

Exercice 1 : Capteur de luminosité

Dans cet exercice on demande de créer un programme permettant d'allumer une LED branchée sur la broche 13. L'intensité lumineuse de la LED sera proportionnelle à la luminosité captée par la photo-résistance branchée sur la broche A0 (notez bien le A0, le A qui précède le 0 signifie que c'est une entrée Analogique).

1. Réaliser le montage électronique,
2. Donner le code de programme

Exercice 2 : Feu tricolore et feu piéton

Dans cet exercice on demande de réaliser un feu tricolore accompagné d'un feu "piéton" bicolore (rouge et vert). Le montage devra également avoir un bouton pour piétons qui viendra passer le feu au orange puis au rouge tout en passant le feu piéton au vert (un peu comme dans la vraie vie finalement).

Fonctionnement normal :

Feu tricolore :

- Une LED orange (branchée sur la broche 11) allumée pendant 1 seconde
- Une LED rouge (branchée sur la broche 12) allumée pendant 3 secondes
- Une LED verte (branchée sur la broche 10) allumée pendant 3 secondes

Feu bicolore (piéton)

- Le feu piéton est constamment rouge (rouge→9 et vert→8) tant que le bouton (broche 2) n'est pas appuyé.

Si le bouton est maintenu appuyé pendant que le feu voiture est au vert, alors le feu tricolore passe au orange allumé pendant 1 seconde puis rouge pendant 5 secondes durant lesquelles le feu vert piéton passe au vert.

Après ces 5 secondes, le feu piéton repasse au rouge et le fonctionnement normal reprend.

1. Donner le code de la fonction setup() qui assure le fonctionnement correct du système.
2. Donner le code de la fonction loop() qui assure le fonctionnement

Exercice 3 : Bouton poussoir

- Objectif : Interfacer Arduino avec des capteurs numériques

Dans cet exercice, on demande de réaliser un programme permettant d'allumer une LED branchée sur la broche 13 lorsque le bouton branché sur la broche 2 est appuyé.

1. Réaliser le montage électronique,
2. Donner le code de programme.

Corrigé 1

- Un Arduino
- Un câble USB
- Deux résistances de 1kΩ
- Des fils de prototypage
- Une platine de prototypage
- Une photorésistance
- Une LED de votre couleur préférée

Code Source

```
1. // Initialisation des constantes :
2. const int analogInPin = A0;    // Numéro de la broche à laquelle est con
   necté la photorésistance
3. const int analogOutPin = 13;    // Numéro de la broche à laquelle est co
   nnectée la LED
4.
5. int sensorValue = 0;            // Valeur lue sur la photorésistance
6. int outputValue = 0;           // Valeur envoyée à la LED
7.
8. void setup() {
9.     // Initialise la communication avec l'ordinateur
10.    Serial.begin(9600);
11.
12.    // Indique que la broche analogOutPin est une sortie :
```

```

13.     pinMode(analogOutPin, OUTPUT);
14.     // Indique que la broche analogInPin est une entrée :
15.     pinMode(analogInPin, INPUT);
16. }
17.
18. void loop() {
19.     // lit la valeur de la photorésistance et
20.     // stocke le résultat dans sensorValue :
21.     sensorValue = analogRead(analogInPin);
22.     // change sensorValue vers une intervalle de 0 à 255
23.     // et stocke le résultat dans outputValue :
24.     outputValue = map(sensorValue, 0, 1023, 0, 255);
25.     // envoie de cette nouvelle valeur sur la LED
26.     analogWrite(analogOutPin, outputValue);
27.
28.     // envoie tout ça vers l'ordinateur
29.     Serial.print("sensor = ");
30.     Serial.print(sensorValue);
31.     Serial.print("\t output = ");
32.     Serial.println(outputValue);
33. }

