

Parking Polytech

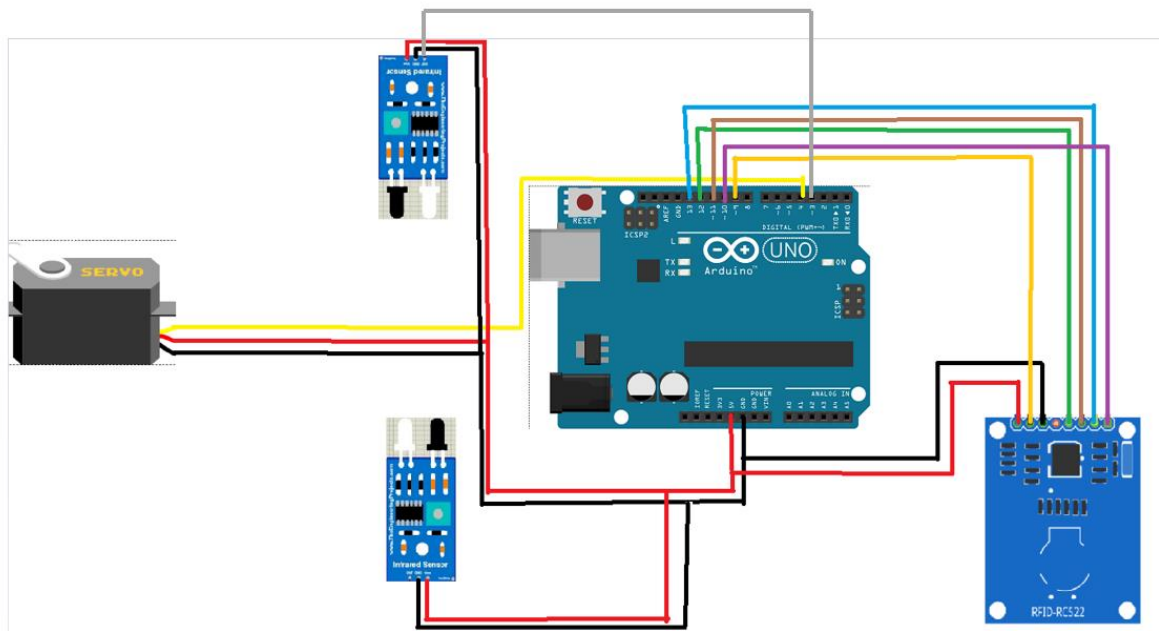
Introduction

Pour notre projet Arduino de PeiP2, nous avons décidé de créer un parking. Il s'agit donc d'un parking pour particulier avec un accès par badge. La voiture doit entrer manuellement dans le parking et tout d'abord s'identifier avec un badge qui va donc déclencher ou non l'actionnement de la barrière. En effet, si la voiture n'est pas autorisée à entrer dans le parking, la barrière ne s'abaisse pas. Les places sont munies de LED allumée en vert si la place est disponible, ou en rouge si la place est occupée. À la sortie, la voiture doit passer par une nouvelle barrière et un écran LCD affiche un message d'au revoir à l'utilisateur.

