

Rapport de projet Arduino

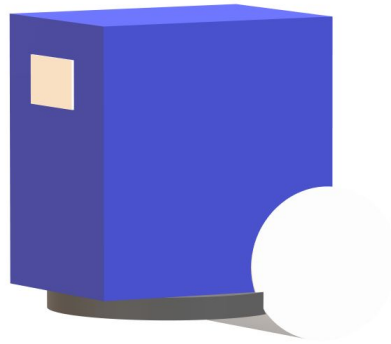
Pique'n Collect

Trieur, mélangeur et distributeur de carte

Guillaume Richerme
Nolwenn Peyratout
G2
2020-2021



SOMMAIRE



Introduction.....	p.2
I. L'idée du projet.....	p.3
II. Les principales idées.....	p.4
1. Ejecteur de carte.....	p.4
2. La bibliothèque et le poussoir	p.7
3. Le plateau tournant.....	p.10
4. Le Bluetooth et l'application.....	p.12
5. Le scanner de carte.....	p.13
6. Vision d'ensemble.....	p.15
Conclusion.....	p.17



Introduction

En tant que joueurs assidus de jeux de cartes, nous nous retrouvons systématiquement face à un problème en début de partie : qui va mélanger ? qui va distribuer ?

Dans ce cas, deux solutions sont possibles : trouver un moyen de se mettre d'accord sur qui mélangera et distribuera.

C'est précisément de là que notre projet tire son origine. Nous voulons créer un outil qui trie, mélange et distribue les cartes.

Trier pour enlever les éventuelles cartes ne servant pas au jeu.

Mélanger plus efficacement que les mélangeurs de cartes du commerce et sans abîmer les cartes.

Distribuer selon le jeu et le nombre de joueurs.

I) L'idée du projet

Avant de nous lancer dans le projet nous nous sommes demandé s'il existait déjà des trieurs, mélangeurs et distributeurs de cartes. Nous avons trouvé un distributeur de cartes, celui du jeu de société UnoExtreme qui projette plusieurs cartes.



Nous avons également trouvé 3 types de mélangeurs de cartes dont les deux premiers sont destinés principalement aux casinos et sont très coûteux.



Enfin, nous avons trouvé [une vidéo](#) correspondant assez fortement à ce que l'on souhaite faire: combiner toutes ces caractéristiques en un objet. Cet ingénieur conçoit une machine qui est capable de trier les cartes pour lui constituer un jeu avantageux, ce qui nous intéresse c'est comment il scanne les cartes et traite l'information. Ensuite, sa machine est capable de mélanger les cartes en les distribuant dans les différentes étagères.

Néanmoins, au fur et à mesure de nos recherches nous nous sommes écartés de nombreux points de ce projet.

De plus, nous ajoutons la fonction de distribution des cartes aux joueurs en utilisant un plateau tournant comme dans [cette vidéo](#).



II) Les principales caractéristiques:

1. Ejecteur de cartes


L'éjecteur de carte correspond à la partie où on place le paquet de carte avant de lancer le Pique'n Collect. Les cartes sont placées depuis l'arrière de la boîte au travers d'une fente. Il s'agit en réalité d'abord d'un boîtier capable de se déplacer verticalement pour ensuite éjecter les cartes une par une dans les étagères. Pour concevoir cette partie, nous nous sommes donc penché séparément sur le déplacement vertical et sur l'éjection des cartes.

1.1. Déplacer l'éjecteur

Piste n°1 : Nous avons d'abord voulu nous inspirer d'un projet d'[éjecteur conçu](#) par des Peip2 de Polytech Angers. Ici, l'éjecteur se déplace grâce à des engrenages et deux rail denté, le tout guidé par deux barres métalliques.

Ce système, tel qu'il apparaît finalement dans les vidéos, ne semble pas très stable. Nous avons donc étudié d'autres pistes.

