

Compte rendu du projet de Formation à Distance

LOIRAT Yoann du Lycée de Val de Saône

Développement d'un clignotant pour cycliste avec ThinkerCAD :

Objectif de la formation :

Il s'agit d'une séance de type « travaux pratique » :
Création d'une IHM à partir de deux boutons et de six LED.



L'objectif de la formation est de développer une capacité attendu du programme de 1^{ière} NSI lié à la robotique :

- Identifier le rôle des capteurs et actionneurs. Réaliser par programmation une IHM répondant à un cahier des charges donné.

Selon le thème de la robotique dans l'UE3 :

- Les entrées et sorties numériques : configuration, lecture d'informations de capteurs numériques et écriture d'informations vers des actionneurs numériques. Mise en œuvre sur simulateur et exemples d'applications pratiques.

Cette activité permet de travailler sur les cartes Arduino sans avoir besoin d'acheter le matériel. Ainsi elle peut être réalisé en distanciel comme en présentiel en salle informatique sans matériel spécifique.

Prérequis de la formation :

- Connaissance simple de la structure de programmation d'un microcontrôleur (Setup/Loop) et de la syntaxe spécifique du code Arduino par rapport au Python :
 - Fonction avec les crochets `{ }` et indentation non nécessaire
 - Finir toutes les instructions par `;`
 - Les commentaires sont indiqués par `//` ou `/*commentaire*/`
 - La boucle `for` en langage Arduino est légèrement différente de Python

Temps imparti : 60 minutes

Liens vers les plateformes d'apprentissage distanciel :

La Quizinière est une plateforme de formation à distance développée par le réseau Canopé.

Code ou lien de l'activité sur l'espace apprenant : EBKXY6

<https://www.quiziniere.com/#/Exercice/EBKXY6>

Lien vers l'espace de classe thinkerCAD :

<https://www.tinkercad.com/joinclass/PRPIXVVEUI8>

Exemple de login élève sur l'espace de classe ThinkerCAD : **nomprenom3107**

Description de la formation :

Dans son navigateur, l'élève doit avoir :

- Un onglet avec l'activité sur la quizinère.com où se trouve les explications et les exercices divers qui vont le guider ;
- Un autre onglet sur la plateforme ThinkerCAD pour développer le projet.

Après une description synthétique du projet original sur lequel se base cette activité, on explique l'objectif principale qui est de développer une version simplifiée de ce clignotant.

Dans la première partie du TP, on guide l'élève afin qu'il ouvre le montage **Clignotant** de ThinkerCAD. Cet exemple est un premier montage électronique fonctionnel. Il permet de montrer à l'élève le fonctionnement de la plateforme : code / simulation / assemblage de composant.

Avec des QCM simples, on souligne les fonctions et paramètres du code qui permettent au montage de fonctionner.

Dans la seconde partie, on utilise une platine d'essai afin de complexifier le montage précédant. On demande à l'élève de recopier le montage proposé puis de compléter le code à l'aide de textes à trous. Il précise le rôle des broches utilisées dans la fonction ***void setup()*** avec la fonction ***pinMode***.

Ensuite on lui demande de compléter le code afin de commander l'allumage successif des LED de droite à gauche puis dans un second temps de gauche à droite. Le but est que l'élève apprenne à utiliser une boucle ***for*** ainsi que la fonction ***digitalWrite*** dans le langage Arduino.

Dans la dernière partie du TP, on utilise deux boutons pour commander le montage. L'élève reproduit le montage proposé puis déclare les deux boutons en entrée dans ***void setup()***.

Pour finir on le guide avec la déclaration de deux variables qui vont faciliter l'utilisation des boutons pour commander l'allumage des LED dans un sens et dans l'autre.

J'ai fait le choix d'utiliser principalement l'outil pédagogique du texte à trous car il permet de guider l'élève (fortement) afin qu'il réussisse à faire fonctionner le montage sans perdre trop de temps avec la syntaxe du langage Arduino.

Le but de l'activité n'étant pas de devenir spécialiste du langage Arduino mais de montrer les possibilités des objets intégrant de l'informatique embarqué.

La dernière question permet d'ouvrir la réflexion de l'élève sur les possibilités d'amélioration de ce projet.

