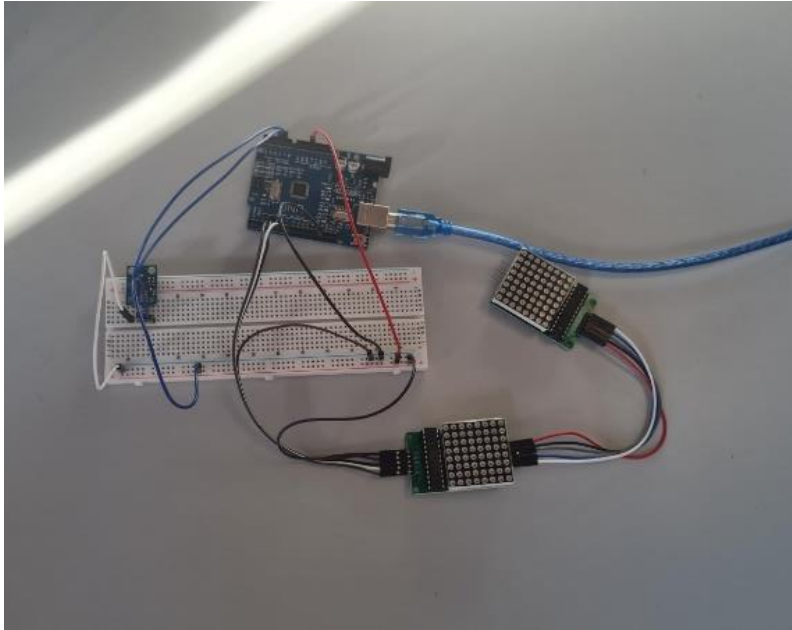


SABLIER NUMERIQUE

Projet Arduino

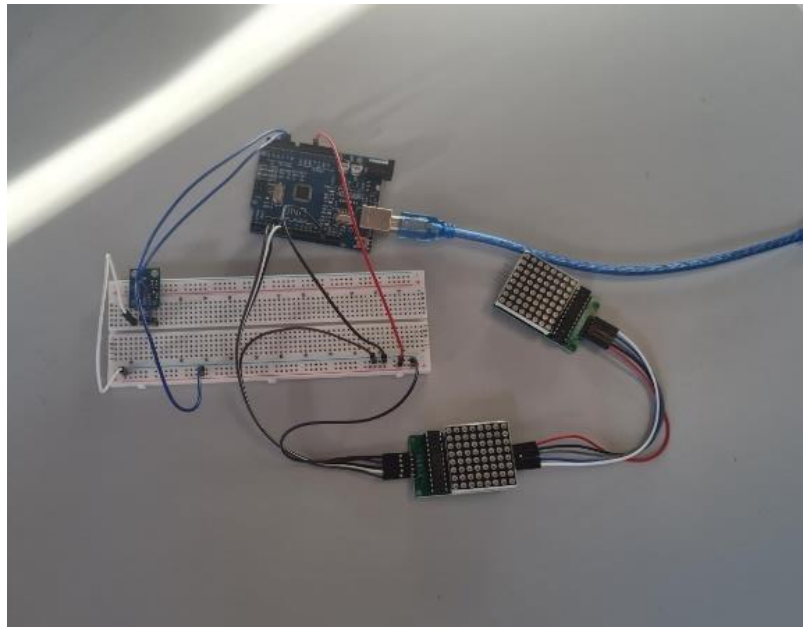


Contributeurs :

ALABEATRIX Axel

BRUCHARD Emmanuel

VADOT Antoine



ESEO - Dijon
3 janvier 2024

Sommaire

- I. Introduction générale
- II. Partie technique
- III. Conclusion et perspectives

I/ Introduction générale

Description générale du projet : un sablier numérique réglable pour différentes durées Arduino qui affiche le temps qui s'écoule sur deux matrices de LED.

Objectifs du projet :

- Mettre en pratique vos connaissances et informatiques et électroniques
- Gagner en connaissances transversales.
- Savoir surmonter les problèmes auxquels on est heurtés.
- Faire de la recherche sur internet : forums, tutos, plateforme Arduino etc.
- Gagner en autonomie

Parties prenantes du projet :

Equipe de projet :

Alabéatrix Axel : Programmeur

Bruchard Emmanuel : Secrétaire et trésorier / montage

Vadot Antoine : CAO

Client : L'ESEO

Contraintes

Le coût total du projet, hors matériel à disposition de l'ESEO, ne doit pas dépasser 40 €.

Le projet doit être réalisé pour le 2 décembre 2023 afin de pouvoir le présenter à la JPO.

Le sablier doit être facile à utiliser, avec des instructions claires pour l'utilisateur.

Pourquoi ce projet ?

Nous avons décidé de moderniser le sablier traditionnel pour le rendre plus polyvalent, plus simple d'utilisation et plus esthétique.

