

# LABORATOIRE D'INITIATION AUX SYSTEMES EMBARQUEES Q2

Cours destiné aux étudiants de première année en  
Technologie de l'Informatique et orientation  
sécurité des systèmes

---

## Table des matières

Laboratoire sur Arduino .....	3
Petits rappels : .....	3
Manip 1 : Allumer une Led via bouton poussoir .....	3
Manip 2 : Allumer une Led via bouton poussoir .....	4
Manip 3 : Télérupteur .....	4
Manip 4 : Dimmer .....	5
Manip 5 : Mesure analogique .....	5
Manip 6 : Clignotement Led .....	5
Manip 7 : Clignotement de 2 Leds .....	5
Récapitulatif : .....	6

## Laboratoire sur Arduino

### Petits rappels :

- Pour une Led verte :  $U_{led} = 2V$  et  $I_{led} = 20mA$  !
- Pour le calcul de la R, prendre la chute de tension aux bornes de la résistance et l'intensité du courant dans la LED (la résistance est en série sur la LED)

### Manip 1 : Allumer une Led via bouton poussoir

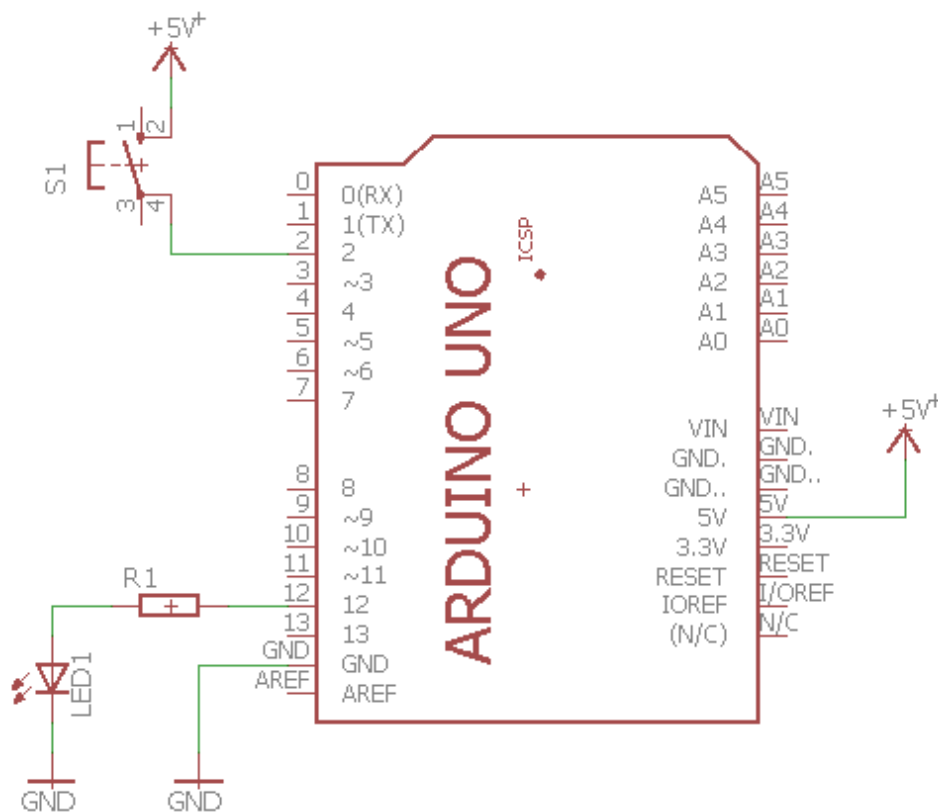
Réalisez un système embarqué qui allume une Led quand on garde l'appui sur un bouton poussoir. Dès qu'on lâche le bouton poussoir, la Led s'éteint.

Installez une surveillance de l'état de la Led par envoi de données sur le moniteur série. Sur ce moniteur série, sera affiché l'état de cette Led à chaque changement d'état de celle-ci.

