

	Sciences et Technologies de l'Industrie et du Développement Durable		
	INGÉNIERIE ET DÉVELOPPEMENT DURABLE		
	Partiel TP CAN + Relais	TP	I2D

## 1. Etude la photorésistance

1. **Mesures** : Mesurer la valeur de la résistance de la photorésistance pour 5 éclairagements différents.
2. **Traitement des données** : Sur Excel, rassembler ces mesures dans un tableau, puis tracer le graphique sous forme de nuage de points. Faire apparaître l'équation de la courbe de tendance sur le graphique.
3. **Analyse** : Déduire la valeur de la résistance (en  $\Omega$ ) dans des conditions d'obscurité totale (nuit).
4. **Validation** : 🙋 Appeler le professeur pour faire vérifier votre travail.

## 2. Conception du schéma électrique

1. Ouvrir le logiciel **QElectroTech**.
2. Dessiner le schéma de câblage d'un pont diviseur de tension intégrant la photorésistance et une résistance fixe.
3. Attribuer à la résistance fixe la même valeur (en  $\Omega$ ) que celle déterminée pour la nuit lors de la manipulation précédente.
4. Relier la sortie de ce pont diviseur à la broche analogique **A0** de la carte Arduino (sans oublier les alimentations 5V et GND).
5. 🙋 Appeler le professeur pour validation du schéma.

## 3. Câblage

1. Sur une platine d'essai, réaliser le circuit en suivant le schéma validé à l'étape précédente.
2. Utiliser pour cela votre photorésistance (LDR) et une résistance de **10 k $\Omega$** .
3. Effectuer les raccordements vers la carte Arduino (GND, 5V et entrée analogique A0).
4. **● NE PAS METTRE SOUS TENSION !** > 5. 🙋 Appeler impérativement le professeur pour validation du montage matériel avant de brancher le câble USB.

## 4. Calcul de la valeur numérique théorique

---

1. En utilisant la formule du pont diviseur de tension, calculer la tension théorique sur la broche A0 lorsqu'il fait nuit. (Rappel : utiliser la valeur de la résistance fixe de 10 k $\Omega$  et la valeur en  $\Omega$  de la LDR pour la nuit).
2. Le convertisseur analogique-numérique (CAN) de l'Arduino de 10 bits transforme une tension de 0 à 5V en un nombre entier compris entre 0 et 1023. À partir de la tension calculée, déduire la valeur numérique qui devra s'afficher sur le moniteur série.
3. 🖐️ Faire valider vos calculs par le professeur.

## 5. Configuration et Programmation

---

1. Ouvrir le logiciel Arduino IDE.
2. Dans le menu Outils, vérifier que le type de carte (ex: Arduino Uno) et le port (Port COM) sont bien sélectionnés.
3. Saisir le code suivant dans l'éditeur :

```
// --- DÉCLARATION DES CONSTANTES ---
// Définition de la broche analogique utilisée pour la LDR
#define BROCHE_LDR A0

// Définition du temps de pause entre chaque mesure (en millisecondes)
#define TEMPS_PAUSE 500

void setup() {
  // Initialisation de la communication avec l'ordinateur à 9600 bauds
  Serial.begin(9600);
}

void loop() {
  // --- LECTURE ---
  // Lecture de la tension convertie sur la broche A0 (donne un nombre de 0 à 1023)
  // On utilise "int" car cette valeur change à chaque tour de boucle
  int valeurLDR = analogRead(BROCHE_LDR);

  // --- AFFICHAGE ---
  // Envoi de la valeur lue sur le moniteur série de l'ordinateur
  Serial.println(valeurLDR);

  // Pause avant de recommencer la prochaine mesure
  delay(TEMPS_PAUSE);
}
```

4. Cliquer sur le bouton **\*\*Téléverser\*\*** (la flèche vers la droite) pour envoyer le programme dans la carte.
5. Ouvrir le Moniteur série en cliquant sur l'icône en forme de loupe en haut à droite de l'écran.
6. Attention : Vérifiez que le menu déroulant en bas à droite du moniteur série indique bien "9600 baud".
7. Plonger la photorésistance dans l'obscurité totale (en la recouvrant avec vos mains ou un capuchon opaque).
8. Relever la valeur numérique qui s'affiche à l'écran : Valeur mesurée = .....
9. 🙋 Faire valider vos calculs par le professeur.

## 6. Ajout d'un éclairage (Schéma) + Câblage

---

1. Reprendre votre schéma précédent sur le logiciel QElectroTech.
2. Ajouter un relais et relier sa commande à la broche numérique 10 de la carte Arduino.
3. Relier une LED 12V au circuit de puissance du relais (en NO) et une source d'alimentation 12V indépendante (en COM).
4. Réaliser hors tension le câblage
5. 🙋 Appeler le professeur pour validation du schéma et du câblage.

## 7. Programmation

---

1. Retournez sur le logiciel Arduino IDE.
2. Vous allez maintenant modifier votre programme pour que la carte Arduino prenne une décision : allumer la LED 12V uniquement si la luminosité correspond à la nuit.
3. Remplacez votre ancien code par celui ci-dessous.
4. ⚠ Attention : À la ligne `#define SEUIL_NUIT ...`, remplacez les pointillés par la valeur numérique de la nuit que vous avez validée à l'étape précédente.
5. Téléversez le programme, puis testez votre système en cachant la photorésistance.
6. 🙋 Appelez le professeur pour la validation finale du fonctionnement de votre système automatisé.

```
// --- DÉCLARATION DES CONSTANTES ---
#define BROCHE_LDR A0    // Broche de lecture de la luminosité
#define BROCHE_RELAIS 10 // Broche de commande du relais
#define TEMPS_PAUSE 500 // Temps de pause en millisecondes
// Indiquez ci-dessous la valeur de basculement pour la nuit
#define SEUIL_NUIT .....

void setup() {
  Serial.begin(9600);

  // Configuration de la broche du relais en tant que SORTIE (actionneur)
  pinMode(BROCHE_RELAIS, OUTPUT);
}

void loop() {
  // --- LECTURE ---
  int valeurLDR = analogRead(BROCHE_LDR);
  Serial.println(valeurLDR);

  // --- COMMANDE DE L'ÉCLAIRAGE ---
  // Si la valeur lue correspond à l'obscurité (nuit)
  if (valeurLDR < SEUIL_NUIT) {
    // On active le relais pour allumer la LED 12V
    digitalWrite(BROCHE_RELAIS, HIGH);
  }
  // Sinon (il fait jour)
  else {
    // On désactive le relais pour éteindre la LED 12V
    digitalWrite(BROCHE_RELAIS, LOW);
  }

  delay(TEMPS_PAUSE);
}
```