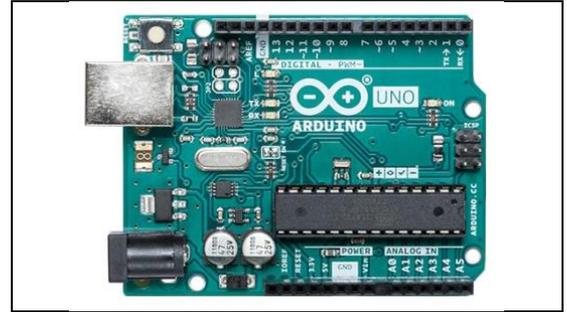


Noms : _____
Prénoms : _____
Classe : _____
Date : _____

Note : /20



1. Compétences abordées :

- Utiliser les lois électriques pour résoudre un problème
- Utiliser des appareils de mesure en toute sécurité
- Comprendre un programme informatique
- Comparer, traiter, organiser et synthétiser les informations pertinentes

2. Critères d'évaluation et barème

Loi d'Ohm (Q1, Q2, Q3, Q4, Q5, Q6, Q7)	/6
Associations de résistances (Q8)	/3
Pont diviseur de tension (Q9, Q10, Q11)	/3
Montage 1 diode (Q12)	/2
Montage 3 diodes (Q13)	/2
Montage bouton poussoir (Q14)	/2
Prototype (accéléromètre, température, distance) (Q15, Q16, Q17, Q18)	/2

3. Documents à utiliser

Sur le site web : <http://lycee-ferry-versailles.fr/spe-si/>

Chapitre 3 / signaux électriques / Lois électriques

Chapitre 5 / Arduino / Cours

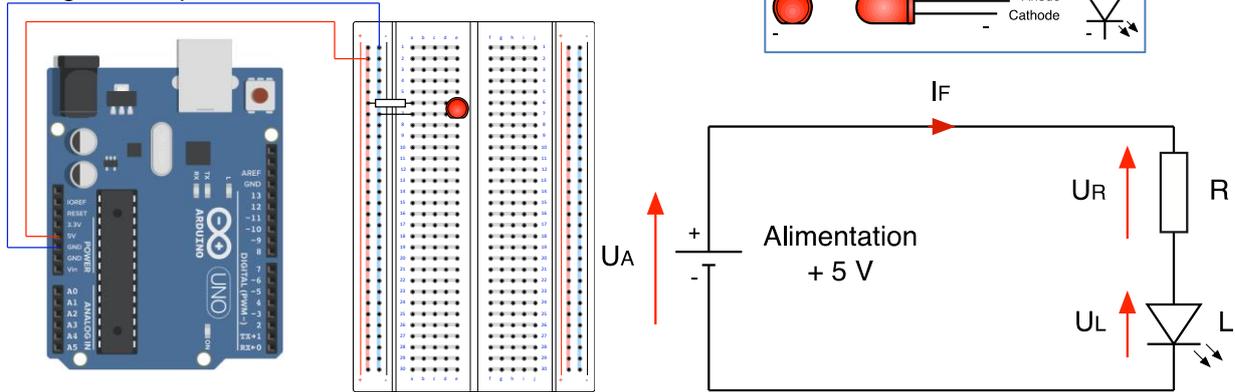
4. Règles de sécurité



**TOUS LES MONTAGES DOIVENT ÊTRE RÉALISÉS HORS TENSION ET
VÉRIFIÉS PAR LE PROFESSEUR AVANT LA MISE SOUS TENSION**

5. Loi d'Ohm

Soit le montage électrique suivant :



Utiliser les bornes « 5V » et « GND » de la carte Arduino



TOUS LES MONTAGES DOIVENT ÊTRE RÉALISÉS HORS TENSION (câble USB débranché de la carte ou de l'ordinateur) ET VÉRIFIÉS PAR LE PROFESSEUR AVANT LA MISE SOUS TENSION

Q1 : Déterminer l'expression de la tension U_R en fonction des tensions U_A et U_L .