Piste n°2 : Nous avons voulu reprendre le concept d'un ascenseur tiré par un treuil vers le haut et qui serait donc doucement relâché pour descendre. Pour guider le boîtier, il y aurait deux rails sur les côtés. Cette solution semblait assez simple au niveau du mécanisme avec uniquement un treuil à contrôler mais à vrai dire elle n'est pas idéale. En faisant ainsi, on n'aurait que très peu de contrôle sur l'éjecteur car on ne peut pas pleinement contrôler la descente. De plus, le boîtier de l'éjecteur devrait disposer d'une structure plus complexe avec une partie supérieure et assez haute pour pouvoir être tiré vers le haut et ne pas empêcher les cartes d'aller dans l'éjecteur. Cette "arche" au-dessus du boîtier demanderait aussi à la boîte du Pique'n Collect d'être plus haute et donc moins compacte ([cet exemple](#) de projet d'ascenseur permet de se faire une idée de la place inutile et encombrante engendrée). C'est à dire qu'il faut encore plus de matériel et que l'on perd de la place. Cette piste étant trop contraignante nous l'avons abandonnée.



Piste n°3 : Nous avons alors gardé l'idée du guidage par des rails sur les côtés mais le moteur du déplacement devient un système avec une courroie de distribution inspirée de [ce projet](#). Cette solution nous permet d'enlever toutes les contraintes qu'il y avait avec le système de treuil. Pour comprendre le fonctionnement de ces courroies et leur utilisation nous utilisons [ce site](#) décrivant un projet de fabrication de CoreXY utilisé surtout pour les imprimante 3D. Cette solution nous permet un contrôle précis de la hauteur essentiel pour placer l'éjecteur en face de la bonne étagère et elle nous permet des déplacements rapides. En revanche, nous n'utiliserons pas le même système de rail mais, dans le même esprit que le projet de la piste 1, nous utiliserons des barres en métal pour guider l'éjecteur. Pour le moteur, le Nema17 proposé pour la fabrication du CoreXY a une puissance suffisante pour déplacer tout l'éjecteur.

Matériel :

- Moteur Nema 17
- 2 barres en métal
- Une courroie

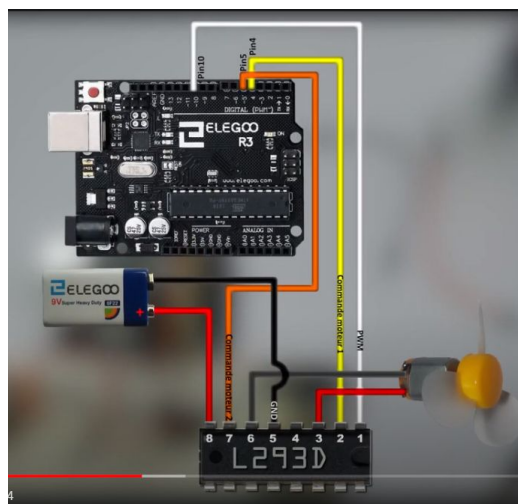
1.2. Éjecter les cartes

Pour éjecter les cartes, on a décidé de s'inspirer du système mis au point dans [cette vidéo](#). Nous souhaitons donc utiliser un rouleau d'entraînement pour pousser les cartes une à une vers les deux rouleaux qui l'éjecteront. Pour faire tourner les rouleaux nous utiliserons un moteur à courant continu. Les rouleaux seront actionnés par un système de courroies.



Néanmoins, en utilisant un seul moteur nous faisons face à une difficulté : les vitesses des 3 rouleaux ne peuvent être les mêmes. D'une part, les deux rouleaux éjectés doivent aller assez vite pour éjecter la carte assez fort dans la bibliothèque. D'autre part, le rouleau d'entraînement, placé sous le paquet doit aller assez lentement afin de n'amener qu'une seule carte vers les rouleaux éjecteurs. Pour résoudre ce problème, la poulie du rouleau d'entraînement sera plus grande que celle des rouleaux d'éjection. De cette manière, quand le rouleau d'entraînement aura fait un tour, les rouleaux d'entraînement en auront fait plus d'un et seront allés plus vite.

Le moteur qui actionnera les rouleaux sera un moteur à courant continu, un tel moteur nous permettra d'avoir une vitesse assez importante. Pour utiliser le moteur à courant continu, on se sert notamment de [cette vidéo](#).



