

Caractéristique d'un conducteur ohmique

La photorésistance

1- Documents

Document 1 - Les capteurs

Un capteur est un circuit électronique ou un simple composant électronique (éventuellement une simple résistance). Il permet de faire le lien entre une grandeur physique que l'on souhaite mesurer (température, lumière, humidité, intensité...) et une grandeur électrique. Chaque capteur possède ses spécificités, ses propres caractéristiques: la courbe d'étalonnage d'un capteur sert à établir une relation mathématique entre la tension mesurée à ses bornes et la grandeur physique que l'on souhaite étudier de manière à modéliser l'évolution de la tension en fonction des valeurs prises par le capteur.

Document 2 - Caractéristique tension intensité

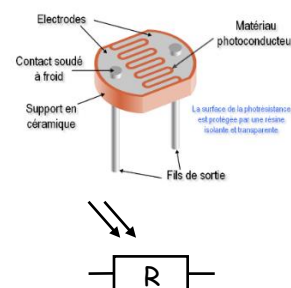
Afin de définir les propriétés électriques d'un dipôle, il est nécessaire d'étudier sa caractéristique tension-intensité $U=f(I)$.

Cette caractéristique est la courbe représentant la tension U entre les bornes du dipôle étudié en fonction de l'intensité I du courant qui le traverse.

Document 3 - La photorésistance LDR

LDR pour Light Dependent Resistor, la photorésistance est un dipôle dont la résistance varie en fonction de l'éclairement E qu'elle reçoit d'une source de lumière: plus elle est éclairée, plus la valeur de sa résistance baisse.

La partie sensible du capteur est une piste de sulfure de cadmium en forme de serpent: l'énergie lumineuse reçue déclenche une augmentation de porteurs de charges libres dans ce matériau, de sorte que sa résistance électrique évolue.



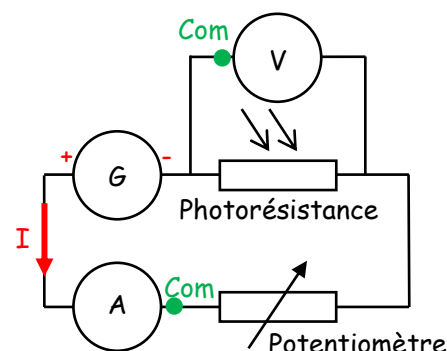
Document 4 - Montage expérimental de base Caractéristique de la photorésistance

La caractéristique d'un dipôle est la représentation graphique de la tension U entre ses bornes, en fonction de l'intensité I du courant électrique qui le traverse.

Pour étudier la caractéristique de la photorésistance, on utilisera le montage schématisé ci-contre.

Le voltmètre est placé aux bornes de la photorésistance.

Attention: la lumière arrivant sur la photorésistance doit rester constante tout au long des mesures. On ne fait varier que la tension d'alimentation.