Polyvalence : Un sablier électronique peut être configuré pour mesurer des durées variables, allant de quelques secondes à plusieurs heures. Cela le rend polyvalent pour une utilisation dans différents contextes, que ce soit pour la cuisine, les jeux, la méditation ou d'autres activités nécessitant une gestion précise du temps.

Facilité d'utilisation : Les sabliers électroniques sont généralement dotés d'une interface conviviale et intuitive, avec des boutons ou des écrans tactiles permettant de régler facilement les paramètres de temps.

Esthétique moderne : Un sablier électronique peut avoir un design moderne et attrayant, avec des options de personnalisation pour s'adapter à différents goûts et styles. Cela en fait également un objet de décoration intéressant.

II/ Partie technique

Fonctionnalités

Le sablier est capable de mesurer le temps écoulé et de l'afficher sur deux matrices de LED. L'affichage se présente de deux manières, au choix de l'utilisateur : soit à l'instar d'un sablier standard, soit par le biais d'un compte à rebours avec des valeurs numériques.

Le sablier est réglable pour différentes durées, en utilisant un encodeur rotatif permettant d'augmenter la durée ou de diminuer la durée.

Le sablier est alimenté par une pile.

Le sablier est portable et facile à utiliser.

Matériel requis

Nous avons utilisé des matrices de Led 8*8 pour un affichage clair sans prendre beaucoup de place.

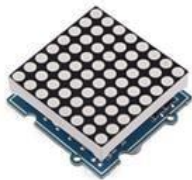




La pile 9V permet l'alimentation du sablier.

La carte Arduino Uno fait office de micro-contrôleur.

L'accéléromètre permet d'obtenir la position des points pour donner l'impression d'avoir un sablier classique.

L'interrupteur permet de choisir d'alimenter ou non le sablier.

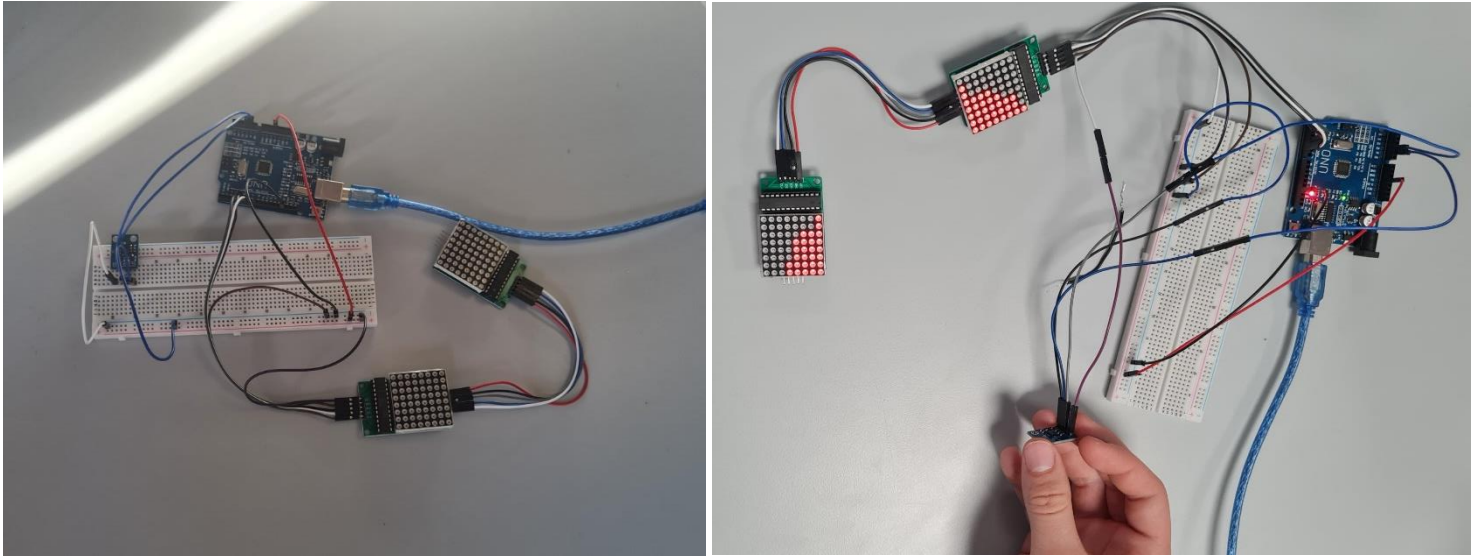
Enfin, l'encodeur rotatif permet de régler le sablier comme l'utilisateur le souhaite.

