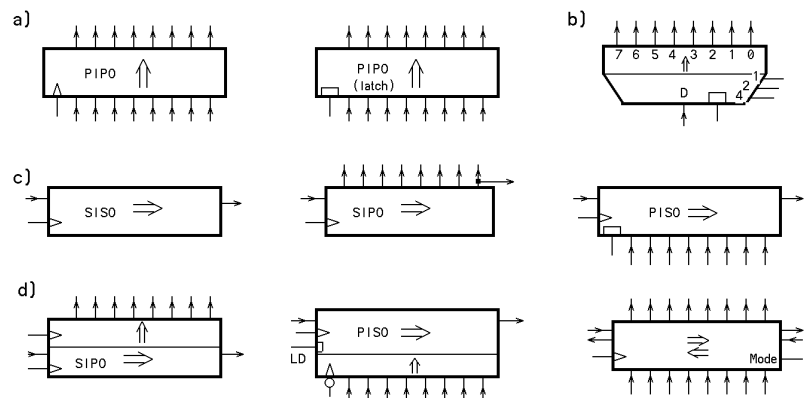


Figure 19

Les registres sont associés à des aiguillages pour avoir des longueurs variables (les transmissions série se font en 7, 8 ou 9 bits).

Les compteurs sont toujours binaires, parfois précédés d'un prédiviseur, qui est un compteur associé à un aiguillage.

Un compteur 16 bits initialisé ou lu 8 bits à la fois nécessite des registres parallèles supplémentaires pour garder la cohérence des données.

## Logique d'interruption

Le schéma que l'on voit pour la logique d'interruption comporte quantité de portes, de bascules d'activation (enable) et de bascules sémaphores (flag). Un bit (interrupt enable)

active ou désactive d'un coup toutes les interruptions. La logique hierarchise les interruptions activées.

### **Blocs fonctionnels**

Les convertisseurs A/D, les interfaces UART, SPI, I2C ont une logique assez complexe pour être compatible avec diverses variantes d'utilisation. C'est un travail de patience de bien comprendre et tester pas à pas pour avoir un comportement parfaitement fiable sous interruptions.

### **Capteurs et circuits extérieurs pour réseaux, affichages, etc**

Le schéma logique interne des circuits qui l'on place autour des processeurs est le plus souvent très mal documentés. Il faut se contenter du séquençement des signaux et de compléments d'information aux data sheets, trouvés sur internet.

Les registres à décalage sont souvent utilisés pour augmenter le nombre d'entrées (claviers) ou de sorties (Leds, affichages) .

Notre documentation se trouve sous

[www.didel.com/diduino/CommandeLeds.pdf](http://www.didel.com/diduino/CommandeLeds.pdf)

[www.didel.com/diduino/CommLedsC.pdf](http://www.didel.com/diduino/CommLedsC.pdf)

[www.didel.com/diduino/CommLedsPinguino.pdf](http://www.didel.com/diduino/CommLedsPinguino.pdf)

Sur les capteurs robotique, citons quelques documents (parfois anciens) :

[www.didel.com/doc/sens/Doclr.pdf](http://www.didel.com/doc/sens/Doclr.pdf)

[www.didel.com/doc/sens/Doclrt.pdf](http://www.didel.com/doc/sens/Doclrt.pdf)

[www.didel.com/doc/sens/DoclrLog.pdf](http://www.didel.com/doc/sens/DoclrLog.pdf)

[www.didel.com/doc/sens/DocSharp.pdf](http://www.didel.com/doc/sens/DocSharp.pdf)