

Système d'éclairage commandé par microcontrôleur

Alexis Dumont, LEGTA de Bourges

Contexte de la séance

Public visé :

Séance mise en œuvre en classe de seconde générale, en groupe.

Adaptable en première générale (partie de projet numérique) ou en STAV.

Durée :

1H30 de TP

Outils numériques utilisés :

PC avec interface de programmation Arduino, microcontrôleur.

Objectifs de la séance

Capteurs électriques.

Citer des exemples de capteurs présents dans les objets de la vie quotidienne.

Mesurer une grandeur physique à l'aide d'un capteur électrique résistif. Produire et utiliser une courbe d'étalonnage reliant la résistance d'un système avec une grandeur d'intérêt (température, pression, intensité lumineuse, etc.).

Utiliser un dispositif avec microcontrôleur et capteur.

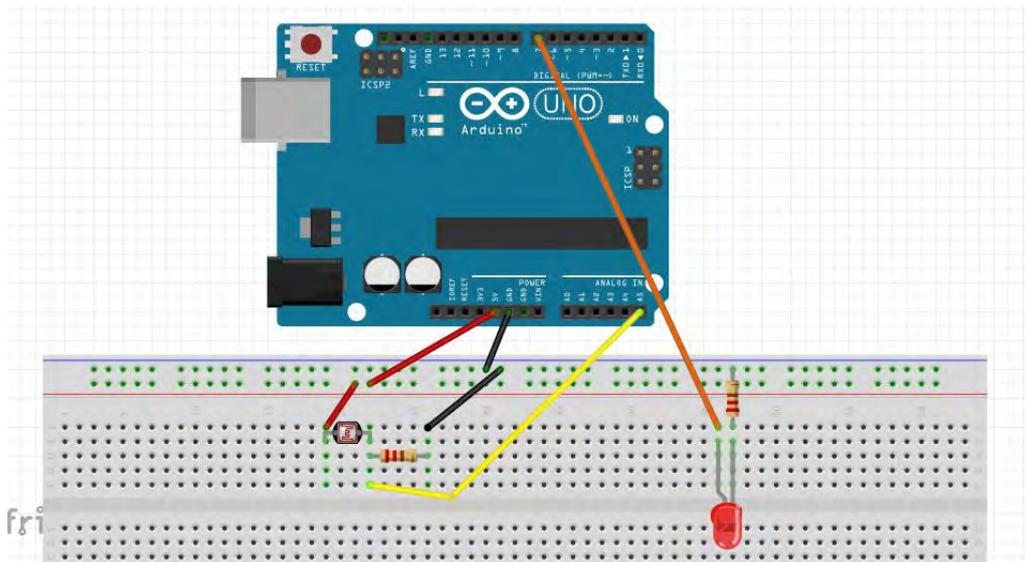
Cette séance est réalisée au terme de la partie « ondes et signaux ».

Pour bien comprendre le fonctionnement d'un capteur résistif, les élèves doivent avoir étudié la partie sur les conducteurs ohmiques.

Description de la séance

Activité : Système d'éclairage automatique

Câblage



Script

Ouvrir l'interface de programmation arduino de manière à faire apparaître le script :

```
//On déclare les variables lumin (pour récupérer la valeur de //luminosité, brochelumin - la broche sur laquelle est câblée le //capteur -  
// et led - la broche sur laquelle la led est branchée)  
  
int lumin;  
int brochelumin;  
int led = 7;  
  
void setup()  
{  
  // On démarre la connexion série :  
  Serial.begin(9600);  
  // On définit en entrée la broche du capteur :  
  pinMode(brochelumin, INPUT);  
  // On définit en sortie la broche de la LED :  
  pinMode(led, OUTPUT);  
  // On éteint la LED :  
  digitalWrite(led, LOW);  
}  
  
void loop()  
{  
  // On relève et on affiche toutes les secondes :  
  lumin = analogRead(brochelumin);  
  // On écrit la valeur sur le port série :  
  Serial.println(lumin);  
  delay(1000);  
  // Si la luminosité est trop faible, on allume la LED :  
  if (lumin >= 770)  
  {  
    digitalWrite(led, HIGH);  
  }  
  else  
  {  
    digitalWrite(led, LOW); // Sinon on ne fait rien :  
  }  
}
```

- 1) Décrire l'intérêt de ce montage par rapport à un éclairage classique au bouton.
- 2) Décrire le principe de fonctionnement du dispositif

Modification du montage

Rajouter une DEL verte sur la broche 6 et modifier le script pour que quand la luminosité est inférieure à 770, la led verte s'allume.