Compte rendu du projet de Formation à Distance

Capture d'écran de cette formation et réponses attendues dans l'activité :

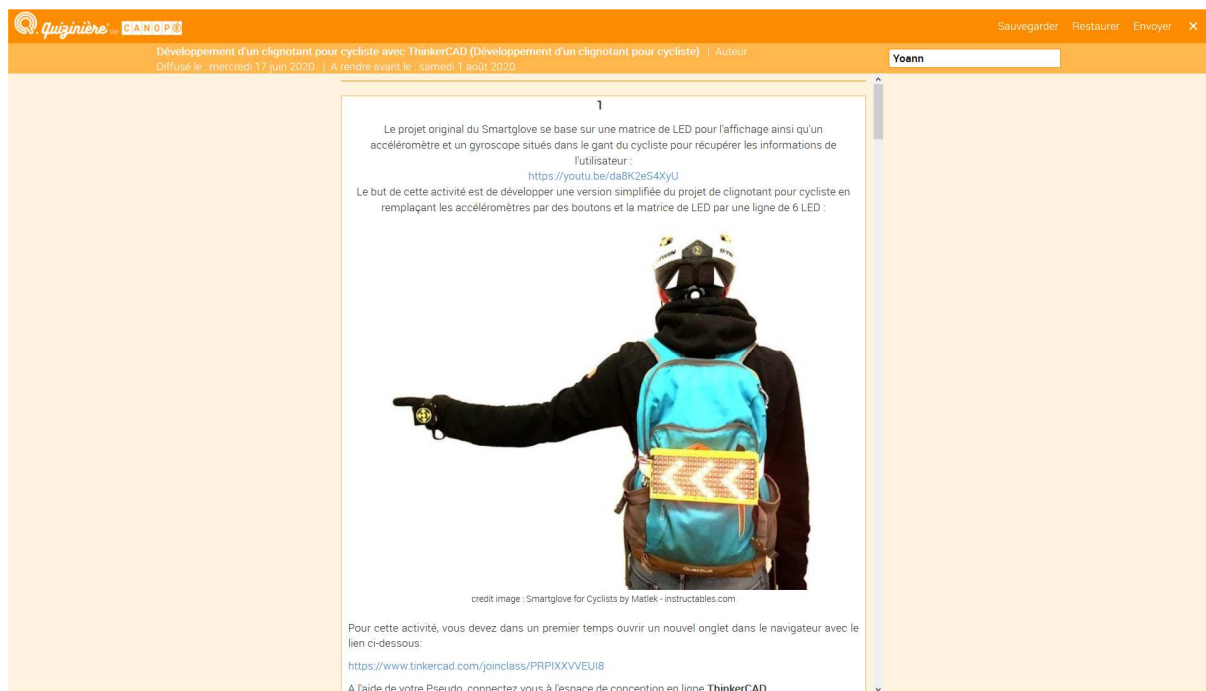


Figure 1- Introduction de l'activité proposé sur la plateforme www.quizziniere.com du réseau CANOPE



Figure 2- Page de connexion à la classe FOAD_robotique avec un exemple d'identifiant fonctionnel

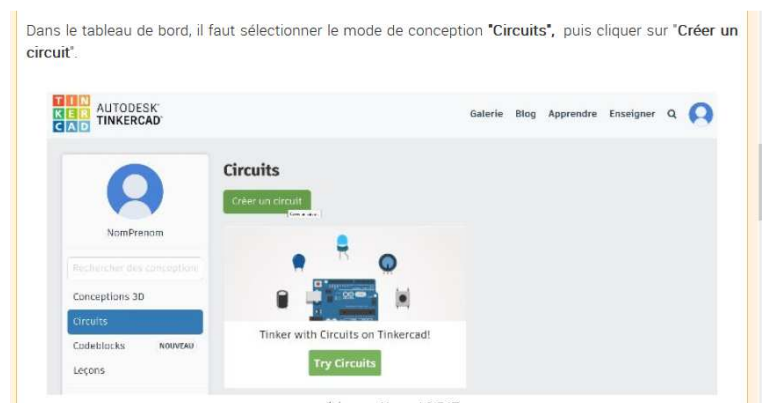


Figure 3- Aide pour naviguer sur la plateforme

Compte rendu du projet de Formation à Distance

Pour notre projet de clignotant pour cycliste, nous allons utiliser les **"Démarreurs" Arduino**.