Nous concevrons la structure de l'éjecteur pour l'imprimer car cette pièce est unique et nous n'avons pas trouvé de modèle correspondant assez à notre projet. Cette structure devra pouvoir contenir les 3 rouleaux, un paquet de 54 cartes, le moteur et elle devra aussi comporter une ouverture vers le bas afin que le scanner puisse identifier les cartes durant le processus de tri. Il s'agira finalement d'une structure assez large qui pourra également accueillir le système de déplacement vertical explicité dans la partie précédente.

Matériel :

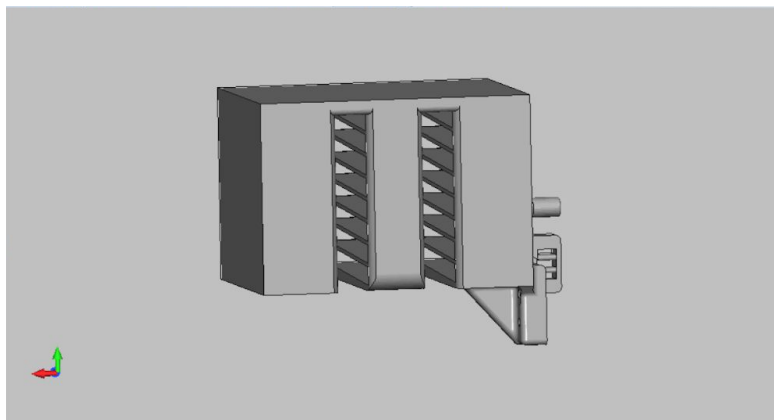
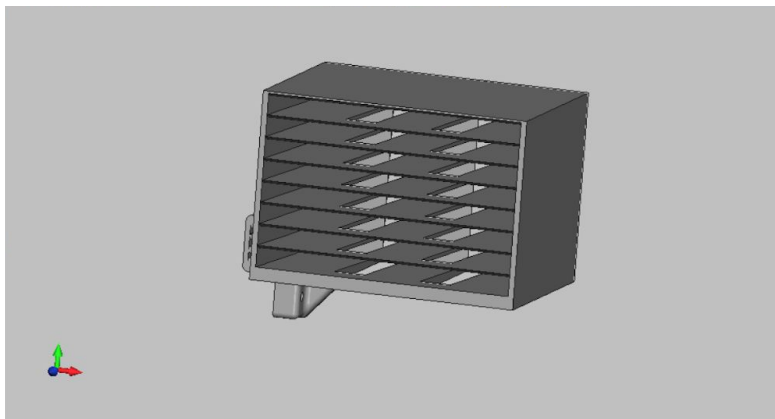
→ Moteur à courant continu

- Circuit intégré L293D
- Module d'alimentation DC

2. La bibliothèque, le poussoir

2.1. La bibliothèque

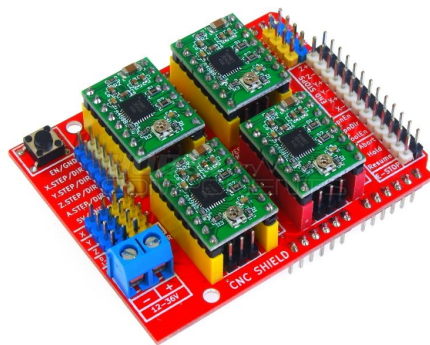
La bibliothèque est constituée de 4 étagères qui devront accueillir chacune un quart du paquet lors du mélange. Ainsi, l'avant de la bibliothèque est totalement ouvert pour pouvoir y insérer les cartes. A l'arrière, une encoche permet de pousser les cartes afin de les faire retomber dans l'éjecteur, elles pourront alors être distribuées. Le matériau dans lequel on peut faire la bibliothèque importe peu mais on privilégiera un matériau léger comme du bois ou du plastique.

