# Séance 3 : Activité 4, le lampadaire automatique

**Sources :** Jean Christophe Quetin
https://arduiblog.com/2019/06/03/lampbit-de-kitronik/

**Thème :** Informatique embarquée et objets connectés
**Environnement envisagé :**

- PC
- Micro:bit
- LAMP:bit de Kitronik
- logiciel Mu de développement pour micro:bit.

**Prérequis :**

- Initiation au python

**Durée :**

- séance 30-40min

**Objectifs :**

- Découverte de LAMP:bit de Kitronik (1 actionneur et un capteur)
- Réaliser le programme d'un premier objet avec de l'informatique embarquée.

# Séance 3- Activité 4: Le lampadaire automatique

## 1. Présentation de l'étude

Le cahier des charges du lampadaire automatique :

Afin de limiter la consommation d'énergie des communes et réaliser des économies, on souhaite faire une étude de faisabilité d'un éclairage public de la voirie innovant. Celui-ci devra :

- Être inactif si personne n'est présent dans la rue.
- Si une personne est détectée, le lampadaire fonctionnera uniquement si l'éclairage naturel est insuffisant.

Pour s'assurer de la faisabilité, les éléments techniques employés seront :

- deux cartes micro:bit, l'une pour commander le lampadaire, l'autre pour simuler le téléphone portable d'un passant.

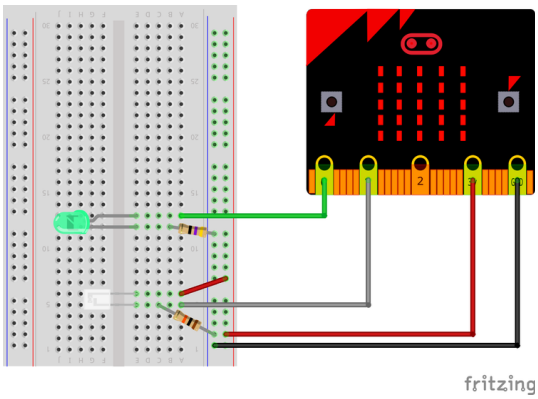
- Le LAMP:bit de Kitronik.


C'est un mini kit de lampadaire connectable solidement à la carte micro:bit à l'aide de 4 vis. Attention toutefois, lors des manipulations le pied du lampadaire est extrêmement instable et fragile.

La structure électrique est la suivante :

La lampe du lampadaire est branchée sur la borne zéro de la carte, elle sera donc utilisée en sortie numérique.

Un capteur de lumière (au sommet du lampadaire) constitué d'un photo transistor est alimenté entre le 3,3V et la masse. La sortie du montage est connectée à la broche 1 qui sera donc utilisée en entrée analogique.




La tension de la broche 0 de la micro:bit sera donc égale à 0 ou 3,3 V selon l'intensité lumineuse reçu par le photo transistor. Cette broche sera utilisée en sortie tout ou rien.

## 2. Les tests unitaires des capteurs et actionneurs

Avant de se lancer dans la réalisation complète du cahier des charges, il faut s'assurer que l'ensemble des capteurs et actionneurs communique bien avec le microcontrôleur. Il faut donc tester indépendamment ces éléments.

Pour faciliter la compréhension du programme et éventuellement le changement des broches de la carte, il est intéressant de donner un nom aux broches utilisées.

Exemples :

Si on souhaite renommer la broche 0, sur laquelle est connectée la lampe du lampadaire :
S_lampe = pin0 # Le S_ informe que la broche 0 est une sortie

Si on souhaite renommer la broche 1, sur laquelle est connectée le capteur de luminosité du lampadaire :
E_lum = pin1 #Le E_ informe que la broche 1 est une entrée

Sauvegardez bien vos tests unitaires. Ils peuvent lever un doute par la suite sur le bon fonctionnement du matériel.

### 2.1 Test Unitaire du lampadaire:

Essayez ce bout de programme, s'il est concluant, vous avez la main sur la lampe. Inutile d'aller plus loin si ce n'est pas le cas.

```
# ###################################################
# TU de la lampe
###################################################
#Import des bibliotheques#############################
from microbit import *

#Déclaration des sorties##########################
S_lampe = pin0

#Boucle infinie###################################
while True:
    S_lampe.write_digital(1) #Allume la lampe
    sleep(500)
    S_lampe.write_digital(0) #Eteint la lampe
    sleep(500)
```

### 2.2 Test Unitaire du capteur de lumière:

Dans cette partie, il faut s'approprier le fonctionnement du capteur. Celui-ci fournit une tension analogique comprise entre 0 et 3,3V que l'on connecte à un convertisseur analogique numérique de 12 bits, soit $2^{12}$ combinaisons. Le nombre issu de cette conversion est donc compris entre 0 et 4095.
Si la tension est de 0V alors le nombre converti sera 0.
Si la tension est de 3,3V alors le nombre converti sera 4095.