Plus spécifiquement le montage **"Clignotant"**, sur lequel il y a une LED branchée en série avec une résistance électrique. Vous pouvez bien entendu **"Démarrer la simulation"** pour observer le résultat du programme associé au montage par défaut.



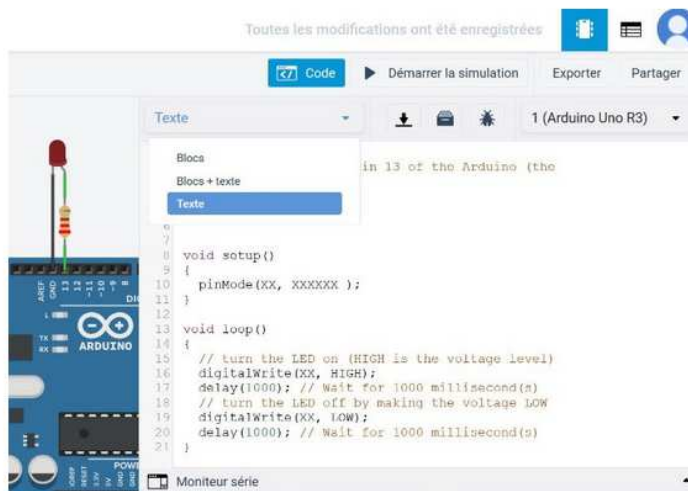
credit image : Yoann LOIRAT

En observant le montage "Clignotant", quelles broches permettent d'allumer la LED?

- ☒ La broche 13
- ☒ La broche GND
- ☐ La broche A0

Figure 4- Question à choix multiple pour analyser le montage électronique

Vous pouvez regarder le code par défaut en cliquant sur le bouton **"Code"** puis en choisissant le mode de programmation **"Texte"**.



credit image : Yoann LOIRAT

Comment doit être déclarée/paramétrée cette broche XX dans la fonction **void setup ()** pour pouvoir allumer la LED ?

- ☐ pinMode(XX, INPUT);
- ☒ pinMode(XX, OUTPUT);

D'après les commentaires du code de l'image ci-dessus et votre observation de la simulation de fonctionnement, à quoi sert la fonction **digitalWrite(XX, HIGH)** ?

- ☐ A établir un niveau bas de tension sur la broche XX
- ☒ A établir un niveau haut de tension sur la broche XX
- ☐ A lire un niveau haut de tension sur la broche

Figure 5- Suite QCM pour lire le code initial du montage Clignotant

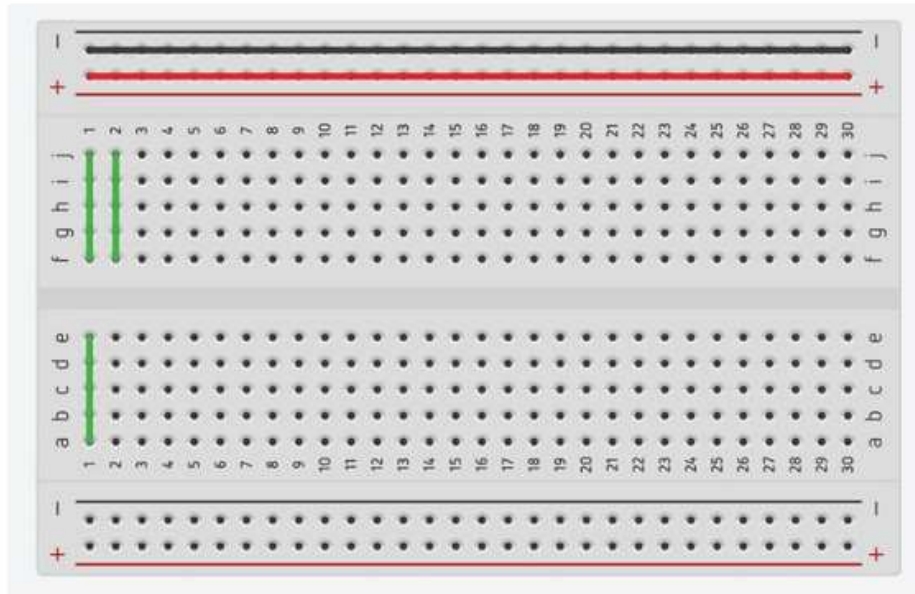
Compte rendu du projet de Formation à Distance

2

On va maintenant modifier ce premier montage et ajouter cinq autres LED avec des résistances de 220 Ohm monté en série. Pour cela il est plus simple d'utiliser une **platine d'essai**.

