

PROJET ARDUINO – PeiP2

Année scolaire 2021-2022

LE PARKING AUTONOME

**Étudiants : Louise Lamoureux, Yohan
Journaud**

**Encadrants : Pascal Masson, Monsieur
Peter, Madame Benouakta**

SOMMAIRE

I. Préambule	3
II. Les objectifs : cahier des charges	3
III. Planning	3
IV. Vision globale du projet.....	4
IV.1. Partie électronique	4
IV.2. Partie conception	5
V. Algorithmes.....	7
VI. Perspectives	8
Conclusion	8
Bibliographie.....	9

PRÉAMBULE

Durant notre 2^{ème} année à Polytech, nous avons dû réaliser un projet d'arduino en binôme. L'objectif était de gagner en autonomie et de découvrir en profondeur l'électronique et l'usinage.

Nous avons comme objectif de trouver un sujet et de s'y tenir durant une période de 8 semaines pour arriver à une présentation finale et une démonstration.

CAHIER DES CHARGES

Dans le cadre de notre projet, nous avons souhaité répondre à la problématique de l'accessibilité des places de stationnement en ville, en particulier pour les personnes en situation d'handicap ainsi que les personnes disposant un véhicule électrique.

Notre idée initiale se tournée principalement sur l'arceau car il a la particularité de pouvoir être placé à l'entrée d'un parking ou tout simplement sur une place en ville. Puis elle a évolué jusqu'à l'idée d'un parking autonome qui serait composé de multiples capteurs qui faciliteraient le parage. Nos idées étaient :

- un arceau (avec servomoteur) à l'entrée pour réguler le trafic.
- une télécommande à signal infrarouge pour commander l'arceau.
- des écrans pour transmettre des informations aux utilisateurs.
- un chronomètre pour calculer le temps et le prix du stationnement de chaque utilisateur.
- des LEDs sur chaque place pour afficher la disponibilité.
- des lumières de reculs pour prévenir de la distance avec le mur.
- un son de recul aussi pour prévenir de la distance avec le mur.
- une maquette réaliste.

PLANNING

	Semaine 1	Semaine 2	Semaine 3	Semaine 4
SEMESTRE 1 INITIAL	Fonctionnement général +Prototype 3D + Liste matériel + Dimensions	Début de la maquette + programme de détection du véhicule et de radiofréquence	Programme LED disponibilité + faux sol + montage de l'arceau + synchronisation des programmes	Finalisation + préparation de la présentation
SEMESTRE 1 FINAL	Acquisition du matériel + Programme servomoteur + Dimensions + Prototype 3D	Programme radiofréquence et détection + début de la maquette (trous pour capteurs et LEDs)	Synchronisation des programmes + construction de la boîte de l'arceau + programme LED	Présentation + planche en bois pour la maquette + remplacement LED avec résistance par un bandeau LED
SEMESTRE 2 INITIAL	Code écran LCD nombres de place + chronométrage du temps de stationnement	Tarifs selon temps de stationnement + affichage sur LCD + réalisation arceau de sortie	Codage arceau de sortie (Radiofréquence, Chrono, Fonctionnement) + assemblage des programmes et de la maquette	Finalisation et Powerpoint
SEMESTRE 2 FINAL	Code Bande LED + programme radar allumant LED et tone + nouveaux capteurs de mouvement + trous pour capteurs dans la maquette	Programme chronomètre + réalisation arceau en aluminium + soudure du bandeau LED + limage de la maquette	Programme des différents signaux de la télécommande + mur de la maquette + code arceau de sortie + problème résolu des 2 écrans LCDs	Murs de la case du rangement des fils + pilier soutenant les LCDs + peinture + assemblage des programmes + début Powerpoint

PARTIE ELECTRONIQUE

Afin de sécuriser l'accès à notre parking et d'y laisser entrer que les personnes autorisées, nous avons dû installer un arceau ainsi que différents capteurs à l'entrée et à la sortie. Nos arceaux sont des barres d'aluminium, bloquant le passage, connectés à des servomoteurs. Ceux-ci sont entourés de capteurs de présence permettant de savoir si un véhicule est placé avant ou après l'un des deux arceaux. Nous avons des capteurs à ultrasons initialement, mais par commodité nous avons préféré utilisé des capteurs de présence simple. Nous avons donc quatre états, ECM1 à ECM4 (État des Capteurs de Mouvements).

