

## Transducteurs

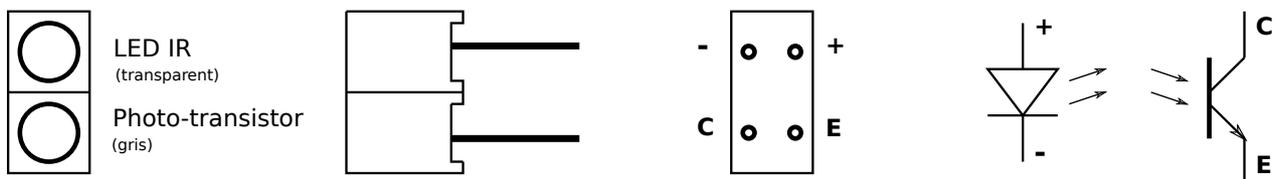
En introduction à ce cours, on présentait le microcontrôleur comme « un ordinateur sans clavier, sans écran, mais avec des tentacules pour interagir avec l'extérieur ». Ces tentacules, on les appelle des transducteurs : une information est transformée d'une grandeur physique à une autre.

Pour ce qui entre dans le système, on parle de capteurs. Pour ce qui sort, on parle d'actuateurs. Les premiers capteurs que nous avons utilisés sont les boutons-poussoirs, qu'il est particulièrement simple de relier à un microcontrôleur.

D'autres capteurs sont plus complexes à mettre en oeuvre. À titre d'exemple, nous présenterons ici un capteur de distance à infrarouges.

## Capteur de distance IR

Ce capteur se présente de la manière suivante :



Il comporte une LED infra-rouge (encapsulée dans du plastique transparent) et un photo-transistor (encapsulé dans du plastique sombre). Les deux éléments ne doivent pas se « voir » l'un l'autre, c'est pourquoi ils sont placés parallèlement dans de petits tubes en plastique noir.

Le fonctionnement de ce composant est le suivant : la lumière infrarouge de la LED va être réfléchiée par les objets proches. Le photo-transistor va détecter la lumière réfléchiée. Plus un objet est proche, plus l'intensité sera grande. On a donc une mesure indirecte de la distance. Il ne s'agit pas d'une mesure précise, comme on pourrait l'obtenir avec des ultra-sons ou des lasers. Mais le faible coût de ce composant le rend intéressant dans de nombreuses applications où une information approximative sur une distance est souhaitée. Les robots mobiles en utilisent très souvent pour capter leur environnement.

## LED et photo-transistor

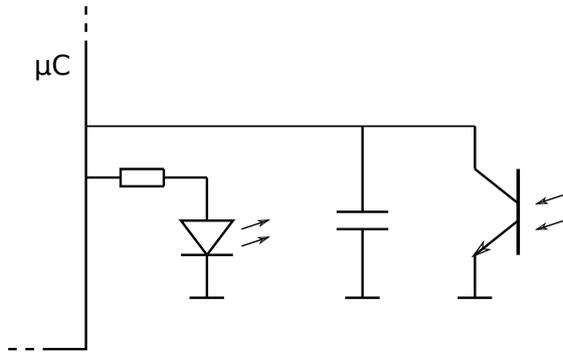
La LED est semblable aux autres LED... sauf que notre œil ne voit pas sa lumière ! Pour le dépannage, il est pratique d'utiliser un appareil photo numérique : même les modèles simples, qu'on trouve dans les téléphones portables, détectent très bien la lumière émise par cette LED. Il s'agit d'infrarouges, dont la longueur d'onde est supérieure à la lumière visible.

Le second élément est un photo-transistor. On se souvient qu'on peut décrire globalement le fonctionnement d'un transistor en disant que la résistance entre le collecteur et l'émetteur diminue lorsque le courant de base augmente. Pour un photo-transistor, la base n'existe pas (ou plus exactement, elle n'est pas connectée), mais c'est la lumière qui arrive

sur le transistor qui joue le rôle du courant de base. C'est la raison pour laquelle le photo-transistor est encapsulé dans une matière qui laisse bien passer les infrarouges.

### Schéma de mise en oeuvre

Voici un schéma pour la mise en oeuvre d'un tel capteur avec un microcontrôleur :



