
	Sciences et Technologies de l'Industrie et du Développement Durable		
	2I2D		
	<i>Clavier + Servo</i>		TP

1. Simulation

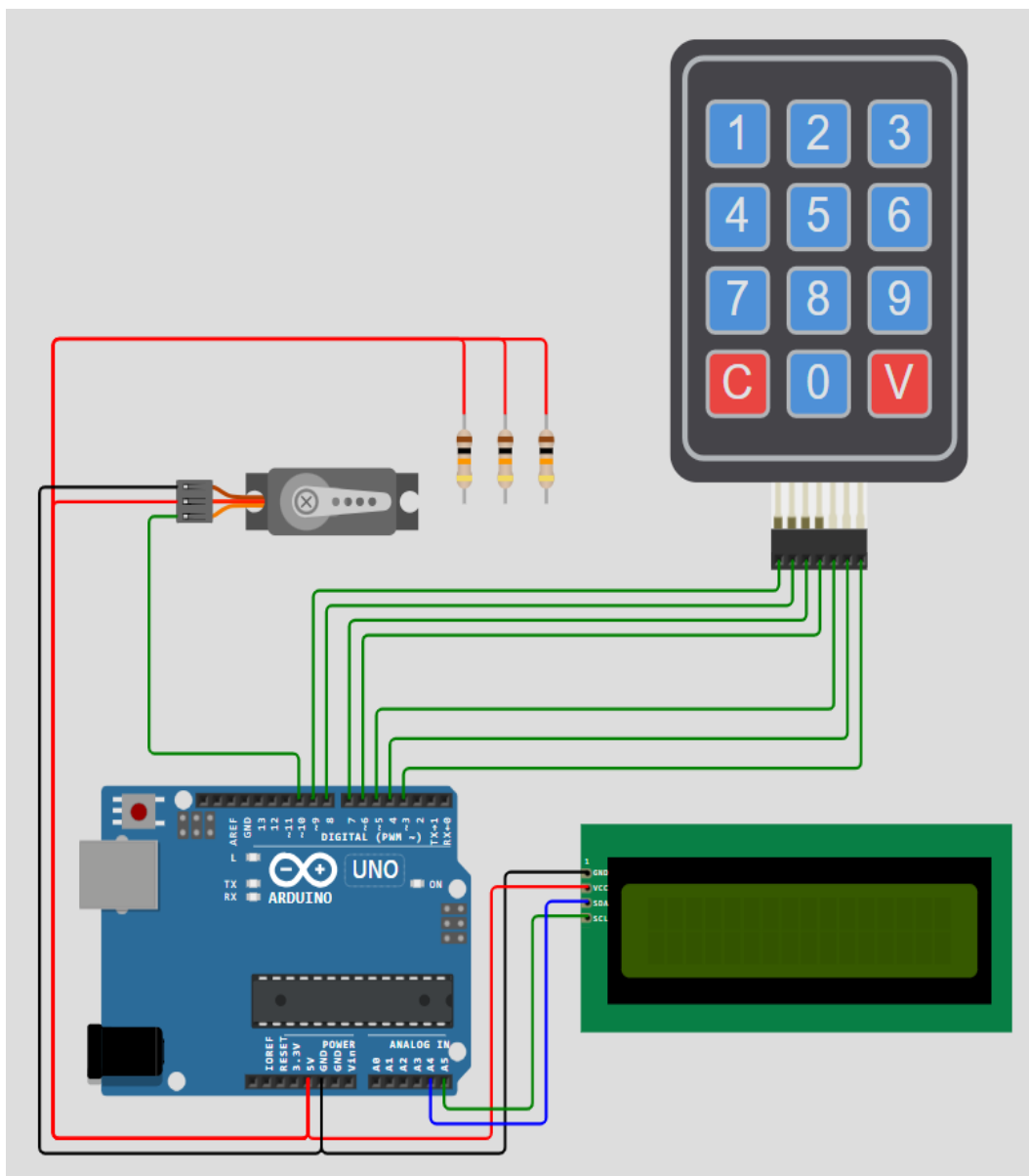
- I. Refaire le câblage sur wokwi (cf document ressource) et câbler les résistances en pull up sur chacune des colonnes du clavier.
- II. Créer une fonction OuvrirBarriere() qui fait : Rotation à 90° (ouverture), Attente 3 secondes, Retour à 0° (fermeture)
- III. Récupérer le code Progr_eleve le mettre dans wokwi, tester et corriger les erreurs s'il y en a. Si l'utilisateur saisie le bon code, le programme dans lancer la fonction OuvrirBarriere().
- IV. *Faire valider par le professeur.*