Pour abaisser un arceau, une télécommande est nécessaire, nous avons donc dû réaliser un programme permettant de savoir si l'utilisateur a utilisé sa télécommande ou non. Pour cela, nous avons recherché le signal en hexadécimal qu'une touche émettait en particulier et nous l'avons mis dans un programme qui permet de faire varier un état ECI (État du Capteur Infrarouge).

A cela s'ajoute la position de l'arceau, si celui-ci est relevé alors `pos_arceau=1` sinon `pos_arceau=0`.

C'est à partir de là que nous avons pu écrire un programme qui permet à l'arceau de se baisser si le capteur infrarouge reçoit le bon signal et si la voiture se situe sur le capteur devant l'arceau. Le servomoteur se relève dès l'instant où le véhicule n'est plus sur un des deux capteurs entourant l'arceau.

Le but principal de notre parking étant de permettre aux usagers de se stationner facilement, nous avons dû concevoir des places équipées de divers capteurs. Tout d'abord, un capteur de mouvement se situe sur chacune des places (ECM5 et ECM6). Ces capteurs permettent de signaler grâce à une LED placée au sol si la place est disponible ou non.

À cela s'ajoute deux capteurs d'ultrasons disposés sur le mur de notre parking. Ces capteurs devaient faire sonner un son ainsi qu'allumer deux LEDs si le véhicule s'approchait trop près du mur. Cependant nous avons fait face à un problème concernant le signal sonore. En effet, lorsque les tones sonnaient, le capteur infrarouge se mettait à détecter des signaux aléatoires, ce qui brouillait donc le signal de notre télécommande. Par manque de temps, nous n'avons pas pu réparer cette panne qui devait surement provenir de la compatibilité timer/tone ou servo/tone. On s'est aussi rendu compte que les branchements devenaient de plus en plus compliqués notamment à cause des résistances nécessaires pour les LEDs, Monsieur Masson nous a alors donné un bandeau LED avec une résistance déjà intégrée dedans.

Afin d'aider les usagers à savoir s'il reste des places ou non, nous avons installé un écran LCD à l'entrée du parking signalant le nombre de place disponible. On connaît ce nombre grâce aux deux capteurs sur les deux places.

Un autre objectif de notre parking était de chronométrer le temps de stationnement d'une voiture ainsi que de l'afficher sur un écran LCD à sa sortie. Pour cela, nous avons utilisé la méthode `millis()` déjà implémentée. Nous avons fait face à un second problème, nous n'arrivions pas à attribuer un chronomètre à une voiture en particulier. Pour faire face à ce problème, nous avons décidé d'utiliser un signal de télécommande pour chaque voiture. Lorsque l'arceau d'entrée se baisse, il démarre le chronomètre de la voiture associée au signal. Enfin, lorsque la voiture actionne sa télécommande pour sortir du parking, le chronomètre s'arrête, et, grâce à quelques lignes de codes, un coût est calculé et est renvoyé dans une variable.