```

```

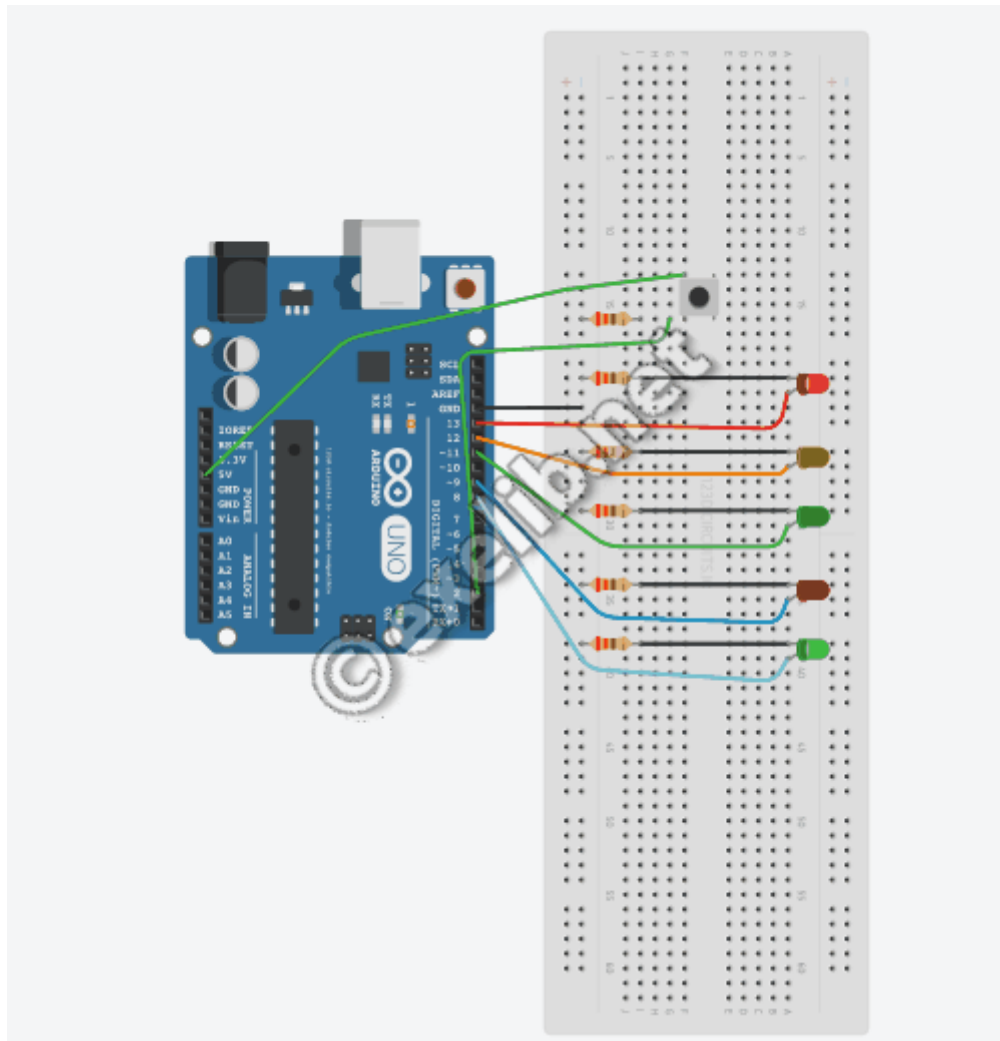
// Déclaration de variables globales
const int BROCHE_LED = 13;
const unsigned int compteur = 2500;

// Fonction d'initialisation : lancée une fois, au début du
programme
void setup() {

```

Corrigé 2

Montage électronique



Montage : Feu tricolore et feu piéton

Pour réaliser ce montage, vous avez besoin de :

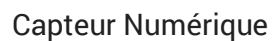
- Un Arduino
- Une platine de prototypage
- Un câble USB
- Cinq résistances de 220Ω
- Des fils de prototypage
- deux LED verte
- Une LED orange