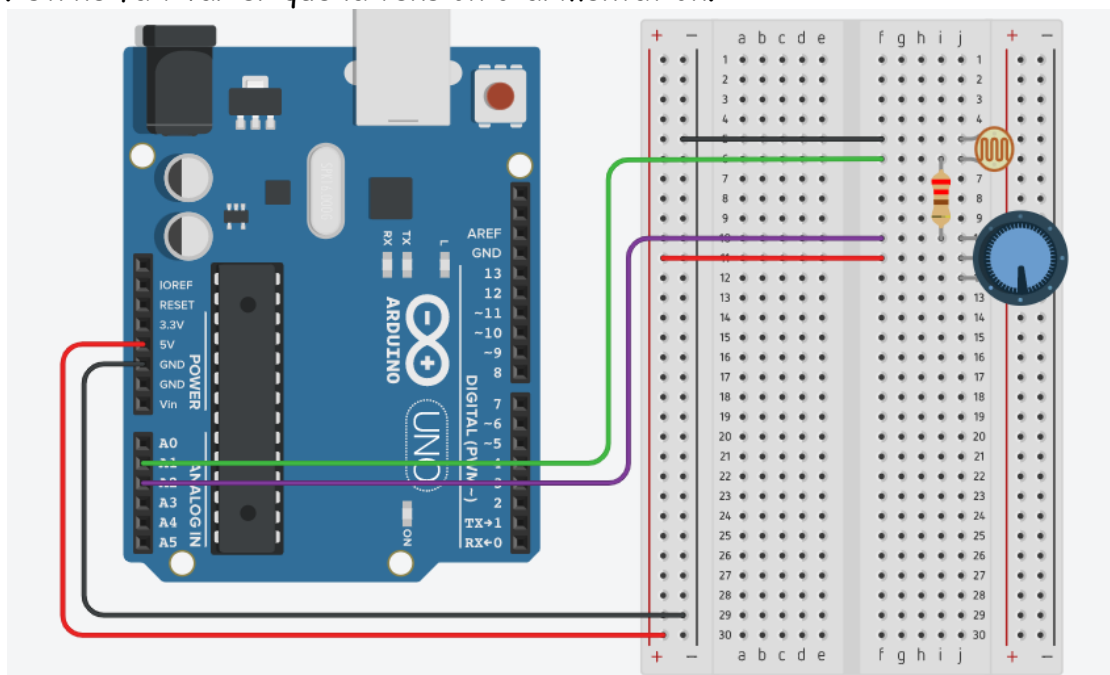


Document 5 - Montage expérimental avec Arduino Caractéristique de la photorésistance

Pour étudier la caractéristique de la photorésistance avec une interface Arduino, on utilisera le montage schématisé ci-dessous.

Le voltmètre est placé aux bornes de la photorésistance.

Attention: la lumière arrivant sur la photorésistance doit rester constante tout au long des mesures. On ne fait varier que la tension d'alimentation.



Document 6 - Mesure de tension et de courant avec Arduino

Afin de mesurer la valeur de la tension aux bornes de la photorésistance on utilise le port A2 de la carte Arduino. Ce port correspond au potentiel U2

Pour mesurer la valeur de l'intensité de courant traversant la photorésistance on utilise la loi d'Ohm. On mesure pour cela la tension aux bornes de la résistance de $R = 220 \Omega$. On utilise le port A1 de la carte Arduino. Ce port correspond au potentiel U1. La différence de potentiel U1-U2 donnera la tension aux bornes de la résistance. L'intensité de courant I traversant la résistance R aura pour valeur $I = (U1-U2)/R$.

Le potentiomètre permet de faire varier la tension aux bornes du dispositif Résistance - Potorésistance.

Document 7 - Matériel pour la caractéristique - Avec interface Arduino

- Ordinateur avec Spyder et Arduino
- Plaque microcontrôleur Arduino avec cordon USB
- Dispositif Ardui-TC-PL de Potorésistance - Luxmètre
- Potorésistance de $1 \text{ k}\Omega$ ou $10 \text{ k}\Omega$
- Résistance de 220Ω
- Potentiomètres de $1 \text{ k}\Omega$ ou $10 \text{ k}\Omega$
- Fils de connexion

2- Python

Python est un langage de programmation interprété (les instructions que vous lui envoyez sont "transcrites" en langage machine au fur et à mesure de leur lecture), à ne pas confondre avec un langage compilé, (où avant de pouvoir les exécuter, le logiciel se charge de transformer le code du programme en langage machine).

Il est possible d'associer des bibliothèques à Python afin d'étendre ses possibilités.

Document 8 - Programme en Python pour la caractéristique de la photorésistance

Le programme à utiliser se nomme: "Caracteristique photoresistance.py"