Il reste donc plus qu'à afficher le prix sur un écran LCD positionné en sortie. Cependant, un problème est apparu, comment allé-t-on brancher nos deux écrans LCDs ? Tout d'abord, nous avons trouvés un site nous montrant comment brancher nos écrans avec un module I2C. Une fois le branchement et le code réalisé, rien ne se passe, seul un écran sur deux fonctionne correctement. Après quelques recherches sur le même site, nous trouvons que les modules I2C attribuent une adresse au composant. Nous pensons donc que le deuxième écran ne marche pas car il a la même adresse que le premier. Or, coup de chance, nous voyons que l'adresse I2C est modifiée sur le module grâce à un point de soudure au dos. Il nous reste donc plus qu'à trouver l'adresse de l'écran. Pour cela nous avons testé un programme affichant tous les port I2C connectés et nous avons enfin trouvée l'adresse.

```
-----  
I2C Device Scanner  
-----  
Scanning...  
-----  
I2C device found at address 0x21  !  
I2C device found at address 0x27  !  
----- Done!!! -----
```

Finalement, nous avons injecté cette adresse dans la librairie I2C associée aux écrans, et le programme a fonctionné. Pour conclure cette partie, nous avons assemblé tous les sous programmes ensemble pour obtenir notre code final.

PARTIE CONCEPTION

Pour cette étape de notre projet, nous voulions réaliser une maquette assez réaliste et esthétique mais aussi fonctionnelle. Au départ, nous étions partis sur un prototype plutôt simple qui au fur et à mesure a été amélioré.

Nous avons commencé par une simple planche de bois, qui correspondait à l'échelle que nous souhaitions (une voiture de 10x15 cm environ), sur laquelle nous avons définis deux places de stationnement. Nous y avons rajouté des murs à l'aide de vis sur deux côtés.

Par la suite, nous avons décidé de l'emplacement de chaque capteurs (mouvement, ultrason, led, servomoteur, écran, tones). Nous avons choisi de les intégrer complètement dans la maquette en les cachant sous le sol ou derrière les murs pour un effet réaliste et esthétique. Il a donc fallu percer, forer, limer à l'endroit de chaque capteur.

Au debout, nous pensions placer les servomoteurs sur la planche de bois mais nous nous sommes rendu compte que les placer en-dessous et fabriquer des rainures serait plus efficace. Il a donc fallu fabriquer les arceaux. Nous avons simplement utilisé un tube fin d'aluminium que nous avons coupé avec une scie, tordu avec une pince et enfin aplati pour pouvoir l'insérer sur le servomoteur.

Puis au fur et à mesure que le programme d'arduino se compliquait, les fils augmentaient. Il a fallu trouver une solution pour les discerner et les organiser. Nous avons donc décidé de faire

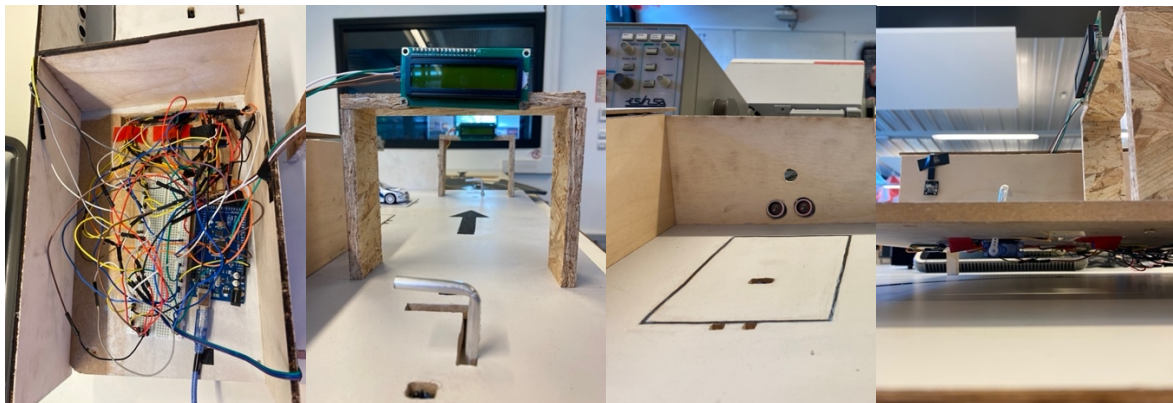
une « cage » où nous avons percé une rainure pour faire passer les fils et ranger la carte arduino. Nous y avons rajouté des murs et organisé les fils en fonction de leur utilisation pour ne pas s'y perdre.