Vous pouvez la trouver dans le menu déroulant : **Composant : De base**

Une platine d'essai permet de relier électriquement des composants entre eux sans soudure. Ainsi tout les points sous la ligne noire sont reliés entre eux, de même pour la ligne rouge et les lignes vertes :



credit image : Yoann LOIRAT

Rappel : On doit raccorder l'anode à la tension positive et la cathode au 0V du circuit pour qu'elle s'allume.

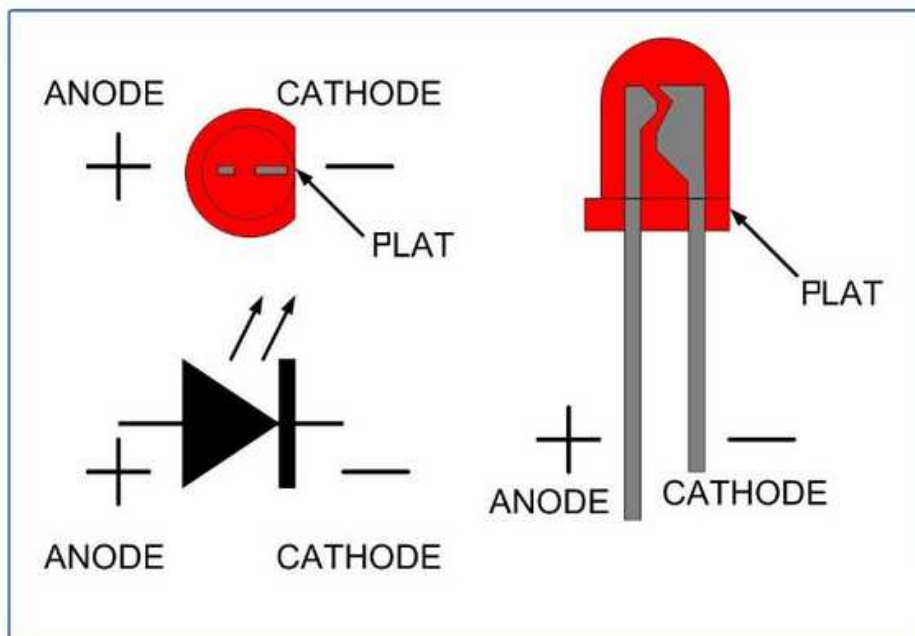


Figure 6- explication technologique sur la platine d'essai et la diode électroluminescente

Compte rendu du projet de Formation à Distance

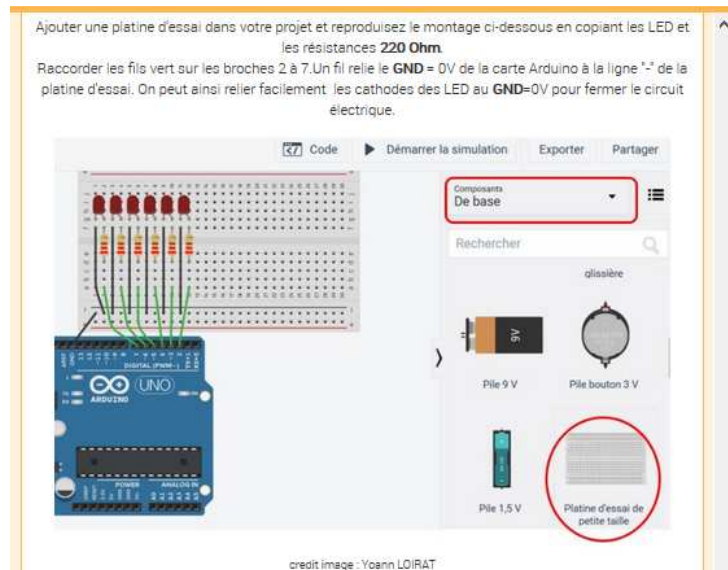


Figure 7-Montage des 6 LED

Les broches de commande vont de 2 à 7. Par conséquent nous allons pouvoir déclarer facilement nos broches dans la fonction **void setup()** avec une boucle **for** dans le langage Arduino.