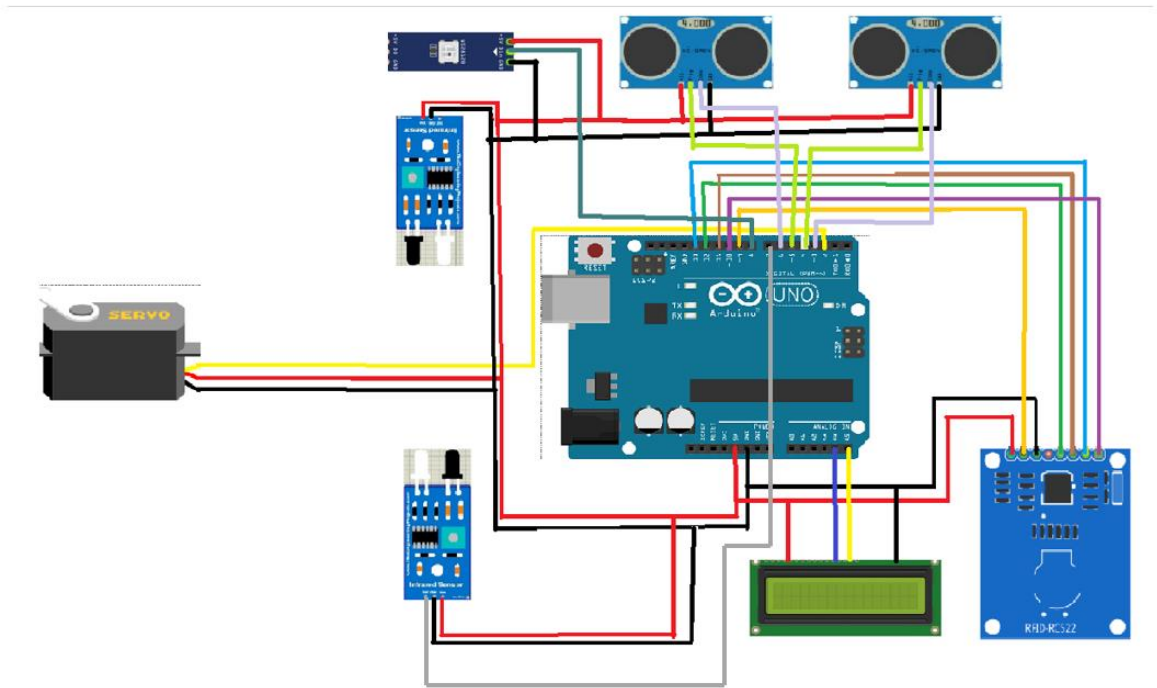
Schémas électriques

Pour notre Parking, nous avons utilisé deux cartes ARDUINO, puisque nous n'avions pas assez de PIN sur une seule carte. Nous disposons ainsi de deux codes différents et par conséquent, deux schémas électriques différents.

Le premier schéma électrique montre le branchement de la barrière de sortie et des éléments nécessaires à son ouverture (module RFID et les capteurs de proximité infrarouge).



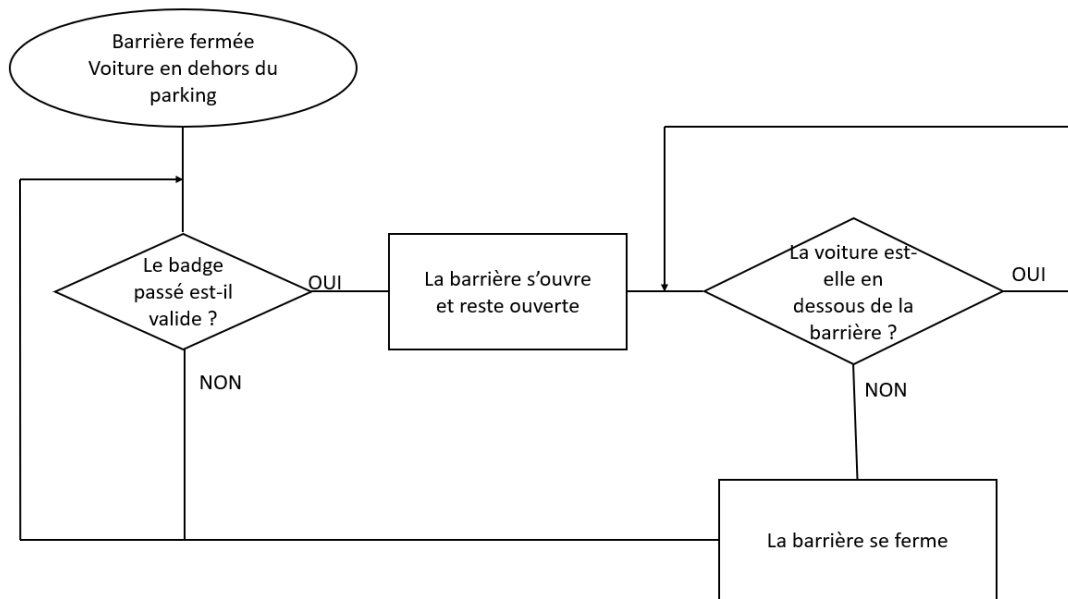
Le deuxième schéma, nous montre le branchement de la barrière de l'entrée avec, comme précédemment, les éléments nécessaires à son ouverture. Nous avons aussi sur ce schéma l'écran LCD, les LEDs (allumées en vert ou rouge) pour les places du parking et les deux capteurs ultrasons.