<pre># Prof-TC # Mai 2121 # Caractéristique d'une photorésistance # Tracé de U=f(I) pour une photorésistance #Affichage du nom du programme print("_____") print("Caractéristique d'une photorésistance") print("_____") #Importation des bibliothèques import matplotlib.pyplot as plt import numpy as np #Valeurs expérimentales à modifier selon l'expérience (U en volt et I en milliampère) print("*****") ImA=[0, 2.42, 2.49, 2.51, 2.64, 2.89, 3.04, 3.18, 3.24, 3.35, 3.44, 3.58, 3.73, 3.84, 4.00, 4.20, 4.95] U = [0, 2.63, 2.55, 2.56, 2.71, 2.94, 3.08, 3.23, 3.31, 3.40, 3.50, 3.62, 3.77, 3.90, 4.05, 4.25, 4.99] print("*****") # Nombre de points expérimentaux K=len(ImA) # Détermination des valeurs de I en ampère I=[] for i in range(K): I.append(round(ImA[i]/1000,6)) #Affichage des valeurs expérimentales et du nombre de points print("valeurs de I: ",I) print("valeurs de U: ",U) print("Il y a ",K," couples de points") print("*****") #Initialisation des variables Imoy=0.0 U moy=0.0 N=0.0 D=0.0 a=0.0 b=0.0 Imin=I[0] Umin=U[0] Imax=I[len(I)-1] Umax=U[len(U)-1] # Initialisation de la figure fig, ax = plt.subplots() #Calcul des coefficients de la droite représentant U=f(I) for i in range (len(I)): Imoy=Imoy+I[i] U moy=U moy+U[i]</pre>	<pre>Imoy=Imoy/(len(I)) U moy=U moy/(len(U)) for i in range (len(I)): N=N+(I[i]-Imoy)*(U[i]-U moy) D=D+(I[i]-Imoy)*(I[i]-Imoy) a=N/D #Coefficient directeur de la droite R=a #Valeur de la résistance b=U moy-a*Imoy #Ordonnée à l'origine de la droite #Mise en écriture avec 5 décimales pour l'affichage de l'équation Rs=round(R,5) bs=round (b,5) #Mise en écriture pour l'affichage de la résistance Rp=int(R+0.5) #Tracé des axes, labels et titre du graphique plt.axis([0, Imax, 0, Umax]) plt.xlabel('I (mA)',color='green', fontsize=20) plt.ylabel('U (V)',color='green', fontsize=20) plt.title("Caractéristique d'une photorésistance", color='red', fontsize=10) plt.grid() #Tracé des valeurs expérimentales plt.scatter(I,U,marker='o',color='r',linewidth = 4) #Tracé de la droite x=np.linspace(0,Imax+1,200) y=R*x+b plt.plot(x,y,linewidth = 2) #Ecriture de l'équation de la droite plt.text((Imax)/2, Umax-Umax*0.1, "U = "+str(Rs)+" x I + "+str(bs), color='green', fontsize=12,horizontalalignment = 'center') #Ecriture de la valeur de la résistance plt.text((Imax)/2, Umax-Umax*0.2, "R = "+str(Rp)+" Ohm", color='green', fontsize=12,horizontalalignment = 'center') #Affichage du graphique plt.show() # Sauvegarde de la figure dans le dossier où se trouve le programme fig.savefig("Caractéristique d'une photorésistance") print("*****") print("L'équation de la droite est: U = ",Rs," x I + ",bs) print("*****") #Fin du programme</pre>
--	--

3- Arduino

L'Arduino est une plateforme de prototypage électronique open-source, basée d'une part sur du matériel et d'autre part sur un ensemble de logiciels faciles à utiliser.

Une carte Arduino communique avec son environnement par l'intermédiaire de ses broches d'entrées/sorties. Sur ces broches, des capteurs, dispositifs permettant de transformer une information de l'environnement en signal électrique et des actionneurs, dispositifs permettant de transformer un signal électrique en action mécanique ou lumineuse, vont être connectés.

L'Arduino étant un ordinateur spécialisé dans la gestion de capteurs et d'actionneurs, c'est un programme qui va décider de la manière donc les capteurs et les actionneurs sont utilisés. C'est donc très différent de l'électronique traditionnelle où les fonctions qui relient les capteurs aux actionneurs, l'appui d'un bouton qui entraîne l'allumage d'un DEL par exemple, sont déterminées en dur par le câblage entre les composants, et par les composants eux-mêmes.