Compléter le code à trous ci-dessous :

```
void setup()
{
  for (int i = 2; i < 8; i = i + 1)
  {
    pinMode(i, OUTPUT);
  }
}
```

Attention : Il est important de respecter la casse (majuscules / minuscules) afin d'obtenir toutes les bonnes réponses.

Pour allumer les LED successivement de droite à gauche, nous allons réutiliser le code initial **digitalWrite(XX, HIGH ou LOW)** et le combiner avec la boucle **for (int i = x; i < x; i = i + 1)** utilisée précédemment.

Compléter le code suivant et testez le dans ThinkerCAD :

```
void loop()
{
  for (int i = 2; i < 8; i = i + 1)
  {
    digitalWrite(i, HIGH);
    delay(100);
    digitalWrite(i, LOW);
    delay(100);
  }
}
```

Attention : Il est important de respecter la casse (majuscules / minuscules) afin d'obtenir toutes les bonnes réponses.

On veut maintenant programmer le montage afin que le bandeau s'allume de gauche à droite.

Compléter le code suivant et testez le dans ThinkerCAD :

```
void loop()
{
  for (int i = 7; i > 1; i = i - 1)
  {
    digitalWrite(i, HIGH);
    delay(100);
    digitalWrite(i, LOW);
    delay(100);
  }
}
```

Figure 8- texte à trous pour faire fonctionner le nouveau montage

Compte rendu du projet de Formation à Distance

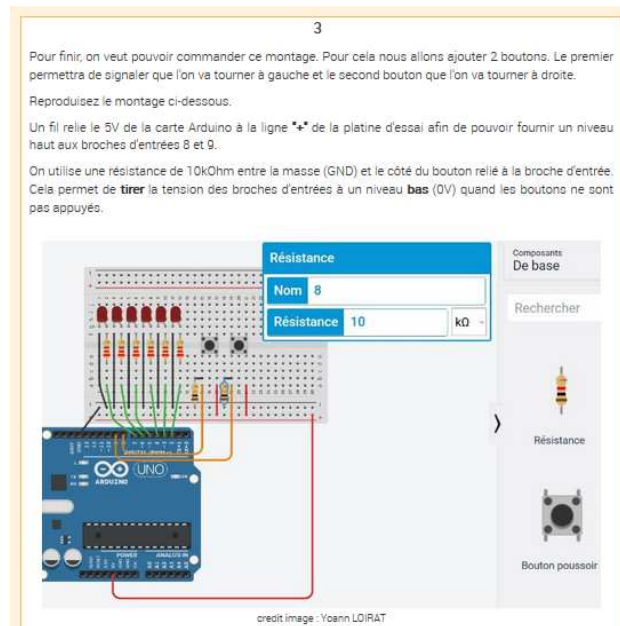


Figure 9- Montage des deux boutons

On va modifier le code de la fonction **void setup()** afin de prendre en compte ces 2 boutons dans le programme. Il faut faire bien attention aux numéros des broches utilisées.

Compléter le code ci-dessous pour déclarer ces 2 boutons dans **void setup()**.

(Utiliser vos réponses précédentes pour la partie concernant les 6 LED)

```
void setup()
{
  for (int i = 2; i < 8; i = i + 1)
  {
    pinMode(i, OUTPUT);
  }
  pinMode(8, INPUT);
  pinMode(9, INPUT);
}
```

Pour faciliter l'utilisation des boutons, on déclare des variables qui vont permettre de stocker l'état des boutons. Ajouter ces deux instructions au début du programme avant la fonction **void setup()** :

```
int bouton_droit=LOW;
int bouton_gauche=LOW;
```