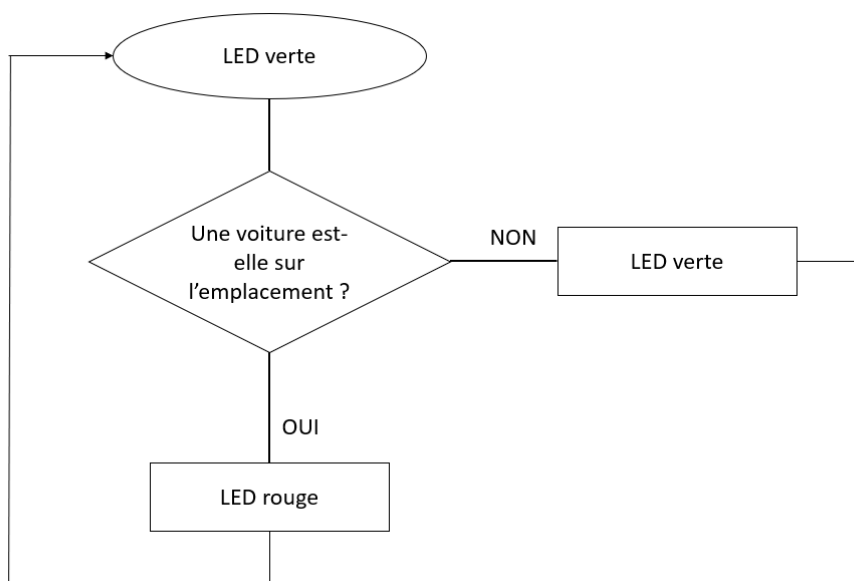
Algorithmes de fonctionnement

Nous avons trois algorithmes de fonctionnement, car comme précédemment, les trois programmes fonctionnent de façon indépendante.

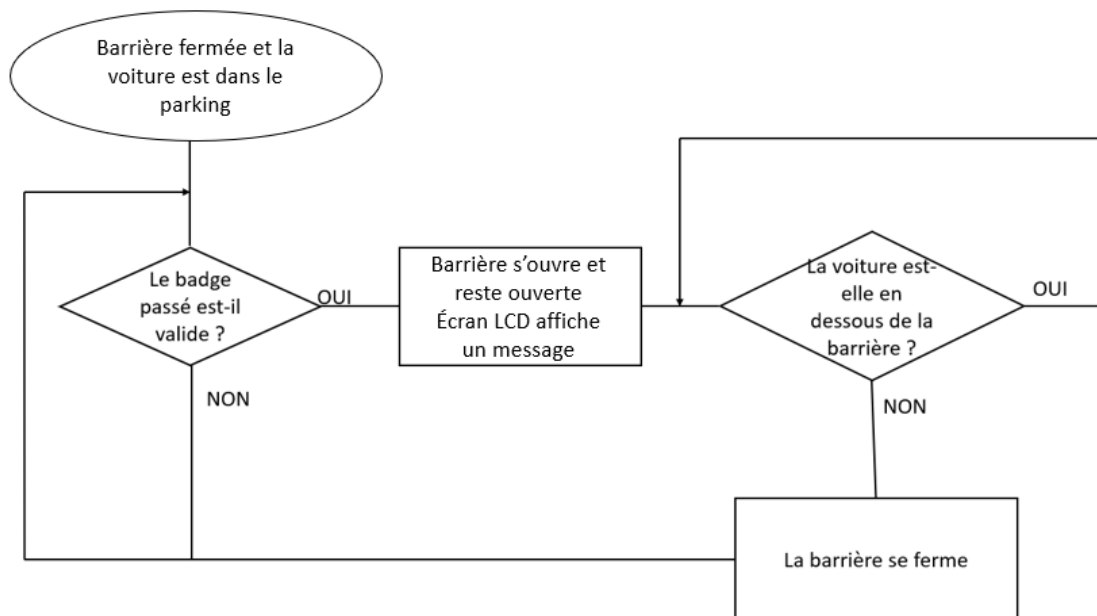
Le premier gère l'entrée du parking avec l'ouverture de la barrière et les conditions nécessaires à leur fermeture.



Le deuxième algorithme de fonctionnement, présente l'organisation du code lié aux LEDs et aux capteurs d'ultrason associés aux places.