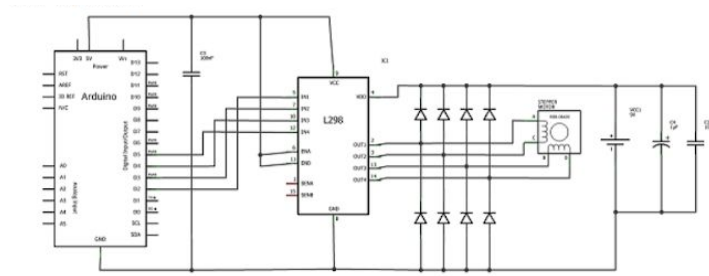


Ces images sont les modèles provenant du projet vu dans [cette vidéo](#). La bibliothèque ressemble en partie à ce que l'on souhaite faire mais il y a quand même plusieurs différences. Nous n'avons besoin que de 4 étagères et non de 8 comme on peut le voir ci-dessus. Ensuite, notre bibliothèque est immobile donc les éléments que l'on retrouve dans la partie inférieure gauche de la pièce ne seront pas présents dans notre bibliothèque. Enfin, notre système pour pousser les cartes se trouve non pas en dessous mais au-dessus de la bibliothèque. Ainsi, l'encoche se prolongera également sur la partie supérieure de la bibliothèque. Nous utiliserons aussi, qu'une seule encoche plus large au lieu de deux.

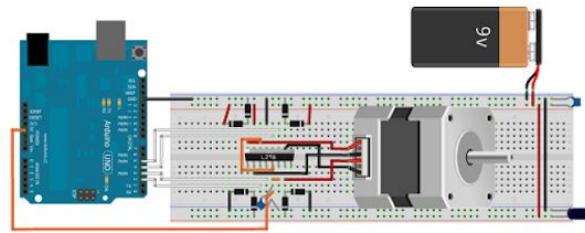
2.2. Le poussoir

Le poussoir est une barre constituée de plusieurs branches qui en avançant poussent les cartes dans l'éjecteur. Ces branches sont de tailles différentes, elles sont de tailles croissantes afin de ne pas faire tomber les cartes en même temps mais paquet après paquet. Pour faire avancer le poussoir, il est fixé à une tige filetée motorisée. Le fonctionnement s'inspire des CNC comme on peut le voir sur [cette vidéo](#). Nous souhaitons utiliser en partie le même matériel (moteur Nema 17, une tige filetée de 8 mm, un accouplement d'arbre et un accouplement à bride rigide). Pour empêcher le poussoir de tourner avec la tige filetée, il est guidé par une tige en métal dans le même esprit que [cette vidéo](#). Pour contrôler le moteur on utilise un driver L298N :





Câblage du moteur bipolaire - Schéma



Câblage du moteur bipolaire - Montage

Pour le montage on se basera sur le principe de ce schéma trouvé [ici](#) avec les moteurs bipolaires. On aura donc besoin de se relier à une source externe en 12V, et un module L298N pour [contrôler](#) le moteur.

Pour la partie code, nous utilisons cette [page](#), ainsi que [celle-ci](#). On devra ainsi, pour utiliser notre moteur pas à pas, utiliser la librairie [Stepper](#). Elle nous permettra de contrôler notre moteur pas à pas pour ne pas faire trop avancer le poussoir dans la bibliothèque. Comme avec cet exemple :

```
#include <Stepper.h>

// pour un moteur de 200 pas par tour et brancher sur les broches 2, 3, 4, 5
Stepper moteur(200, 2, 3, 4, 5);

void setup()
{
  moteur.setSpeed(30); // 30 tours par minute
  // (rappel : ici le moteur fait 200 pas par tour,
  // on fera donc 6000 pas par minute)
}

void loop()
{
  moteur.step(1000);
  delay(100);
  moteur.step(-1000);
  delay(2000);
}
```

Matériel :