- Deux LED rouge
- Un bouton poussoir

Code Source

```
1. // Initialisation des constantes pour les LED
2. const int ledRouge = 12;
3. const int ledOrange = 11;
4. const int ledVerte = 10;
5.
6. const int ledRougePieton = 9;
7. const int ledVertePieton = 8;
8.
9. // Numéro de la broche à laquelle est connecté le bouton poussoir
10.     const int boutonPin = 2;
11.
12.     // Déclaration des variables :
13.     int boutonState = 0;
14.
15.     // le code dans cette fonction est exécuté une fois au début
16. void setup() {
17.     // indique que les broches des LED
18.     // sont des sorties :
19.     pinMode(ledRouge, OUTPUT);
20.     pinMode(ledOrange, OUTPUT);
21.     pinMode(ledVerte, OUTPUT);
22.
23.     pinMode(ledRougePieton, OUTPUT);
24.     pinMode(ledVertePieton, OUTPUT);
25.
26.     // indique que la broche bouton est une entrée :
27.     pinMode(boutonPin, INPUT);
28. }
29.
30. // le code dans cette fonction est exécuté en boucle
31. void loop() {
```

```
32.          // Dans le fonctionnement normal, le feu piéton est toujours
           rouge
33.          digitalWrite(ledRougePieton, HIGH);
34.          // Fonctionnement normal du feu voiture
35.          digitalWrite(ledVerte, HIGH);
36.          delay(3000);
37.          digitalWrite(ledVerte, LOW);
38.          // lit l'état du bouton et stocke le résultat
39.          // dans etatBouton :
40.          boutonState = digitalRead(boutonPin);
41.
42.          // Si boutonState est égal à HIGH
43.          // c'est que le bouton est appuyé
44.          if (boutonState == HIGH) {
45.              digitalWrite(ledOrange, HIGH);
46.              delay(1000);
47.              digitalWrite(ledOrange, LOW);
48.              digitalWrite(ledRouge, HIGH);
49.              // Le feu piéton passe au vert pendant 5s
50.              digitalWrite(ledRougePieton, LOW);
51.              digitalWrite(ledVertePieton, HIGH);
52.              delay(5000);
53.              // On remet le feu piéton au rouge
54.              digitalWrite(ledRougePieton, HIGH);
55.              digitalWrite(ledVertePieton, LOW);
56.              // Puis on remet le feu au rouge
57.              digitalWrite(ledRouge, LOW);
58.          }else {
59.              // Fonctionnement normal du feu voiture
60.              digitalWrite(ledOrange, HIGH);
61.              delay(1000);
62.              digitalWrite(ledOrange, LOW);
63.              digitalWrite(ledRouge, HIGH);
64.              delay(3000);
65.              digitalWrite(ledRouge, LOW);
66.          }
67.      }
```



- Un Arduino
- Un câble USB
- Deux résistances de 1K Ω
- Des fils de prototypage
- Une platine de prototypage
- Un bouton poussoir
- Une LED de votre couleur préférée

Code Source

```
1. /*
2.   Bouton
3.
4.   Allume une LED branchée sur la broche 13 lorsque le bouton
5.   branché sur la broche 2 est appuyé.
6. */
7.
8. // Initialisation des constantes :
9. const int buttonPin = 2;      // Numéro de la broche à laquelle est conn
   ecté le bouton poussoir
10.     const int ledPin = 13;    // Numéro de la broche à laquelle e
   st connectée la LED
11.
12.     // Déclaration des variables :
13.     int buttonState = 0;       // variable qui sera utilisée pour
   stocker l'état du bouton
14.
15.     // le code dans cette fonction est exécuté une fois au début
16. void setup() {
17.     // indique que la broche ledPin est une sortie :
18.     pinMode(ledPin, OUTPUT);
19.     // indique que la broche buttonPin est une entrée :
20.     pinMode(buttonPin, INPUT);
21. }
22.
23.     // le code dans cette fonction est exécuté en boucle
24. void loop(){
25.     // lit l'état du bouton et stocke le résultat
26.     // dans buttonState :
27.     buttonState = digitalRead(buttonPin);
28.
29.     // Si buttonState est à 5V (HIGH→bouton appuyé)
30.     if (buttonState == HIGH) {
31.         // on allume la LED
32.         digitalWrite(ledPin, HIGH);
33.     }
```



```
34.         else {
35.             // sinon on éteint
36.             digitalWrite(ledPin, LOW);
37.         }
38.     }
```