Le troisième algorithme gère la sortie du parking avec l'ouverture de la barrière et les conditions nécessaires à leur fermeture ainsi que l'affichage d'un message sur l'écran LCD.



Coût du projet

Le coût de notre projet, est d'environ 825 euros. Une majeure partie de ce budget est constituée par la main-d'œuvre (700 euros). Le coût du travail a été calculé en salaire brut. Nous avons travaillé durant les heures de cours mais aussi dix heures en dehors des cours.

D'autre part, nous avons fait fixer les prix de chaque élément utilisé en se basant sur la moyenne des tarifs affichés sur différents sites de vente.

matériels	unité	prix (€)	total
capteur ultrason	2	4,5	9
capteur proximité	4	4	16
servomoteur	2	4	8
capteur rfid	2	7	14
carte arduino	2	28	56
plaque arduino	1	4	4
reuban led	2	2	4
bois/peinture	-	5	5
travail	34heures	709,92 brut (20,88*34)	709,92
total			825,92

Plannings (initial et final)

Finalement, le planning n'a pas réellement été respecté à cause d'imprévus et de problèmes rencontrés tout au long du projet. Nous nous sommes ainsi rendu compte, en avançant dans le projet, que certaines étapes devaient être faite avant d'autre.

Par exemple, nous avions prévu de faire la maquette au début, mais avec les changements de matériel, il nous a fallu attendre les dernières séances pour prendre les dimensions de chacun des éléments et faire la maquette finale.

• Planning initial

Séances	Quentin	Chloé
Séance 1	maquette dimensions / matériaux	matériels nécessaire (moteur, capteur)
Séance 2	mise en places barrières (moteur pour lever barrière)	capteurs de voitures + leds
Séance 3	capteur présence pour la barrière	badge voiture pour identifier les voitures
Séance 4	capteur barrière pour ouvrir la barrière si détection badge	badge voiture pour identifier les voitures + capteur barrière lire le badge
Séance 5	capteur barrière pour ouvrir la barrière si détection badge	code pour connaître le temps de la voiture dans le P
Séance 6	écran LCD	création pièces + début du montage
Séance 7	assemblage des "gros travaux"	assemblage des pièces (capteurs..)
Séance 8	Vérification des codes/ montages/derniers tests	
Séance 9	Présentation projet	

• Planning final

Séances	Quentin	Chloé
Séance 1	prise en main du matériels nécessaires (moteurs, capteurs, LEDs)	
Séance 2	code pour lier le capteur ultrason à la barrière (servomoteur)	code pour lier capteurs ultrason pour les places avec leds
Séance 3	design de la boîte	prise en main du module rfid pour les badges
Séance 4	code pour afficher un message de sortie sur écran LCD	code rfid pour seulement reconnaître les badges enregistrés
Séance 5	remplacer l'ultrason par le capteur de proximité infrarouge	relier tous les codes entre eux
Séance 6	design des éléments de la maquette sur Inkscape	disposition des éléments sur maquette / réglage capteurs infrarouge
Séance 7	trous maquette et peinture	construction des barrières et peinture
Séance 8	Vérification des codes/ montages (branchements)/derniers tests	
Séance 9	Présentation projet	

Problèmes surmontés

Durant notre projet nous avons fait face à différents problèmes et imprévus.

L'un des principaux problèmes était lié au code en lui-même. En les reliant les codes entre-eux, certains `delay()` ont provoqué des erreurs. Nous avons finalement décidé de les supprimer.

Autre problème : les capteurs ultrason. Le fait d'émettre et de réceptionner les deux impulsions au même moment bloquait le code qui suivait ; en effet, les distances obtenues ne correspondaient pas aux distances réelles entre le capteur et l'objet. De ce fait, les leds ne s'allumaient pas correctement. Nous avons donc imposé un délai entre les deux émissions des signaux.

Nous n'avions pas prévu non plus le fait que, lorsque nous voulions découper les différents trous dans notre maquette en bois, beaucoup de personnes seraient présentes elles aussi. Pour ne pas prendre trop de retard durant cette séance, nous avons décidé alors de vérifier le bon fonctionnement des différents éléments du Parking (servomoteur, module RFID).

Un autre problème très handicapant et indépendant de notre volonté a été provoqué par nos ordinateurs qui ne reconnaissaient plus les différents ports. Ce problème a été réglé par différentes mises à niveau proposées par Windows.