Analyse de la séance

Les avantages

L'activité concrète suscite l'intérêt des élèves.

Le script reste relativement simple, ce qui permet une bonne initiation aux microcontrôleurs.

Les inconvénients

La maîtrise de l'interface de programmation est nécessaire, notamment pour la configuration du port sur lequel la carte est branchée.

Une maîtrise du script reste nécessaire également, notamment au moment où les élèves vont coder : avec la multiplicité des groupes et des scripts répondant correctement à la demande, il est important de repérer rapidement les erreurs de codage.

```
void loop()
{
  // On relève et on affiche toutes les secondes.
  lumin = analogRead(brochelumin);
  // On écrit la valeur sur le port série
  Serial.println (lumin);
  delay (1000);
  //Si la luminosité est trop faible, on allume la LED
  if (lumin >= 770)
  {
    digitalWrite(led, HIGH);
    digitalWrite(ledverte, LOW);
  }
  else
  {
    digitalWrite(led, LOW); //Sinon on ne fait rien
    digitalWrite(ledverte, HIGH);
  }
}
```

```
void loop()
{
  // On relève et on affiche toutes les secondes.
  lumin = analogRead(brochelumin);
  // On écrit la valeur sur le port série
  Serial.println (lumin);
  delay (1000);
  //Si la luminosité est trop faible, on allume la LED
  if (lumin >= 770)
  {
    digitalWrite(led, HIGH);
  }
  else
  {
    digitalWrite(led, LOW); //Sinon on ne fait rien
  }
  if (lumin <= 770)
  {
    digitalWrite(ledverte, HIGH);
  }
  else
  {
    digitalWrite(led, LOW); //Sinon on ne fait rien
  }
}
```

2 exemples de programmation correcte

Les éventuelles améliorations qui peuvent être proposées

La principale amélioration sur laquelle on peut travailler est l'introduction d'une courbe d'étalonnage entre la valeur de la résistance (convertie numériquement en valeur numérique allant de 0 à 1023) et la valeur de la luminosité.

Cela permettrait de répondre à un objectif supplémentaire du BO en poussant les élèves à trouver cette valeur admise de 770 citée dans le script.

Le tracé de la courbe reste difficile à mener (fluctuation des valeurs renvoyées par arduino, fluctuation des valeurs de luminosité).

Bilan par rapport aux élèves

Les microcontrôleurs ont été abordés durant la scolarité des élèves au collège en technologie et ils semblent à l'aise pour en identifier l'utilité. Ils identifient donc clairement la plus-value de l'outil de programmation.

Les élèves semblent également bien comprendre l'intérêt d'étudier préalablement à cette activité les conducteurs ohmiques et la grandeur « résistance » associée puisqu'elle permet la compréhension du fonctionnement d'un capteur résistif.

Les difficultés des élèves restent dans l'utilisation de la syntaxe liée à la programmation (oubli d'un point-virgule, d'une accolade, ...)

Une solution pourrait être la participation de l'enseignant de sciences numériques et technologie (SNT) à cette activité, les microcontrôleurs étant abordés dans cette discipline.

La séance entraîne des questions intéressantes et professionnelles des élèves (peut-on avoir un capteur qui permet le remplissage automatique de pots de miel avec arduino ?)

Bilan par rapport à l'enseignant

Une séance sportive comme un TP mais avec des choses différentes à appréhender, et plus particulièrement le repérage des fautes syntaxiques des élèves dans les scripts. Plusieurs corrections sont possibles pour l'activité (voir les exemples de programmation correcte ci-dessus).

Une séance qui donne encore plus de sens à la discipline en mettant en œuvre simplement une application simple.