- Moteur pas-à-pas Nema 17
- Module L298N

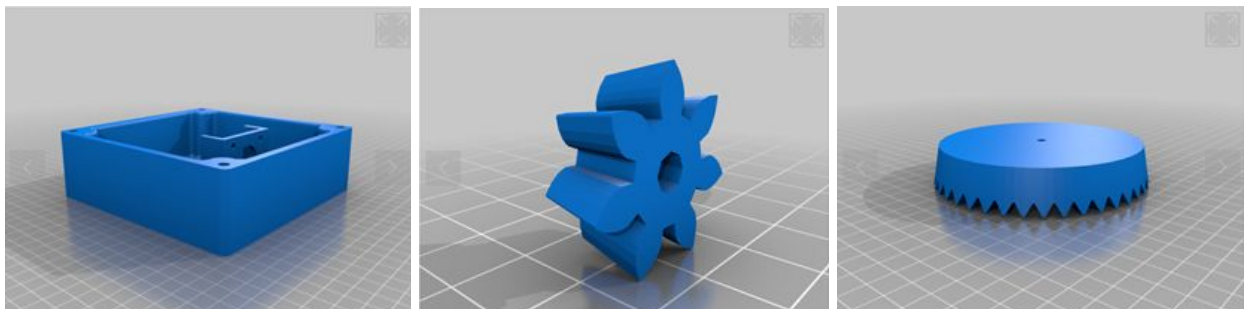
- Tige filetée de 8 mm
- Accouplement d'arbre
- Accouplement à bride rigide

3. Le plateau tournant:

Nous souhaitons que le Pique'n Collect puisse distribuer les cartes à chaque personne, il doit donc se déplacer pour atteindre chaque personne. Lorsque l'on joue à un jeu de carte entre amis, nous sommes, dans la majorité des cas, autour d'une table, le Pique'n Collect pourrait donc être au milieu de celle-ci et tourner sur lui-même afin de distribuer les cartes à chaque personne.

Ainsi on se base sur cette [vidéo](#) pour aborder l'idée de notre plateau tournant. Mais en développant les recherches sur le moteur utilisé dans la vidéo, on s'est rendu compte que le moteur n'est pas assez puissant pour supporter une charge importante. On a donc trouvé ce [modèle](#) 3D qui utilise un moteur plus puissant afin d'être sûr que le Pique'n Collect tourne efficacement pour distribuer les cartes.

On devra donc créer via une imprimante 3D, le socle pour implémenter le moteur pas à pas Nema 17, ainsi qu'un engrenage et un plateau, où sera disposé notre Pique'n Collect, lié au socle via l'engrenage. Il faudra que le Pique'n Collect soit bien fixé au plateau pour qu'il ne tombe pas du plateau tournant.



Pour la partie code sur Arduino ainsi que pour le circuit nous utilisons les mêmes ressources que celles données précédemment pour le poussoir. Il faudra pour la partie code, réarranger le code présenté avant, avec d'autres méthodes de la librairie Stepper,

afin de ne faire tourner le moteur que d'un certain nombre de pas précis en fonction du nombre de joueurs.



Moteur pas-à-pas Nema17

Matériel:

- [moteur](#) pas à pas Nema 17
- [module](#) L298N
- socle et support en 3D
- vis et écrou pour solidifier le socle et le moteur ainsi que pour relier le support avec la boîte.

4. Le Bluetooth, l'application:

Afin de contrôler notre Pique'n Collect nous souhaitons que les joueurs puissent faire le choix du type de jeu qu'ils souhaitent. Pour ce faire nous ne savions pas si nous voulions utiliser la technologie Bluetooth ou un écran LCD avec des boutons pour leur permettre de faire leur choix.

Nous avons éliminé la piste de l'écran LCD car celle-ci est moins confortable pour les joueurs. En effet, avoir une liste de plus de 10 types de jeux de cartes n'est pas facile à lire sur un écran LCD. Mais aussi pour un mode manuel, si le type de jeu voulu par les joueurs n'est pas dans la liste, devoir lire les instructions sur un petit écran n'est pas très pratique.

Ainsi nous avons choisi la solution du Bluetooth. La liaison entre l'arduino et le téléphone se fait via une boîte de dialogue, où l'arduino envoie une information et la

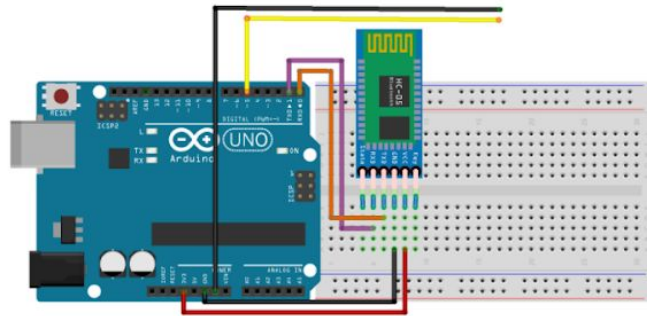
personne sur son téléphone y répond. Pour rendre la liaison entre l'arduino et le téléphone plus agréable pour le joueur, nous allons créer une application comme le propose cette [vidéo](#). Pour se faire il va falloir utiliser MIT App Inventor, cela nous permettra de créer l'application android facilement et avoir une liaison bluetooth avec notre arduino.