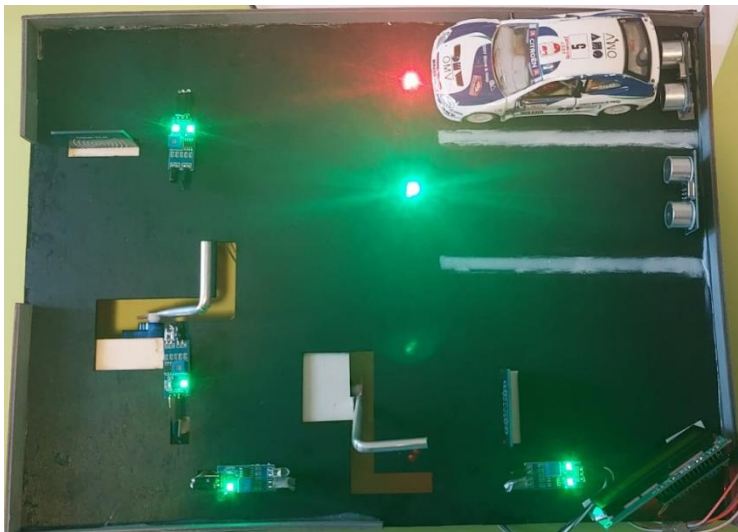
Conclusion

Dans notre Parking, nous avons bien les deux barrières (à l'entrée et à la sortie) qui se s'abaissent lorsque nous passons le badge devant le capteur RFID. Les capteurs de proximité fonctionnent afin d'éviter que les barrières se lève sur la voiture. Enfin, les LEDs s'allument en rouge ou en vert suivant la disponibilité de l'emplacement

Nous voulions relier les deux capteurs RFID entre eux afin de calculer la durée de stationnement de chaque voiture dans le parking. Mais Les nombreuses difficultés rencontrées dans ce projet ne nous ont pas permis de finaliser ce point

Maquette finale

Voici notre maquette finale, avec tous les éléments nécessaires à son bon fonctionnement. Pour l'alimentation nous branchons l'une des cartes ARDUINO à un ordinateur.



Perspectives

Neuf séances supplémentaires nous auraient permis de calculer la durée de stationnement puis de l'afficher sur l'écran LCD de sortie.

Nous avons aussi envisagé de compter le nombre de places restantes avec un autre écran positionné cette fois-ci à l'entrée du Parking, en comptant notamment le nombre de LEDs rouges ou bien le nombre de LEDs vertes restantes.

Nous aurions pu aussi soigner un peu plus le design de notre Parking, et trouver des solutions pour cacher convenablement les éléments utilisés.

Avec des séances en plus, nous aurions aussi mis en place des LEDs pour éclairer le parking avec un détecteur de mouvement.

Bibliographie

Voici la liste des différents liens qui nous ont aidés pour le code mais aussi le branchement des différentes pièces utilisées pour notre projet :

Les deux liens ci-dessous ont été utilisés pour le code mais aussi pour les branchements du module RFID :

<https://mataucarre.fr/index.php/2022/03/03/arduino-et-module-rfid-rc522-gerer-plusieurs-cartes/>

<https://www.electronique-mixte.fr/microcontrolleurs/rfid-controle-dacces-par-badge-avec-arduino/#:~:text=L'application%20consiste%20l'ouverture,alarme%20dans%20le%20cas%20%C3%A9ch%C3%A9ant.>

Le lien suivant nous a servi à relier l'écran LCD à la carte et à avoir un code pour afficher un message :

<https://ledisrupteurdimensionnel.com/arduino/comment-relier-lecran-lcd-a-l-arduino-uno/>
<https://www.e-techno-tutos.com/2021/01/11/arduino-bus-i2c/>

Ci-dessous un site internet avec lequel nous avons appris à nous servir d'un servomoteur :

<https://arduino.developpez.com/tutoriels/arduino-a-l-ecole/?page=projet-12-utiliser-un-servomoteur>

Le lien ci-dessous nous a servi à comprendre le fonctionnement des LEDs pour pouvoir les allumer aux couleurs souhaitées :

<https://www.aranacorp.com/fr/utilisation-dun-ruban-de-led-ws2812b-avec-arduino/>