Cahier des charges :

- La led doit être de couleur verte.
- La led doit être connectée sur la pin 12 de l'Arduino.
- Le bouton poussoir doit être connecté sur la pin 2 de l'Arduino.
- La lecture du bouton poussoir doit se faire via l'Arduino (entrée).
- La Led doit être commandée via l'Arduino (sortie).
- **Attention, il est strictement interdit d'utiliser une alimentation extérieure !! Prenez le 5V de l'Arduino pour alimenter vos composants !! L'alimentation sera assurée par la connexion USB.**
- Calculez la R1 à mettre en série avec la Led.
- Réalisez un code dans l'Arduino qui **lit** le bouton poussoir et qui **écrit** sur la Led en fonction de la lecture du bouton poussoir.
- Utilisez les fonctions : pinMode, digitalRead, digitalWrite.
- Soignez le code !
- Disposez les composants (Led, R1, BP) sur un breadboard et utilisez des câbles pour vous raccordez à l'Arduino.

Schéma :



Attention, le montage ne fonctionne pas !! (Cf. Cours de logique)

Pourquoi ?

Modifiez le montage pour rendre le système embarqué fonctionnel.

## Manip 2 : Allumer une Led via bouton poussoir

Faites la manipulation de la manip 1 en utilisant le mode "INPUT\_PULLUP" de la fonction "pinMode".

**Adaptez le schéma** pour piloter la Led sans avoir à inverser l'information du bouton poussoir. (La Led doit toujours s'allumer si on pousse sur le bouton poussoir.) La solution à trouver est donc Hardware !

## Manip 3 : Télérupteur

Faites fonctionner le bouton poussoir de la manip2 en télérupteur. C'est à dire que lorsque vous appuyez sur le bouton poussoir, la Led s'allume, et lorsque vous rappelez sur le bouton poussoir elle s'éteint et ainsi de suite.

#### **Manip 4 : Dimmer**

Modifiez le fonctionnement du bouton poussoir de la Manip3 pour faire fonctionner la led en dimmer. Chaque appui sur le bouton poussoir (BP) fera s'inverser le sens du réglage de la variation de luminosité de la LED (comme un dimmer piloté par BP).

Un appui sur le BP en continu amènera la led à un maximum ou un minimum en fonction du sens de la variation.

Que devez-vous modifier au schéma pour que cela fonctionne ? (PWM)

#### **Manip 5 : Mesure analogique**

Remplacez le fonctionnement du BP de la manip 4 par un potentiomètre (sur A0) pour régler la luminosité de la Led.

Envoyez l'information donnée par le potentiomètre sur le port série à chaque modification de celle-ci. (Voyez comment utiliser la commande "serial.println()").

#### **Manip 6 : Clignotement Led**

Faites clignoter une Led avec la fonction delay.

#### **Manip 7 : Clignotement de 2 Leds**

Idem que Manip 6 mais faites clignoter 2 Leds avec la librairie Tempo (qui se trouve sur Moodle) à des fréquences voisines.

Exemple : une période d'1 seconde pour la première Led et 1,1 seconde pour la seconde.

## Récapitulatif :

Au terme de ces séances, nous avons configuré des entrées/sorties de l'Arduino et les avons utilisées en entrée, sortie, sortie PWM, ou entrées analogiques.

Nous avons vu que l'**entrée** d'un composant ne peut être laissée "en l'air". Nous avons vu les deux solutions : nous pouvons mettre une résistance de pull-up ; nous pouvons aussi mettre une résistance de pull-down. Il est même possible de placer une résistance de pull-up sans connecter de composants extérieurs grâce au bénéfice du mode "**INPUT\_PULLUP**".

Nous avons utilisé une Led en la connectant de la **sortie** d'un Arduino à la **masse** ou au **Vcc**. L'intérêt de connecter une Led entre le Vcc et la sortie de l'Arduino permet d'alimenter une Led sans "pomper" sur l'alimentation interne du microcontrôleur.

Pour surveiller le fonctionnement du système, mais aussi, du programme lui-même, nous avons récupéré des données sur un port série. Vous avez donc **envoyé des données de l'Arduino au PC**. Cette opération peut se faire dans les deux sens, mais nous l'expérimenterons plus tard.

Nous avons aussi utilisé des éléments des bibliothèques de l'Arduino après une recherche.