Q2 : Calculer la tension U_R sachant que la tension U_L est de 2,1 V lorsque la Led est alimentée.

Q3 : Indiquer quel appareil de mesure et comment vous devez le brancher pour mesurer la tension U_R aux bornes de la résistance. L'ajouter sur le schéma électrique.

Q4 : Déterminer la valeur de la résistance R pour un courant I_F de 15 mA.

Q5 : Parmi les résistances mises à votre disposition, choisir celle dont la valeur est immédiatement supérieure

Q6 : Expliquer à quoi signifient, couleur par couleur, les anneaux colorés peints sur la résistance.

Q7 : Utiliser un ohmmètre pour vérifier la valeur de la résistance : faire le schéma du branchement ***Préciser si le montage doit être sous tension ou non***, noter la valeur lue, exprimer l'écart relatif.

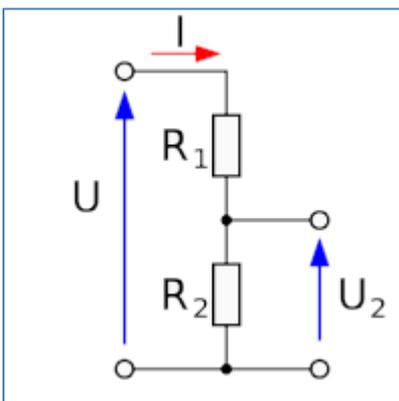
6. Association de résistances

Q7 : Défi : Vous disposez de résistances de 100, 220, 510 et 1 000 Ω

Comment devez-vous les brancher (faire un schéma) pour obtenir une résistance équivalente de :

- 1 730 Ω
- 50 Ω
- 475 Ω

7. Pont diviseur de tension



Le diviseur de tension est un montage très classique. Il permet, à partir d'une tension de référence, par exemple 5V, de construire une tension moins élevée, par exemple 3,3V. Il trouvera naturellement sa place lorsqu'il s'agit de connecter une carte Arduino délivrant du 5V à un module ou un composant fonctionnant en 3,3V.

Sur le schéma suivant, la tension d'entrée U est divisée selon R_1 et R_2 et donne la tension de sortie U_2

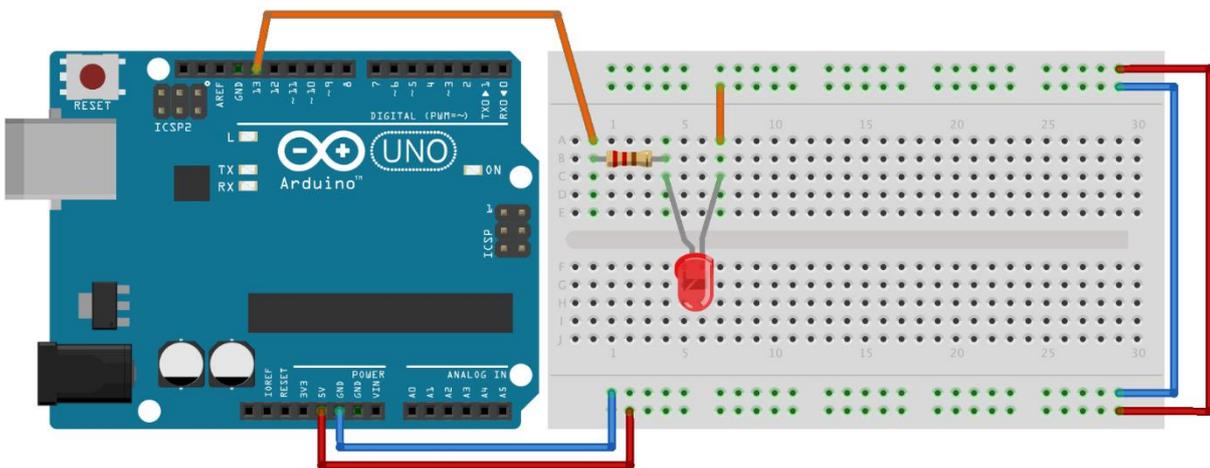
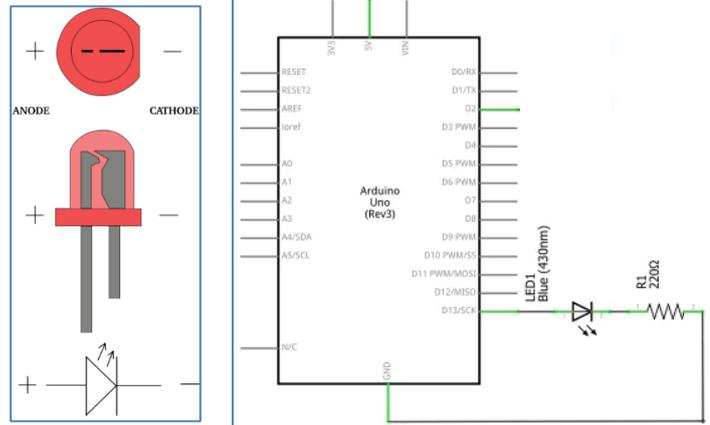
Q9 : Exprimer U_2 en fonction de U , R_1 et R_2