2. Expérimentation

- I. Réaliser HORS TENSION le câblage.
- II. *Faire valider par le professeur.*
- III. Tester le programme.
- IV. *Faire valider par le professeur.*

3. Ressources

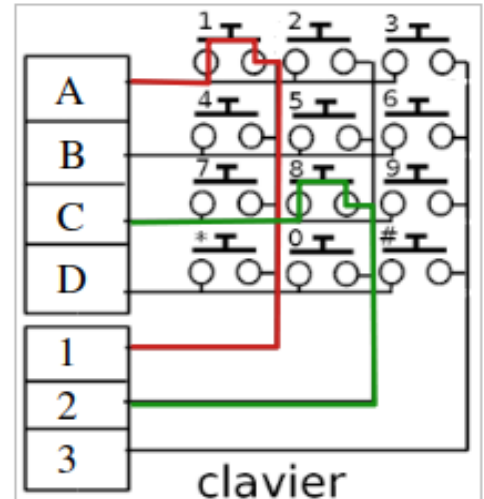
3.1. Câblage wokwi



3.2. Principe du clavier matricé

Tableau de correspondance ligne/colonne :

	1	2	3
A	1	2	3
B	4	5	6
C	7	8	9
D	*	0	#



Exemple :

- Si on appuie sur la touche " 1 ", on va relier " A " à " 1 "
- Si on appuie sur la touche " 8 ", on va relier " C " à " 2 "

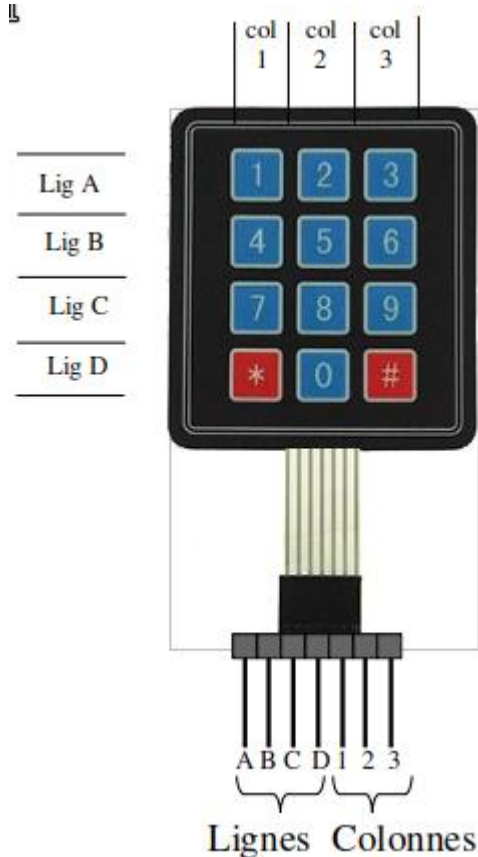
Pour un clavier 12 touches, on a besoin de 7 broches :

- 3 broches pour les 3 colonnes
- 4 broches pour les 4 lignes

À chaque fois que l'on appuie sur une touche, un contact à l'intérieur du clavier vient relier les broches correspondant à la colonne et la ligne de la touche appuyée.

Le principe pour détecter qu'une touche est appuyée est le suivant :

- 1- Appliquer un niveau logique 0 sur la broche d'une des 4 lignes. Les 3 autres lignes sont positionnées au niveau logique 1.
- 2- Regarder sur chacune des broches correspondant aux colonnes si on retrouve ce niveau logique 0. Si on retrouve ce niveau logique 0 sur une colonne, cela signifie que la touche située à l'intersection de cette colonne et de la ligne au niveau logique 0, est enfoncée.
- 3- On passe à la ligne suivante et on recommence.
- 4- On réalise ces opérations suffisamment rapidement pour détecter même de brefs appuis sur le clavier.



3.3. Connexions sur simulateur

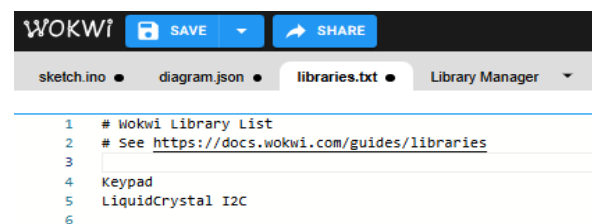
Pour gérer/programmer le clavier sous Arduino, on utilise ici le logiciel en ligne WOKWI à l'adresse : <https://wokwi.com/projects/new/arduino-uno>.