Référence	Photo	Spécifications caractéristiques
Matrices de LED : Module de contrôle d'affichage MCU LED, 8 x 8, microcontrôleur de matrice de points		<ul style="list-style-type: none"> Résolution : La matrice de LED 8x8 offre une résolution de 8 pixels par 8 pixels, ce qui signifie qu'elle peut afficher jusqu'à 64 points lumineux individuels. Contrôle individuel des LED : Chaque LED dans la matrice peut être contrôlée individuellement. Cela signifie que vous pouvez allumer ou éteindre chaque LED séparément, ce qui permet de créer des motifs, des images ou des animations. Interface : La matrice de LED 8x8 est dotée d'une interface simple.
Carte Arduino UNO		<ul style="list-style-type: none"> Micro contrôleur
Pile : Varta, Long Life Power, 9V		<ul style="list-style-type: none"> Alimentation du sablier
Accéléromètre : module d'accéléromètre 3 axes ADXL335.		<ul style="list-style-type: none"> Axes de mesure : Le module ADXL335 mesure l'accélération sur trois axes : X, Y et Z. Il est capable de détecter l'accélération linéaire dans toutes les directions, ce qui permet de déterminer l'orientation et les mouvements d'un objet.
Interrupteur		<ul style="list-style-type: none"> Fonctionnement binaire : Les interrupteurs fonctionnent selon le principe binaire, ce qui signifie qu'ils ont deux états distincts : ouvert (OFF) ou fermé (ON). Lorsqu'un interrupteur est en position ON, il ferme le circuit et permet au courant de circuler. Lorsqu'il est en position OFF, il ouvre le circuit et interrompt le flux de courant.
Encodeur rotatif		<ul style="list-style-type: none"> Via un appui : l'encodeur permet de préchoisir quelle durée l'utilisateur veut initier (heure/minutes) Via une rotation : l'encodeur permet de choisir la durée (ex : 8 minutes, 3 heures) Plus facile à utiliser par rapport aux boutons poussoir utilisés en E1. De plus il regroupe en 1 ce que 3 boutons poussoirs faisaient.

Montage

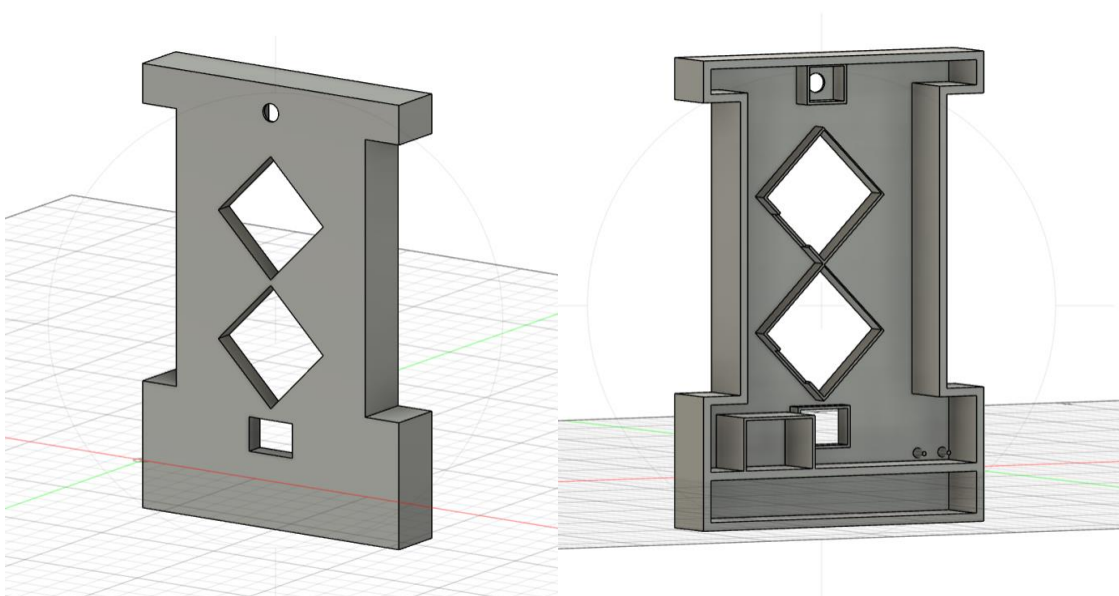
Nous avons commencé la réalisation du sablier par un montage simple associant les matrices de Led et l'accéléromètre.

Par la suite, nous avons téléversé un code sur la carte arduino permettant d'afficher les Led des matrices selon la position de l'accéléromètre.



Pour se familiariser avec les matrices de Led, nous avons réalisé deux mini-projets, présentés en fin de rapport.

Ensuite, nous avons modélisé le boîtier. En haut, l'espace est conçu pour l'encodeur rotatif permettant la modification de la durée souhaitée. Les 2 espaces centraux sont prévus pour les matrices de Led. Le dernier espace permet d'insérer l'interrupteur. A l'intérieur, les différents composants sont maintenus en place par de petite parois, de sorte à pouvoir retourner le sablier. Pour maintenir l'accéléromètre, nous avons modélisé deux « piques » qui viennent s'emboîter dans les trous de ce dernier.