Afin d'observer le comportement du capteur en présence d'une lumière plus ou moins intense, nous allons renvoyer le résultat de la conversion analogique numérique dans la console micro python.

Essayez ce bout de programme, poursuivez s'il est concluant.

```
# ###################################################
# TU du capteur de luminosité
###################################################
#Import des bibliotheques#############################
from microbit import *

#Déclaration des Entrées##########################
E_lum = pin1

#Déclaration des sorties##########################
S_lampe = pin0

#Boucle infinie###################################
while True:
    lum = E_lum.read_analog()    # stocke la conversion dans la var lum
    print(lum)                   # renvoie dans la console lum
    sleep(500)
```

**Vers quelle valeur tend la conversion analogique numérique en l'absence de lumière ?**
**Vers quelle valeur tend la conversion analogique numérique en présence d'une lumière intense?**

2.3 Test Unitaire de la radio:

Ici il faudra deux cartes micro:bit, l'une sera l'émetteur du passant, l'autre le récepteur du lampadaire.

Pour activer la radio de la carte micro:bit : radio.on()
Pour désactiver la radio de la carte radio.off()

La carte du passant :

Pour envoyer un message : radio.send(''le message'')

Essayez ce bout de programme et vérifiez, dans la console, qu'il boucle bien.

```
# ###################################################
# L'emmetteur du passant
###################################################
# Import des bibliotheques#############################
from microbit import *
import radio

radio.on()

# Boucle infinie
while True:
    radio.send("passant")
    print("passant")
    sleep(100)
```

La carte récepteur du lampadaire :

Plusieurs méthodes existent pour la réception des messages sur une micro:bit. Nous n'en n'utiliserons qu'une seule, car celle-ci contient un octet indiquant la puissance du signal reçu.

radio.receive_full() est une méthode qui renvoie un tuple contenant trois valeurs :

- le prochain message entrant dans la file d'attente en octets.
- le RSSI (Received Signal Strength Information) (intensité du signal reçu): valeur comprise entre 0 (le plus fort) et -255 (le plus faible), mesurée en dBm.
- un horodatage en microsecondes: la valeur renvoyée par time.ticks_us() lors de la réception du message.

S'il n'y a pas de message en attente, alors ‘’**None**” est renvoyé.

Le RSSI nous permettra de régler la distance pour laquelle on souhaite allumer et éteindre le lampadaire.

Essayez ce bout de programme et notez les valeurs extrêmes du RSSI que vous obtiendrez ainsi que celle correspondant à la distance de détection souhaitée en approchant la carte passant vers le lampadaire.

```
# ######################################################
# TU recepteur lampadaire
######################################################
# Import des bibliotheques#############################
from microbit import *
import radio

radio.on()

# Boucle infinie
while True:
    Details_Reception = radio.receive_full()
    if Details_Reception:
        print(Details_Reception[0])
        print(Details_Reception[1])
        print(Details_Reception[2])
    print("coucou")
    sleep(100)
```

Voilà, à ce stade, si l'ensemble des tests unitaires sont positifs, c'est que la carte micro:bit communique bien avec l'ensemble des capteurs et actionneurs du sujet d'étude. La phase de réalisation du cahier des charges peut commencer.
Par la suite, au moindre doute sur le fonctionnement de l'un de ces éléments, vous pourrez relancer le test et vous assurer de son bon fonctionnement.

## 3. Mise en œuvre du cahier des charges

3.1 Le lampadaire (jour et nuit):

Concevez un programme qui déclenche l'allumage du lampadaire automatiquement lorsque l'intensité lumineuse devient trop faible. Si l'intensité augmente, le lampadaire s'éteint de nouveau. N'hésitez pas à renvoyer la valeur du capteur dans la console afin de vous assurer du fonctionnement cohérent de votre programmation. Quelle solution peut-on apporter pour éviter le clignotement ?

3.2 Le lampadaire fonctionne s'il y a un passant :

Concevez un programme qui déclenche l'allumage du lampadaire lorsqu'un ''passant'' s'approche de ce dernier de 30cm environ et qui s'éteint si le ''passant'' s'en éloigne.

3.3 Le lampadaire automatique:

Concevez un programme qui satisfait le cahier des charges.

Quel élément pourrait-on changer pour améliorer la sûreté ?