Nous avons aussi soudé les leds entre elles en respectant la distance entre chaque place.

Enfin, nous avons réalisé les piliers qui soutiennent les écrans LCD. Durant cette étape nous avons scié, percé et vissé des bouts de bois entre eux et sur la maquette. Puis nous avons fixé les écrans dessus.

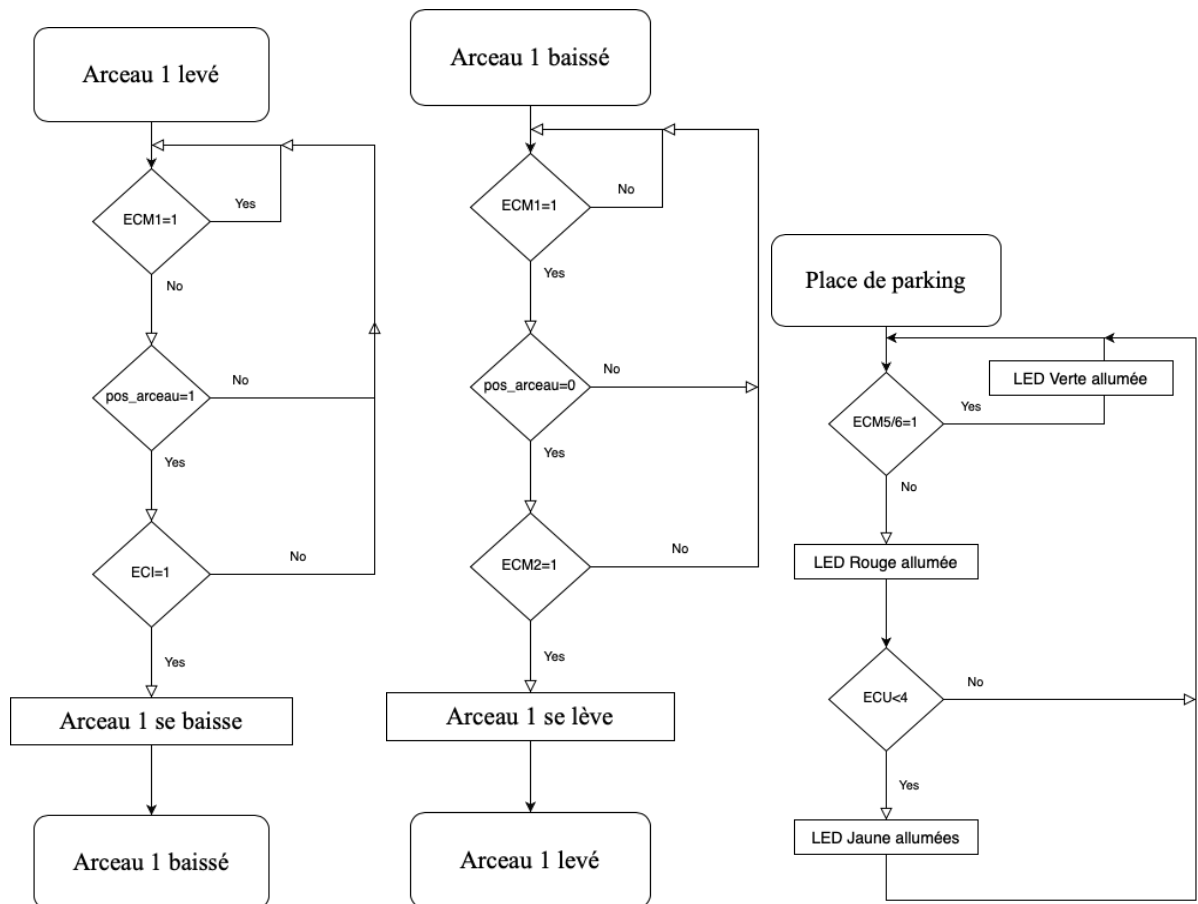
Finalement, nous avons rajouté un petit coup de peinture.