Q10 : Sachant que U est à 5V, quelle est la valeur de U_2 lorsque $R_1=510\Omega$ et $R_2=220\Omega$. Faire le montage pour vérifier l'exactitude de votre réponse.

Q11 : Pour que U_2 soit $U/5$, comment doivent être la valeur de R_1 par rapport à celle de R_2 . Faire un montage permettant de le vérifier. Expliquer votre démarche.

8. Faire clignoter la diode branchée sur la sortie 13

Réaliser le montage suivant. Attention à bien respecter le sens de branchement de la diode
La valeur de la résistance est de 220Ω .



fritzing

<code>int pinLed=13;</code>	1	Attendre 1 seconde
<code>void setup() {</code>	2	Attendre 1 seconde
<code> pinMode(pinLed, OUTPUT);</code>	3	Définir la variable
<code>}</code>		Initialiser la variable comme une sortie
<code>void loop() {</code>	4	Allumer la diode
<code> digitalWrite(pinLed, HIGH);</code>	5	Initialisation
<code> delay(1000);</code>	6	Eteindre la diode
<code> digitalWrite(pinLed, LOW);</code>	7	Faire en boucle
<code> delay(1000);</code>	8	
<code>}</code>		

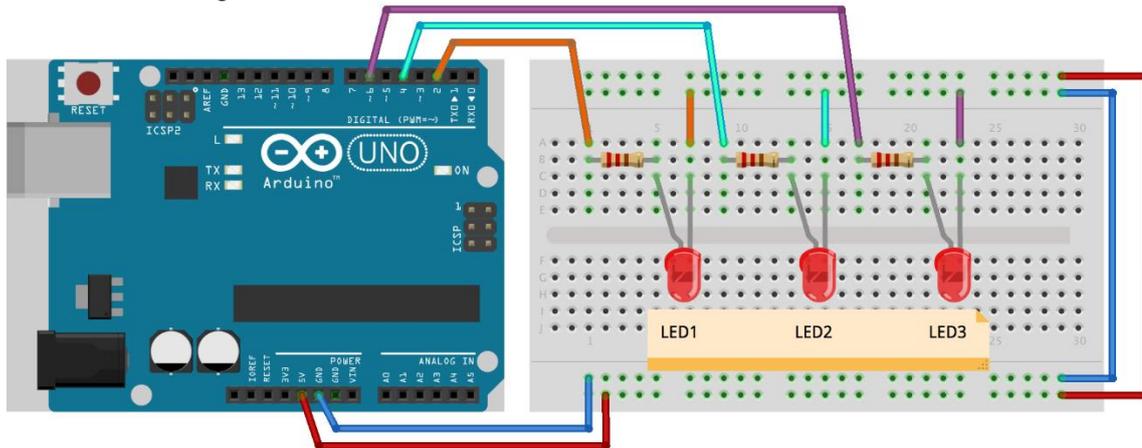
Q12 : Associer les différentes parties du programme à leur fonction