La mise en place de l'environnement de simulation se fait en 4 étapes :

1. Le système est initialement démarré avec 3 onglets (`sketch.ino`, `diagram.json`, Library Manager).
2. Insérer le schéma de démarrage donné par le fichier `diagram.json` en réalisant un copier-coller à partir d'un éditeur de texte (Notepad++, bloc-note, nano, etc.). Cela permettra d'obtenir le diagramme présenté en page 2 (pour le premier programme).
 - Pour ajouter une connexion (un câble), cliquer sur une broche à connecter, puis cliquer sur la broche cible (la broche à connecter à l'aide du câble).
 - Pour ajouter un composant, appuyer sur la touche « a » du clavier puis sélectionner le composant voulu.
3. Pour ajouter les différents programmes à compléter dans la partie « conception », sur le même principe que pour les diagrammes, il suffit de copier le contenu du fichier programme dans l'onglet `sketch.ino`.



4. Il reste enfin à ajouter les librairies requises par le système. Pour cela, dans l'onglet Library Manager, en utilisant le bouton + visible, ajouter les librairies suivantes :
 - Keypad
 - LiquidCrystal_I2C



Dès lors, un nouvel onglet sera visible (`libraries.txt`), récapitulant les bibliothèques utilisées.

3.4. Aide à la programmation

Les objets créés

<pre>Reservation { String code; int etage; int place; }</pre>	<p>L'objet <code>Reservation</code> contient trois paramètres : le code de réservation, l'étage de la place réservée, ainsi que le numéro de la place réservée. Pour accéder aux différents éléments :</p> <ul style="list-style-type: none"> • <code>objet.code</code> • <code>objet.etage</code> • <code>objet.place</code>
---	--

Les fonctions

<code>AfficherPlace(Reservation data)</code>	Affiche la place affectée à la réservation fournie en paramètre.
<code>AfficherCodeErrone()</code>	Affiche un message précisant que le code fourni est faux.
<code>getCode()</code>	Récupère le code saisi par l'utilisateur. Cette fonction est à coder par vos soins.
<code>controlCode(String code)</code>	Vérifie la validité du code et enclenche l'affichage de la place le cas échéant. Cette fonction est à coder par vos soins.

Structure de la fonction `getCode()`

La structure de la fonction `getCode()` est la suivante.

Une boucle attend la validation de l'utilisateur (appui de la touche V) pour renvoyer le code.

Dans cette boucle, il est attendu l'appui d'une touche, puis, en fonction de la touche appuyée, les actions suivantes sont exécutées :

Touche C : retirer le dernier caractère du code (corriger)

Touche V : renvoie le code saisi par l'utilisateur (valider)

Autre touche (0 à 9) : ajout du chiffre choisi au code à 4 chiffres

Les variables disponibles

NUM_ROWS	Nombre de lignes sur le clavier
NUM_COLS	Nombre de colonnes sur le clavier
RESERVATION_COUNT	Nombre de réservations enregistrées dans le système
RESERVATION_DATA	Un tableau contenant l'ensemble des réservations
keys	Un tableau à 2 dimensions contenant les caractères du clavier
rowsPin	Broche de connexion des lignes du clavier matricé
colPins	Broche de connexion des colonnes du clavier matricé
kp	L'objet représentant le clavier
lcd	L'objet représentant l'écran LCD

Les bibliothèques disponibles

L'objet String

<code>String message = String("Coucou")</code>	Créer un objet String contenant le message coucou
<code>message.length()</code>	Renvoie la longueur de la String (6 dans ce cas)
<code>message.concat("un truc")</code>	Ajoute "un truc" à la fin du message existant
<code>message.remove(index)</code>	Retire le caractère n°index du message

La librairie Keypad

<pre>Keypad kp = Keypad(makeKeymap(keys), rowPins, colPins, NUM_ROWS, NUM_COLS);</pre>	Initialise une variable (kp) représentant le clavier matricé. Il prend en paramètre la liste des touches, les pins des lignes et des colonnes, le nombre de lignes et de colonnes.
<code>kp.getKey()</code>	Récupère la touche appuyée. Renvoie <code>NO_KEY</code> en l'absence de touche appuyée.