Il n'y a pas de code à proprement parler, seulement des blocs que l'on relie les uns aux autres afin de mettre en place les actions que l'on peut faire via l'application. Ainsi les commandes permettront de renvoyer à la carte Arduino des informations sur le nombre de cartes total, le nombre de joueurs ainsi que le nombre de cartes par joueur afin que l'éjecteur et le plateau tournant puissent s'actionner correctement.

Pour la partie code sur Arduino, il y a une librairie pour les modules bluetooth : [SoftwareSerial](#). Sur le code de ce [site](#), ils ne l'utilisent pas mais utilisent les sorties 0 et 1 de la carte Arduino. Alors que sur ce [site](#) on nous explique qu'il faut l'utiliser pour établir les voies d'entrée et sorties de notre module. De plus, cette librairie fonctionne sur le même principe que Serial, ce qui simplifie son utilisation.

Pour la partie branchement on se basera sur [cette page](#), comme cela est montré ci dessous (sans le fil jaune et le deuxième fil noir).



Pour se faire on aura besoin d'un module HC-05 que l'on reliera à notre carte arduino afin de contrôler les retours de l'application.

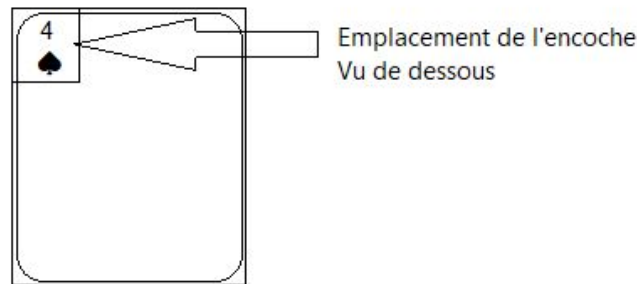
Matériel:

- un module HC-05
- Un téléphone android

5. Le scanner de carte:

Dans notre idée, le Pique'n Collect ne fait pas que mélanger et distribuer des cartes, il peut aussi les trier. C'est à dire que le joueur donne au Pique'n Collect un jeu de 54 cartes et en fonction du jeu choisi, la machine enlève les cartes inutiles, comme par exemple au tarot où il faut enlever toutes les cartes de 2 à 6 afin d'avoir un jeu de 32 cartes.

Donc il faut qu'il y ait une caméra qui puisse lire les informations de la carte pour ensuite les transmettre à l'arduino qui doit enfin voir où mettre la carte dans la bibliothèque. Il faut donc que dans la boîte de l'éjecteur il y ait une petite encoche pour pouvoir lire les informations de la cartes comme cela:



Pour cela nous avons étudié plusieurs pistes:

Première piste: On pensait utiliser un capteur Pixy (CMUcam5), or cette caméra n'identifie que les couleurs perçues et très mal les formes, et elle n'en enregistre que 7, ce qui n'est pas assez, comme le montre la [doc](#). Donc nous ne pourrions pas utiliser cette piste.

Deuxième piste: on souhaitait utiliser le même principe que cette [vidéo](#). On prenait une caméra et on enregistrerait chaque image de chaque carte que l'on stocke dans notre code arduino, avant d'utiliser le Pique'n Collect. Puis à chaque fois que la carte était photographiée, on la comparait avec notre stock de photo, et en fonction du résultat on la faisait sortir de la machine ou on la rangeait. Mais comparer les cartes les unes aux autres est assez complexe.

On s'est penché sur une troisième piste. Elle se base sur le principe du code barre. Ainsi, sur chaque carte il y aurait un code barre et dans le Pique'n Collect un scanner. Il faut donc penser à la lumière: en effet le scanner a besoin de lumière pour fonctionner, donc notre Pique'n Collect doit être constitué de plusieurs parois transparentes.