Tester le programme, soit :

- En utilisant l'IDE Arduino en double cliquant sur :  , puis en téléversant le programme sur la carte
- En lançant la simulation sur le site Tinkercad : <https://www.tinkercad.com/>

9. Ex.3: Faire un chenillard de diodes

Réaliser le montage suivant :

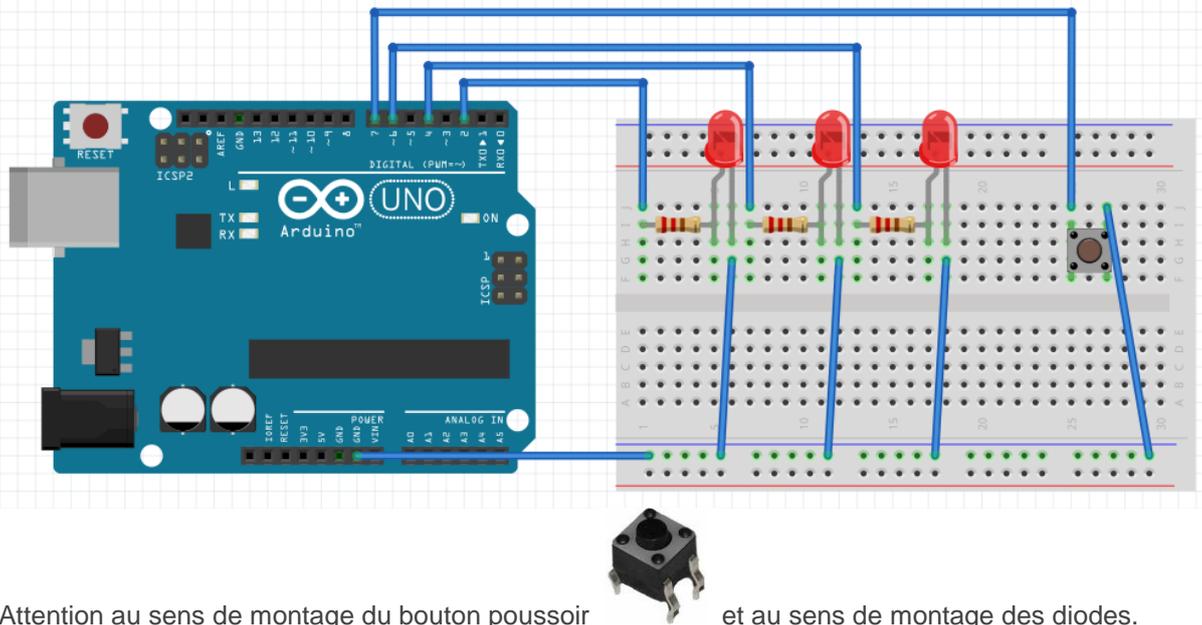


<pre>int pinLed1, pinLed2, pinLed3;</pre>	1	Allumer la diode 1, attendre 0.5s puis allumer la diode 2, attendre 0.5s, puis allumer la diode 3 et attendre 0.5s
<pre>void setup() { pinLed1 = 2; pinLed2 = 4; pinLed3 = 6; </pre>	2	Attendre 0.5s
<pre> pinMode(pinLed1, OUTPUT); pinMode(pinLed2, OUTPUT); pinMode(pinLed3, OUTPUT); </pre>	3	Initialiser les variables en indiquant que ce sont des sorties (output)
<pre> digitalWrite(pinLed1, LOW); digitalWrite(pinLed2, LOW); digitalWrite(pinLed3, LOW); </pre>	4	Eteindre les 3 diodes
<pre>}</pre>		Initialiser l'état des diodes en les éteignant
<pre>void loop() { digitalWrite(pinLed1, HIGH); delay(500); digitalWrite(pinLed2, HIGH); delay(500); digitalWrite(pinLed3, HIGH); delay(500); </pre>	5	Initialiser les variables en indiquant sur quelle sortie elles sont branchées
<pre> digitalWrite(pinLed1, LOW); digitalWrite(pinLed2, LOW); digitalWrite(pinLed3, LOW); delay(500); </pre>	6	Définir les 3 variables
<pre>}</pre>	7	