Ici les fonctions sont déterminées par un programme. Par conséquent les fonctions peuvent être, d'une part, beaucoup plus élaborées et, d'autre part, modifiables à volonté sans changer la quincaillerie.

Le langage de programmation Arduino peut être divisé en trois parties principales: structures, valeurs (variables et constantes), et fonctions.

Le langage Arduino est très proche du langage C++. Les règles de syntaxe sont communes.

Document 9- Programme Arduino pour l'acquisition des données Le programme à utiliser se nomme: "Caracteristiquephotoresistance.ino"	
<pre> * *Caracteristique photoresistance *18 mars 2022 *Prof-TC */ //----- // Mesure de la résistance d'une Photoresistance //----- #define R 220 // Résistance du pont float U1; // Potentiel a la borne de la resistance R float U2; // Potentiel a la borne de la photoresistance float I; // Courant dans le circuit //----- void setup() { Serial.begin(9600); // Paramétrage du port série Serial.println("Caracteristique d'une photoresistance"); //Titre du programme Serial.println(""); // saute une ligne Serial.print("I (mA) "); // donne la valeur du courant en milliampere Serial.println("U (V)"); // donne la valeur de la tension en Volt Serial.println(""); // saute une ligne } //----- void loop() { U1 = analogRead(A1)*5.0/1023; // Lecture potentiel A1 U2 = analogRead(A2)*5.0/1023; // Lecture potentiel A2 I=1000*(U1-U2)/R; // Calcul courant en milliampere Serial.print(I); // Affichage du courant I en milliampere traversant la photoresistance Serial.print(" "); // Espacement Serial.println(U1); // Affichage de la tension U en Volt aux bornes de la photoresistance delay(5000); // Temporisation de 5s } //----- </pre>	

