**Descriptif du système**

L’objectif de ce projet est de concevoir un système de gestion de parking à l’aide du **pic16F877.**

**Composants**

* Pic16F877
* Deux capteurs de présence
* Un afficheur LCD
* Deux leds : rouge et verte
* Capteur de température LM35
* Servomoteur+ L293D
* buzzer
* Une mémoire EEPROM interne

**Rôle**

1. Le système est muni de deux capteurs de présence, l’un à l’entrée et l’autre à la sortie.
2. Le système est, aussi, muni de deux barrières, l’une à l’entrée et l’autre à la sortie. Il est à noter qu’il n’est possible de quitter le parking sans avoir effectué de paiement.
3. Les deux barrières sont commandées par un **servomoteur**, chacune. Les servomoteurs sont contrôlés par le microcontrôleur via un circuit L293D.
4. Pour donner une indication rapide de la disponibilité de places, le système possède deux lampes : une verte et une autre rouge.
5. Le système comptabilise le nombre de places disponibles. Ce nombre est sauvegardé en mémoire EEPROM. En cas de démarrage ou de réinitialisation du système, cette valeur est lue depuis la mémoire EEPROM.
6. Grâce à un capteur de température de type LM35, le système peut mesurer la température à l’intérieur du parking.
7. Enfin, le système possède un écran LCD montré pour afficher les informations suivantes :
   1. Le nombre de places disponibles.
   2. La température ambiante (en °C).
   3. Les messages « Bienvenue » et « Au revoir ».

**Fonctionnement**

Principalement, l’afficheur LCD affiche « Bienvenue », puis la température grâce au capteur LM35 et le nombre de place libre qui est égal à « 9 ».

Le système dispose quatre capteurs de présences **Capt1**, **Capt2** , **Capt 3** et **Capt4** qui permettent de détecter l’entrée ainsi que la sortie d’une voiture selon le principe suivant:

* **A l’entrée :**
* **Capt1** permet de détecter la présence d’une voiture à l’entrée pour rendre la barrière à l’entrée « **barrière1 »** levée.
* **Capt 2** permet de s’assurer le passage de la voiture avant la désactivation de la barrière « **barrière1 »**.
* Si **Capt1** a détecté la présence d’une voiture la barrière « **barrière1 »** sera levée (servomoteur 1 dans le sens de l’aiguille d’une montre), cela permet l’entrée de la voiture.
* Uune fois l’entrée de cette dernière est validée, le test suivant sera effectué : **Dès** que **Capt2** détecte le passage de la voiture, les actions suivantes seront engendrées :
* la barrière « **barrière1 »** sera fermée (le servomoteur 1 aura le sens inverse de l’aiguille d’une montre).
* Décrémentation de nombre de place libre qui sera affiché par l’afficheur LCD.
* Une led rouge s’allume.
* **A la sortie :**
* **Capt3** permet de détecter la sortie d’une voiture pour rendre la barrière à la sortie « **barrière2 »** levée.
* **Capt 4** permet de s’assurer du passage de la voiture avant la désactivation de la barrière « **barrière2 »** .
* Si **Capt3** détecte la présence d’une voiture et le paiement a été effectué (bouton appuyé),la barrière « **barrière2»** sera levée (servomoteur 2 dans le sens de l’aiguille d’une montre), cela permet la sortie de la voiture.
* Un buzzer qui sera déclenché si le paiement est effectué et **Capt4** n’a pas détecté la sortie de la voiture après un délai de 10s. Ce délai sera calculé en utilisant le timer0. Notre système fonctionne à une fréquence oscillateur égale à 4 MHz.
* Dès que **Capt4** a détectéle passage de la voiture, les actions suivantes sont engendrées :
* La barrière sera fermée (le servomoteur 2 aura le sens inverse de l’aiguille d’une montre).
* Incrémentation de nombre de place libre qui sera affiché par l’affihceur LCD.
* L’afficheur LCD affihce le message « Aurevoir »
* Une led verte s’allume

**Travail de groupe demandé :**

1. **Conception :** La première étape du mini-projet consiste à :

• Dégager les entrées sorties du système ainsi que leurs caractéristiques techniques.

• Dresser l’organigramme global du système.

• Pour désigner les exceptions résultant d'un événement externe :

o Préciser les sources d’interruptions nécessaires pour la réalisation de votre système.

o Dresser l’organigramme des interruptions.

1. **Montage sur ISIS :** Une fois la liste des périphériques établis ainsi que la spécificité de chacun assimilée, la deuxième phase consiste à dresser le montage ISIS qui placera chaque composant dans les pins appropriés du PIC 16F877.
2. **Le code C :**

• Ecrire en langage C le code relatif à chaque fonctionnalité en précisant les composants internes à utiliser au sein du PIC 16F877.

• Intégrer les différentes fonctionnalités afin de répondre aux besoins du système complet.

En effet, l’environnement **MikroC for PIC** offre des **librairies** pour le codage en langage C pour le PIC 16F877 essentiellement pour le traitement de différents types de périphériques notemment : LCD library, ADC library … avec la liste des instructions spécifiques  
nécessaires, ainsi qu’une **documentation riche** en exemples [Menu **HELP**]

La soutenance est le samedi 29/06/2024 :

Vous devez préparer :

* une présentation
* un montage ISIS complet
* un code C complet
* un mini rapport qui répond au besoin de la conception.