Durant cette partie, étonnamment, nous n'avons pas spécialement rencontré de problème. Nous aurions peut-être souhaité un peu plus de temps pour pouvoir peaufiner des détails comme l'ajout de rampe pour l'entrée et la sortie du parking.



ALGORITHMES

ECM = 1 : Personne ne se situe sur le capteur
 ECM = 0 : Quelqu'un se situe sur le capteur
 ECI = 1 : Le signal de la télécommande est détecté
 ECI = 0 : Le signal de la télécommande n'est pas détecté
 ECU < 4 : Véhicule situé à moins de 4cm du mur
 pos_arceau = 1 : L'arceau est levé
 pos_arceau = 0 : L'arceau est baissé



Algorithme de
l'arceau qui se baisse.

Algorithme de
l'arceau qui se lève.

Algorithme d'une
place de parking.

PERSPECTIVES

Afin d'aider au mieux nos usagers, nous avons pensé à diverses améliorations pour le parking. Tout d'abord, un système de lecteur de plaque du véhicule qui éviterait à l'utilisateur d'avoir et de sortir une télécommande à chaque passage au parking. De plus, nous pourrions attribuer un chronomètre à chaque plaque.

La sécurité du parking est primordiale, c'est pourquoi nous avons pensé à un système de caméras et de capteurs qui détecteraient les potentiels dégradations afin d'assurer le confort et la sérénité de nos usagers.

Plus le temps avance, et plus les véhicules se modernisent, certaines d'entre eux sont quasiment autonome. Pour répondre au mieux à cette évolution, nous avons réfléchi à un parking qui stationnerait automatiquement les voitures. Pour cela l'utilisateur aurait juste à placer son véhicule sur un ascenseur, qui monterait et déplacerait la voiture jusqu'à une place libre grâce à un système de rails.

CONCLUSION

Finalement, ce projet que nous appréhendions d'une part à cause de l'inconnu mais aussi pour la quantité de travail, c'est révélé enrichissant et divertissant.

Il nous apprend à se fixer des objectifs, ce qui au début n'était pas si facile en ne sachant pas à quoi s'attendre. Néanmoins, le fait de faire le point après chaque séance, pour savoir où nous en étions, nous a bien aidé. La découverte de l'usinage avec tous les outils et machines a été très intéressant, nous remercions au passage nos professeurs qui ont su faire preuve de patience et de bons conseils.

Nous avons réussi à finir le projet avec un programme qui fonctionnait (mise à part les tones). Nos objectifs ont été remplis. La présentation finale était fonctionnelle. Nous aurions vraiment aimé prolonger le projet sur un parking complètement autonome.

Avec le recul et l'expérience, nous aurions pu améliorer

BIBLIOGRAPHIE

Bibliothèque pour la télécommande et guide migration version 2.0 vers 3.0 :

<https://github.com/Arduino-IRremote/Arduino-IRremote>

Cours d'arduino du professeur Monsieur Masson :

<http://users.polytech.unice.fr/~pmasson/Enseignement/Arduino%20Projection%20-%20MASSON.pdf>

Guide d'utilisation de télécommande avec fréquence :

<https://www.volta.ma/comment-utiliser-un-recepteur-ir-et-une-telecommande-avec-arduino/arduino/>

Utile pour la connexion de plusieurs écran LCD :

<https://electronics-project-hub.com/how-to-connect-multiple-lcd-to-arduino/>

Utile pour l'utilisation d'un buzzer avec arduino :

<https://arduino-france.site/buzzer-arduino/>

Utile pour le servomoteur :

<https://www.aranacorp.com/fr/pilotez-un-servo-avec-arduino/>