Document 10- Programme Arduino pour allumer une DEL

Le programme à utiliser se nomme: "PhotoresistanceLED.ino"

```

*
*Photorésistance et LED
*30 mai 2021
*Prof-TC
*/

const char led = 2;           // Cette LED indiquera s'il fait jour
const char capteur = 0;      // Sur la broche A0 est connecté le pont diviseur de tension

float tension = 0;          // "tension" est la variable qui enregistre la tension lue en sortie du capteur
float seuil = 2.5;          // "seuil" est la valeur en V qui détermine le niveau auquel l'obscurité est présente

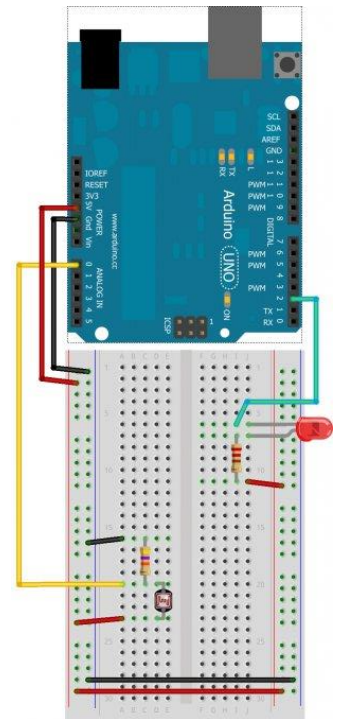
void setup()
{
  pinMode(led, OUTPUT);      // définition des broches utilisées
  Serial.begin(9600);        // la voie série pour monitorer
}

void loop()
{
  tension = (analogRead(capteur) * 5.0) / 1024; // conversion de cette valeur en tension

  if(tension <= seuil)
  {
    digitalWrite(led, LOW); // On allume la LED
  }
  else
  {
    digitalWrite(led, HIGH); // On éteint la LED
  }
  Serial.print("Tension = "); // envoi de la valeur de la tension lue vers l'ordinateur via la liaison série
  Serial.print(tension);
  Serial.println(" V");

  delay(500); // délai pour ne prendre des mesures que toutes les demi-secondes
}

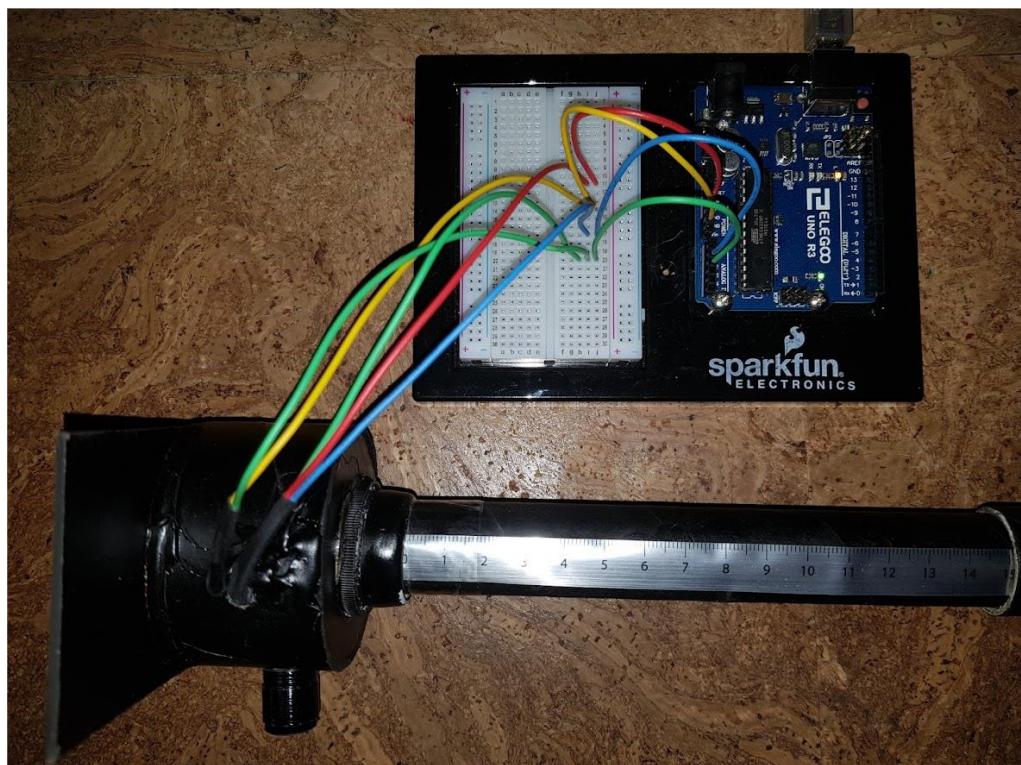
```



4- Expériences

4.1- Caractéristique de la photorésistance

Faire le montage du document 5 afin d'étudier la caractéristique de la photorésistance.



Pour les connexions des fils sur la carte Arduino on devra respecter les codes de couleur suivants:

- Le fil rouge sur l'alimentation "5V".
- Le fil jaune sur la masse "GND".
- Le fil bleu au port "A1".
- Les fils verts au port "A2".

Attention à bien maintenir le même éclairage sur la photorésistance pour toute la durée de l'expérience.

Faire vérifier le montage par le professeur avant de réaliser les mesures.

Utiliser le programme "Caracteristiquephotoresistance.ino".

Allumer la lampe uniquement pendant la durée des mesures.

Relever les différentes valeurs de la tension et du courant en utilisant le potentiomètre et compléter le tableau suivant.

I (mA)													
U (V)													

Utiliser le programme "Caracteristique photoresistance.py" afin de tracer la caractéristique de la photorésistance.

Relever l'équation de la caractéristique et la valeur R de la résistance de la photorésistance.

4.2- Photorésistance et DEL

A l'aide de l'Ohmmètre mesure la valeur de la résistance de la photorésistance quand celle ci est dans l'obscurité. Faire de même lorsque qu'elle est éclairée.

Comment varie la valeur de la résistance de la photorésistance en fonction de l'éclairage?

Faire le montage Arduino du document 10 afin de créer un système d'allumage automatique en fonction de la luminosité.

Utiliser le programme "PhotoresistanceLED.ino".

Tester ce montage en faisant varier la luminosité et en prenant des valeurs différentes pour la tension seuil.