Nous avons ensuite réalisé une deuxième impression pour la partie arrière du sablier qui permettra de fermer le « boîtier ».

Une fois le boîtier imprimé et le code final rédigé, nous avons monté le sablier.

Afin de pouvoir ouvrir et refermer le boîtier, nous avons opté pour un système de vis.

Ci-dessous une photo du sablier entièrement monté :



Conclusion et perspectives

L'objectif du projet est de concevoir un sablier facile à utiliser, qui peut être réglé pour différentes durées et qui est alimenté par une batterie. L'objectif est donc atteint. En effet, le sablier est capable de mesurer le temps écoulé et de l'afficher sur deux matrices de LED ; le sablier est réglable pour différentes durées, en utilisant l'encodeur rotatif ; le sablier est alimenté par une batterie rechargeable ; le sablier est portable et facile à utiliser.

En conclusion, la réalisation du sablier numérique a été un projet intéressant et pratique pour nous, étudiants en école d'ingénieur. Ce projet nous a permis de mettre en pratique nos compétences en électronique, en programmation et en conception de circuit imprimé, tout en offrant une solution pratique pour mesurer le temps.

De plus, ce projet peut également être adapté pour répondre à des besoins spécifiques en ajoutant des fonctionnalités telles que des alarmes sonores ou visuelles. Egalement, il est possible de concevoir un modèle qui peut être contrôlés à distance via une application mobile.

Bibliographie

<https://github.com/wayoda/LedControl/>

<https://www.electronicwings.com/arduino/adxl335-accelerometer-interfacing-with-arduino-uno>

Annexe : code

Voir dans le deuxième fichier partagé.

Annexe : mini-projets

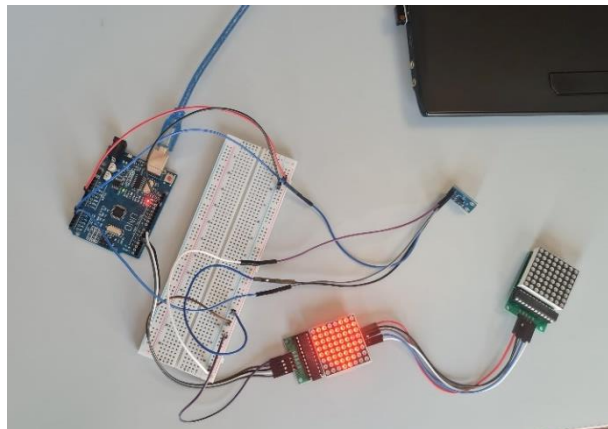
En parallèle du projet, nous nous sommes amusés avec l’affichage matriciel. Nous avons donc travaillé sur des mini-projets que nous nous sommes permis de présenter dans ce rapport.

Premièrement, nous avons réussi à afficher sur une matrice de LED un cœur.

Voici le résultat ainsi que le code nécessaire à la réalisation.

Mini-Projet 1

```
const int MATRIX_A = 0;
const int PIN_DATAIN = 5;
const int PIN_CLK = 4;
const int PIN_LOAD = 6;
LedControl lc = LedControl(PIN_DATAIN, PIN_CLK, PIN_LOAD, 2);
byte shape[8] = {
  B01100110,
  B11111111,
  B11111111,
  B11111111,
  B11111111,
  B11111111,
  B01111110,
  B00111100,
  B00011000
};
};
void setup() {
  lc.shutdown(MATRIX_A, false);
  lc.setIntensity(MATRIX_A, 8);
  lc.clearDisplay(MATRIX_A);
  for (byte row = 0; row < 8; row++) {
    lc.setRow(MATRIX_A, row, shape[row]);
  }
}
void loop() {
}
```

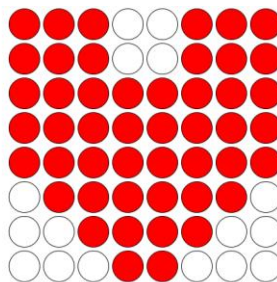


Deuxièmement, nous avons réussi à afficher une matrice de LED en HTML où l'on peut sélectionner les LED à allumer. Sur un presse-papier, est copié le code binaire associé à la matrice. Ceci peut être inséré dans le code du mini-projet 1, ainsi sur la matrice physique on a le rendu souhaité, rentré sur le lien HTML.

Voici le code copié dans le presse-papier à partir de la sélection de LED sur la matrice, affichée à la suite du code généré.

Mini-Projet 2

```
byte shape[8] = {
  B11100111,
  B11100111,
  B11111111,
  B11111111,
  B11111111,
  B01111110,
  B00111100,
  B00011000
}
```



Copy to clipboard