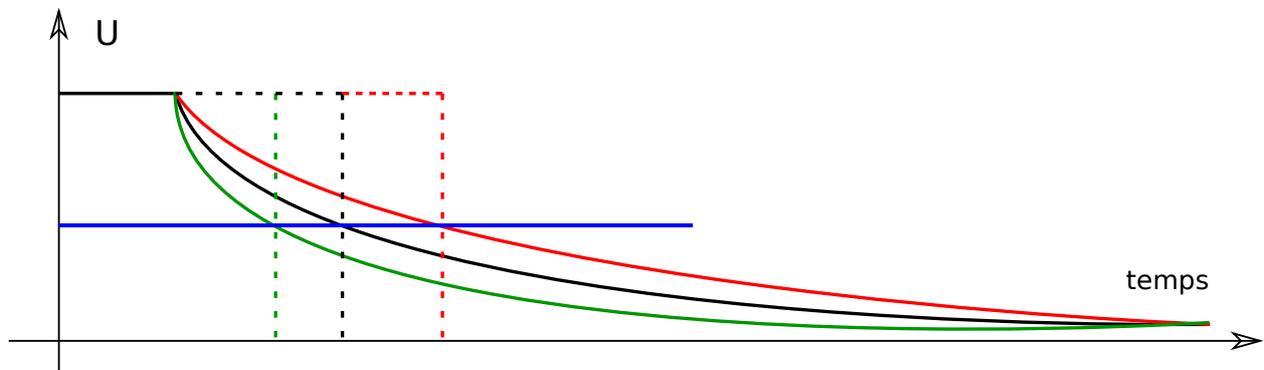
La LED est commandée par une patte du microcontrôleur, pour éviter de l'allumer en permanence et de consommer inutilement du courant. La résistance est calculée pour qu'un courant d'environ 10mA circule dans le LED, pour un éclairement maximum.

Pour le photo-transistor, plusieurs montages sont possibles. On pourrait réaliser un diviseur de tension avec une résistance fixe d'un côté

et le circuit collecteur - émetteur de l'autre, et mesurer ensuite la tension sur une entrée analogique.

Nous avons utilisé ici une autre technique : mettre l'émetteur à la masse, relier le collecteur à une patte du microcontrôleur et ajouter un condensateur entre la patte et la masse.

L'idée est la suivante : plutôt que de mesurer la résistance, on va charger le condensateur et le laisser se décharger dans la résistance. Or les lois de l'électricité montrent que le temps de décharge est proportionnel au produit  $R \cdot C$  (la résistance multipliée par la capacité). Comme  $C$  est fixe, le temps de décharge est donc proportionnel à la résistance.



### Mesure indirecte

On peut vraiment parler de mesure indirecte dans ce cas :

- La distance est convertie en intensité lumineuse (par l'ensemble LED – photo-transistor)
- l'intensité lumineuse est convertie en résistance électrique (par le photo-transistor)
- la résistance électrique est convertie en temps de décharge (par le schéma R-C)
- le microcontrôleur va finalement effectuer une mesure de ce temps de décharge.

De manière générale, cette technique de mesure de la résistance par mesure du temps de décharge d'un condensateur présente un intérêt : elle offre une dynamique très grande. Un microcontrôleur est capable de mesurer des temps allant de quelques dizaines de  $\mu s$  jusqu'à de seconde (ou même des minutes ou des heures). Dans notre cas, on a

surtout l'avantage d'un schéma très simple et qui évite des réglages : le programme sera facilement calibré en fonction de l'application.

## Le programme

Voici une procédure qui effectue une lecture de la distance :

```
#define CaptOut P1OUT
#define CaptDir P1DIR
#define CaptIn P1IN
#define CaptBit 1
#define CaptIrBit 0

int MesureDistance () {
    int distance = 0; // variable pour le résultat
    volatile int i; // pour les attentes actives
    int j; // variable pour la boucle de mesure
    CaptDir |= (1<<CaptBit); // patte en sortie
    CaptOut |= (1<<CaptBit); // charge du condensateur
    for (i=0; i<20; i++) {} // petite attente
    CaptDir &=~(1<<CaptBit); // patte en entrée

    CaptDir |= (1<<CaptIrBit); // IR en sortie
    CaptOut |= (1<<CaptIrBit); // IR allumé

    for (j=0; j<254; j++) { // boucle de mesure, 255 lectures
        for (i=0; i<4; i++) {} // petite attente
        if (CaptIn & (1<<CaptBit)) { // test de l'entrée
            distance++; // le condensateur est encore chargé
        }
    }
    CaptOut &=~(1<<CaptDirBit); // IR éteint
    return distance; // rend le résultat
}
```

La patte du microcontrôleur sur laquelle est connectée le photo-transistor est utilisée tantôt en entrée, tantôt en sortie. Les registres de direction, de sortie et d'entrée doivent donc tous être utilisés. La définition des ports utilisée a été mise dans des définitions de constantes `#define`, pour faciliter le changement du programme pour utiliser un autre port. Par contre, le choix a été fait de manipuler les ports directement dans la procédure.

Le programme commence par la charge du condensateur. La patte doit alors être une sortie, mise à l'état 1. Ensuite la patte devient une entrée. Durant un temps fixe, donné par les 255 itérations de la boucle `for` et le petit délai, le programme compte combien de fois il lit la valeur 1 à l'entrée (signe que le condensateur n'est pas encore suffisamment déchargé).

Le choix a été fait ici d'un nombre fixe de mesures, plutôt que d'attendre simplement que le condensateur se décharge. L'avantage réside dans le fait que la procédure dure un temps fixe, ce qui est souvent pratique pour l'intégration dans un système plus complexe.