

**INTRODUCTION : CHOIX PÉDAGOGIQUES**

Le thème 5, projet expérimental et numérique, constitue la dernière partie du programme de l'enseignement scientifique de première générale. Cette partie n'est pas structurée comme les quatre précédentes autour de *savoirs* et de *savoir-faire*. Il s'agit de remobiliser les connaissances acquises en secondes (Ondes et signaux / Signaux et capteurs) afin de mettre en place un projet scientifique expérimental complet : utilisation voire réalisation d'un capteur, acquisition des données et exploitation. Le projet reste libre et peut s'inscrire en dehors du programme de l'année. L'objet du thème 5 tel que nous le présentons est de fournir les briques élémentaires permettant de comprendre ce qu'est un capteur (Partie A), éventuellement comment il est possible d'en réaliser un (Partie E), qu'est-ce que l'étalonnage indispensable d'un capteur (Partie B), comment facilement le mettre en œuvre avec un microcontrôleur comme Arduino™ (Partie C), et enfin comment exploiter les données (Partie D). Le choix d'Arduino™ comme microcontrôleur cible tient à plusieurs raisons : le prix modeste du microcontrôleur et sa robustesse, la facilité de sa mise en œuvre, la gratuité de l'environnement de développement, et enfin le nombre très important de références que l'on trouvera notamment sur la toile tant pour l'usage de capteurs que pour les bibliothèques logicielles associées.

**COMPLÉMENTS****Les capteurs du téléphone portable**

Sans mettre en œuvre une chaîne d'acquisition complète, il est aussi possible d'utiliser un *smartphone* récent afin d'utiliser un ou plusieurs de ses très nombreux capteurs (accéléromètre, gyroscope, capteur de lumière ambiante, microphone, etc.) pour des projets essentiellement de physique. S'il est possible de programmer sa propre application, on trouvera des outils génériques et gratuits comme le remarquable PhyPhox ([www.phyphox.org](http://www.phyphox.org)), développé à la RWTH Aachen University en Allemagne.



De nombreux projets sont également proposés sur leur site internet.

**Arduino : quelques références sur la toile.**



- <https://www.arduino.cc/> pour les références logicielles et le téléchargement de l'outil de développement. On y trouvera également les produits estampillés Arduino.

- <https://www.arduino-france.com/> pour une référence en français, qui comprend des tutoriels et des listes de projets.

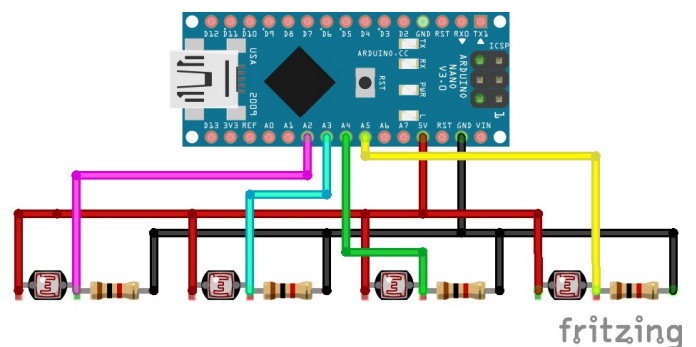
- <https://www.fun-mooc.fr/courses/MinesTelecom/04017/session01/about> qui est un des nombreux MOOC que l'on pourra trouver, celui-ci a l'avantage d'être dispensé en français.

**Idées de projets**

Se référer à la partie F p. 246 pour des idées de projets dans lesquels utiliser des capteurs.

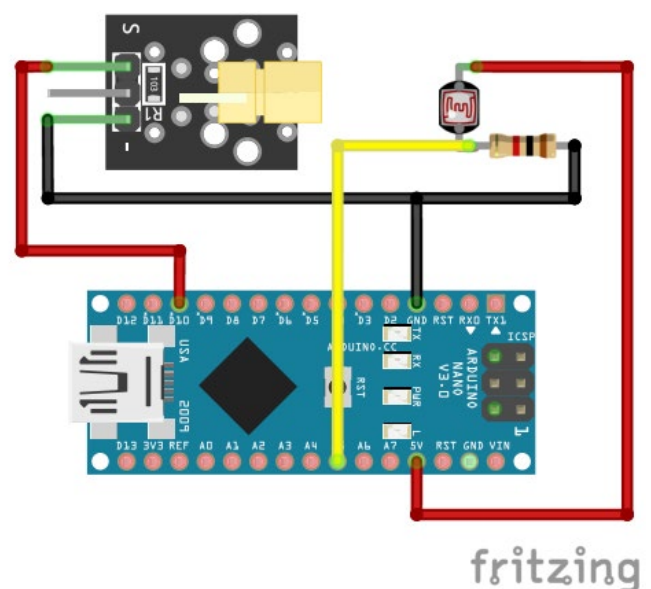
Exemple de Projet 4 p. 247 : Réaliser une carte de luminosité dans un jardin.

→ Plan de montage



Exemple de Projet 5 p. 247 : Réaliser une porte optique

→ Plan de montage



## Exemple de Projet 6 p. 247 : Réaliser un modèle de lampadaire automatique

→ Programme Arduino

```
lampadaire | Arduino 1.8.8
Echier Edition Croquis Outils Aide
lampadaire
1 int pinPhotoresistance = 5;
2 int valeurPhotoresistance;
3 int pinDEL = 10;
4 int seuil = 300; //a modifier en fonction du seuil souhaité
5
6 void setup() {
7   pinMode(pinDEL,OUTPUT);
8   digitalWrite(pinDEL,LOW);
9
10  Serial.begin(115200);
11}
12
13 void loop() {
14  valeurPhotoresistance = analogRead(pinPhotoresistance);
15  if (valeurPhotoresistance>seuil){
16    digitalWrite(pinDEL,HIGH);
17  }
18  else{
19    digitalWrite(pinDEL,LOW);
20  }
21
22  Serial.println(valeurPhotoresistance);
23}
Compilation terminée
Le croquis utilise 2090 octets (6% de l'espace de stockage de programmes. Le maximum est
Les variables globales utilisent 190 octets (9% de mémoire dynamique, ce qui laisse 1056
Arduino Nano, ATmega328P (016 Bootloader) sur IDEv1.8.8
```

→ Plan de montage