Q13 : Associer les différentes parties du programme à leur fonction

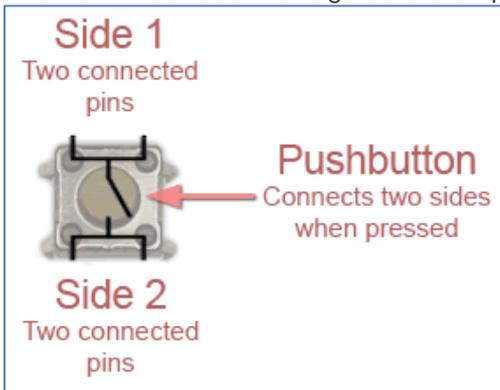
10. Faire s'allumer les diodes lorsque l'on appuie sur un bouton

Réaliser le montage suivant :

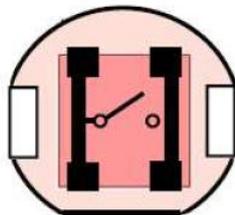


Attention au sens de montage du bouton poussoir

et au sens de montage des diodes.

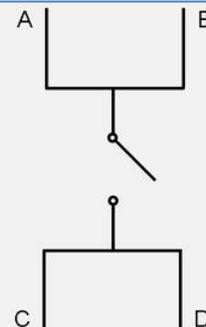
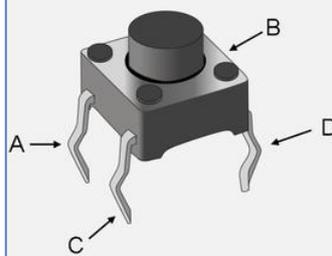


Connexion intérieure



Le bouton fourni a 4 connecteurs, mais seulement 2 sont utilisés. 2 sont interconnectés

eachwithict.com



Une fois le programme analysé, recopiez-le sur Arduino et testez-le.

Vous pouvez maintenant changer le motif d'allumage des diodes et/ou en ajouter.

<pre>int pinLed1, pinLed2, pinLed3; int pinBouton;</pre>	 <p>Allumer la diode 1, attendre 0.5s puis allumer la diode 2, attendre 0.5s, puis allumer la diode 3 et attendre 0.5s</p>
<pre>void setup() { pinLed1 = 2; pinLed2 = 4; pinLed3 = 6; pinBouton = 7;</pre>	 <p>Faire une boucle infinie</p>
<pre>pinMode (pinLed1, OUTPUT); pinMode (pinLed2, OUTPUT); pinMode (pinLed3, OUTPUT); pinMode (7, INPUT_PULLUP);</pre>	  <p>Attendre 0.5s</p>
<pre>digitalWrite (pinLed1, LOW); digitalWrite (pinLed2, LOW); digitalWrite (pinLed3, LOW); }</pre>	 <p>Initialiser les variables associées aux diodes en indiquant que ce sont des sorties (output)</p>
<pre>void loop() { boolean etatBouton = digitalRead (pinBouton);</pre>	<p>Eteindre les 3 diodes</p>
<pre>if (etatBouton==LOW) { while(1) {</pre>	  <p>Initialiser l'état des diodes en les éteignant</p>
<pre>digitalWrite (pinLed1, HIGH); delay(500); digitalWrite (pinLed2, HIGH); delay(500); digitalWrite (pinLed3, HIGH); delay(500);</pre>	 <p>Initialiser la variable associée au bouton poussoir en indiquant que c'est une entrée (input)</p>
<pre>digitalWrite (pinLed1, LOW); digitalWrite (pinLed2, LOW); digitalWrite (pinLed3, LOW); delay(500);</pre>	 <p>Tester l'état du bouton poussoir pour voir s'il est enfoncé (low)</p>
<pre>digitalWrite (pinLed1, LOW); digitalWrite (pinLed2, LOW); digitalWrite (pinLed3, LOW); delay(500); } } }</pre>	 <p>Initialiser les variables en indiquant sur quelle sortie elles sont branchées</p> <p>Définir les 4 variables</p>