Pour finaliser le programme, compléter la fonction **void loop()** ci-dessous :

```
void loop()
{
  bouton_droit = digitalRead(9);
  bouton_gauche = digitalRead(8);

  if (bouton_gauche == HIGH)
  {
    for (int i = 2; i < 8; i = i + 1)
    {
      digitalWrite(i, HIGH);
      delay(100);
      digitalWrite(i, LOW);
      delay(100);
    }
    delay(500);
  }
  if (bouton_droit == HIGH)
  {
    for (int i = 7; i > 1; i = i - 1)
    {
      digitalWrite(i, HIGH);
      delay(100);
      digitalWrite(i, LOW);
      delay(100);
    }
    delay(500);
  }
}
```

Figure 10- texte à trous pour ajouter les boutons dans le code

Compte rendu du projet de Formation à Distance

Expliquez en quelques mots comment vous pourriez modifier le code ou le montage pour améliorer ce projet de clignotant pour cycliste ?



Mots: 54

Caractères: 299

On peut ajouter une boucle for pour effectuer par exemple 2 ou 3 clignotements complet même si le bouton est relâché. On peut augmenter le nombre de LED et/ou créer de nouvelle animation comme un allumage de toutes les LED quand le cycliste freine et que les deux boutons sont appuyés en même temps.

Figure 11- Question ouverte pour stimuler la créativité de l'élève

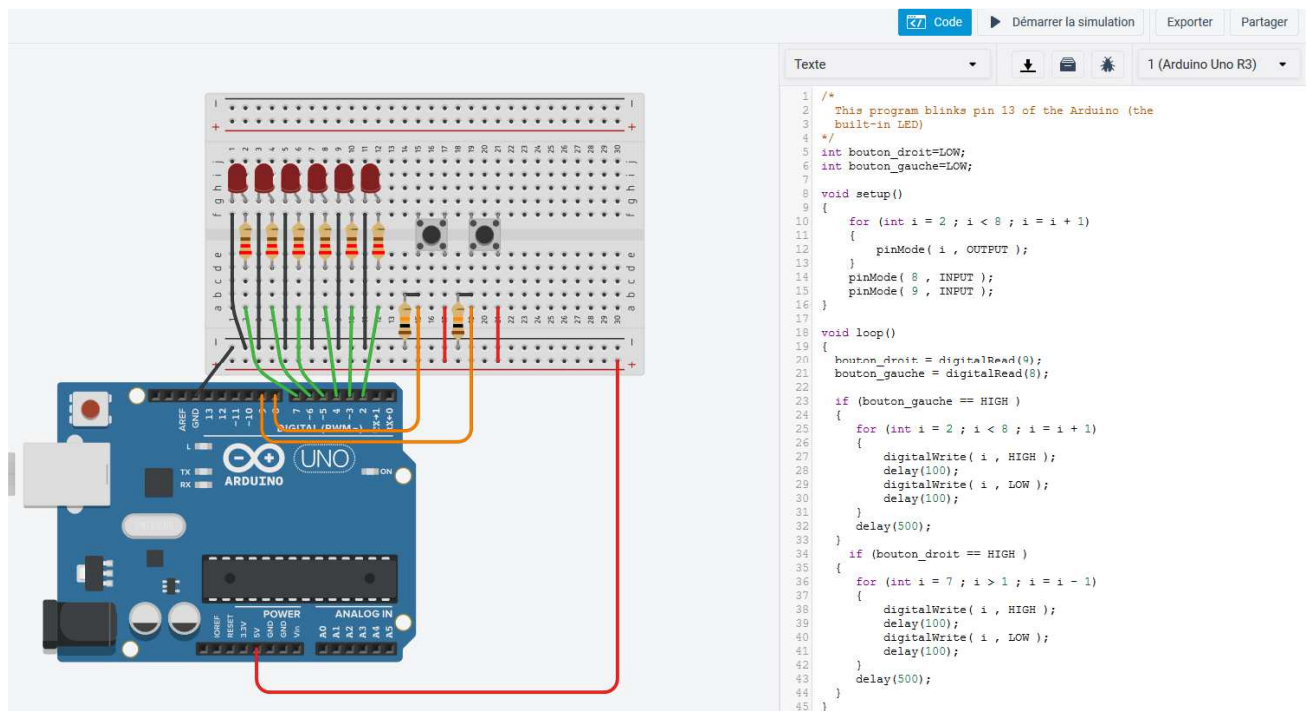


Figure 12-Résultat du projet sur la plateforme thinkerCAD à la fin de l'activité