Pour ce qui est du fonctionnement du scanner, il faut appuyer sur un bouton que cela soit comme une gâchette pour les modèles plus classiques ou sur un bouton comme sur celui-ci:



Nous avons trouvé des informations sur cette [page](#), pour la partie montage et code pour utiliser une douchette avec l'arduino.

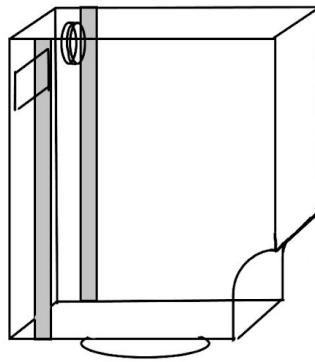
Mais il faudrait plutôt disposer d'un scanner qui scanne en continu afin de ne pas avoir de mécanisme qui appuie sur le bouton/ gâchette à certains moments. Ainsi, nous avons trouvé une autre piste : les modules fixes comme [celui-ci](#). On pourrait ainsi brancher notre module avec les câbles TTL fournis par le constructeur que l'on relierait avec des câbles Dupont femelle-femelle afin de connecter le module à la carte arduino, en s'inspirant de cette [vidéo](#). Comme cela on évite d'ajouter un USB Host Shield à notre carte arduino, qui est plus encombrant.

Matériel:

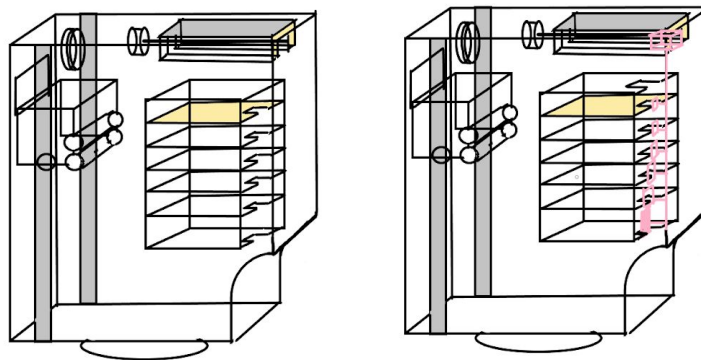
- un scanner fixe comme [celui-ci](#) avec le câble TTL
- câble Dupont femelle-femelle

6. Vision d'ensemble:

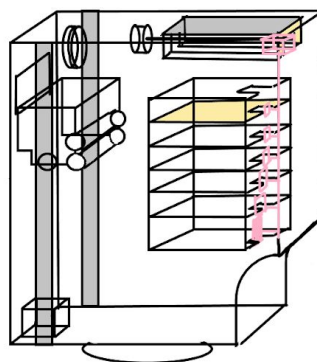
Notre Pique'n Collect sera placé sur le plateau tournant. Le Pique'n Collect à l'allure d'un grand bloc, dans la partie inférieure droite se situe une grande encoche permettant l'éjection des cartes par l'éjecteur vers l'extérieur :



Puis nous y ajoutons l'éjecteur sur la partie de gauche. L'idée finale n'est pas la même que sur le dessin, mais il reste tout de même à la même place, ici nous avons d'abord représenté des rails mais finalement nous utiliserons des barres de guidage. Nous ajoutons en haut à droite le poussoir, qui a subi la même modification que l'éjecteur, ainsi que la bibliothèque. Le deuxième schéma montre les branches du poussoir dessiné en rose.



Sur ce dernier schéma, nous ajoutons l'emplacement pour le scanner en bas à droite.





Conclusion

Malgré nos difficultés pour résoudre tous les problèmes nous avons réussi à faire une bibliographie relevant toutes nos pistes de recherche pour préparer au mieux notre futur projet.

Ainsi nous avons trouvé des solutions viables pour aboutir à une machine compacte. En effet, nous recherchions cela pour conserver un aspect pratique. Notre “croupier polyvalent” pourra ainsi débarrasser les joueurs du tri, du mélange et de la distribution des cartes.

Dans une version ultérieure, nous pourrions intégrer d'autres fonctions comme par exemple une fonction de défausse et de distribution des cartes comme dans le jeu du Uno ou celui du cabot.