Q14 : Associer les différentes parties du programme à leur fonction

11. Brancher les composants sur la carte

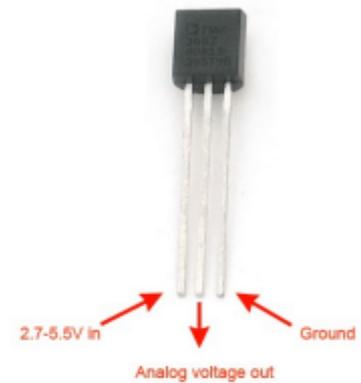
Le prototype étudié contient :

- Une led (et sa résistance de 220Ω)
- Un bouton poussoir et sa résistance de pull up (1 000Ω)
- Un capteur de distance SEN0042
- Un capteur de température TMP36
- Un accéléromètre ADXL345

Documentations techniques :**TMP36 :**

These sensors use a solid-state technique to determine the temperature. That is to say, they don't use mercury (like old thermometers), bimetalic strips (like in some home thermometers or stoves), nor do they use thermistors (temperature sensitive resistors). Instead, they use the fact as temperature increases, the voltage across a diode increases at a known rate. (Technically, this is actually the voltage drop between the base and emitter - the V_{be} - of a transistor.) By precisely amplifying the voltage change, it is easy to generate an analog signal that is directly proportional to temperature (0-5V). There have been some improvements on the technique but, essentially that is how temperature is measured.

To convert the voltage to temperature, simply use the formula: Temp in °C = $[(V_{out} \text{ in mV}) - 500] / 10$

**Triple Axis Accelerometer Breakout - ADXL345**

This new version adds 2 standoff holes as well as an extra decoupling capacitor. The ADXL345 is a small, thin, low power, 3-axis MEMS accelerometer with high resolution (13-bit) measurement at up to ± 16 g. Digital output data is formatted as 16-bit twos complement and is accessible through I2C digital interface (SDA, SCL).



The ADXL345 is well suited to measures the static acceleration of gravity in tilt-sensing applications, as well as dynamic acceleration resulting from motion or shock. Its high resolution (4 mg/LSB) enables measurement of inclination changes less than 1.0 degrees;

Simplest wiring : VCC, GND, SCL and SDA

SEN0042

This is an edge detection sensor from DFRobot. It will help your robot detect the edge of a precipice, preventing it from falling off a table or down the stairs to it's certain demise! This IR distance sensor is connected to an arduino digital pin.

**Specification**

Supply Voltage: 2.7~6.2v

Current: < 10mA

Range distance: 2~10cm (Low), <2cm or >10cm (High)

Interface: 1 digital pin

Signal Voltage: Vcc-0.6V(High), 0.6V(Low)

Q15 : La diode doit-elle être reliée à une entrée ou à une sortie sur la carte Arduino ? Justifiez votre réponse. Cette entrée ou sortie doit-elle être analogique ou numérique ?

Q16 : Le bouton poussoir doit-il être relié à une entrée ou à une sortie sur la carte Arduino ? Justifiez votre réponse. Cette entrée ou sortie doit-elle être analogique ou numérique ?

Q17 : Pour chaque capteur donner le type et les caractéristiques du signal fourni

Q18 : Proposer un câblage pour le montage :

