

Gantelet d'assassin commandé par Arduino



Réalisé par : Othmane Bennani

Clara Woodman

Sommaire

1. Introduction	3
2. Inspiration.....	4
3. Premier concept.....	5
4. Spécificités du fonctionnement du bloc « lame ».....	6
5. Spécificités du fonctionnement du bloc « arbalète ».....	9
6. Base du code Arduino.....	12
7. Eléments contrôlés par Arduino.....	13
8. Algorithme.....	15

Introduction

Notre projet consiste à réaliser la « Phantom Blade » des assassins des Templiers en référence à l'univers de Assassin's Creed, en la rendant commandable par Arduino.

Assassin's Creed est une série de jeu vidéo historique d'aventure et d'action où chaque opus se déroule à une certaine époque de l'histoire (Troisième Croisade, la Renaissance, Révolution américaine, ...). Dans ce jeu, les confréries assassins/templiers sont en conflit depuis bien longtemps, et au fil des époques, nous incarnons généralement les assassins pour suivre les tensions générées par ce combat acharné.

Ce qui nous a intéressé dans ce jeu, c'est la lame secrète utilisée par les assassins, qui faisait double fonction : lame tranchante pour éliminer furtivement les cibles au corps à corps, et une arbalète pour tirer des petites fléchettes à distance de sa cible. Evidemment, nous comptons réaliser un gantelet similaire à cette « Phantom Blade », mais automatiser son fonctionnement avec notre carte Arduino.



Inspirations

Dans le jeu, la lame et l'arbalète sont à l'arrière de la paume de la main mais pour notre projet, nous allons séparer ces deux fonctions pour rendre l'arbalète accessible lorsque la paume de la main est dirigée vers le haut, et la lame accessible de l'autre côté.



Une fois que nous avons défini la fonction de notre gantelet Arduino, il nous fallait réaliser un cahier des charges pour lister les différents détails du fonctionnement de ce gantelet.

Deux youtubeurs nous ont inspirés pour l'élaboration de notre projet : [RAWICE511](#) et Colin Furze. Le premier a dédié sa chaîne entièrement à la réalisation et à l'amélioration d'une lame cachée d'*Assassin's Creed* et le second est un inventeur faisant des projets élaborés et complexes se basant sur de la pop culture ou des concepts originaux.

Nous avons réalisé les différents schémas que vous trouverez tout au long de cette bibliographie. Ils représentent nos idées, nos théories et nos adaptations sur les fonctionnements montrés dans les vidéos, qui sont rarement expliqués ou décrits.

Nous avons également implémenté des mécanismes d'objet déjà existant, par exemple pour l'ouverture de l'arbalète nous nous sommes grandement inspirés d'un ouvre-bouteille pour les bouchons en lièges comme ci-dessous.



Cahier des charges :

- Le gantelet a deux modes accessibles avec un seul capteur tactile
- Les deux modes ne peuvent être actifs en même temps
- Un indicateur lumineux signalera le mode à l'utilisateur
- Le gantelet doit s'adapter aux avant-bras des personnes le portant
- L'utilisateur doit être protégé au niveau de la main pour ne pas se blesser (gant protecteur renforcé)
- Le gantelet doit être au maximum automatisé
- Les éléments doivent être les plus légers possibles

Premier concept :

Le gantelet est composé de deux blocs : une lame cachée sur le dessus de l'avant-bras et une arbalète sur le dessous.

Tout se contrôle avec un seul capteur tactile accroché sur une bande passant sur la paume de la main.

Un capteur d'inclinaison placé sur le gantelet à l'intérieur du bras permet de spécifier la commande au boîtier Arduino.

- **Mode arbalète :**

Si le bras est tourné vers le haut (le capteur d'inclinaison envoie le signal « HIGH »), l'utilisateur peut alors ouvrir l'arbalète avec le capteur tactile en maintenant le contact pendant 2 à 3 secondes. Pour charger et tirer avec celle-ci, on appuiera sur le capteur.

Pour refermer l'arbalète, il suffit d'appuyer pendant 2 à 3 secondes à nouveau.

- **Mode lame cachée :**

Si le module Arduino reçoit le signal « LOW » du capteur d'inclinaison, alors la lame peut rentrer et sortir en appuyant sur le capteur tactile.

Plus précisément, en maintenant le contact avec le capteur tactile, la lame reste sortie (le capteur envoie un signal « HIGH » en continu) et en le relâchant (le module Arduino doit recevoir un signal « LOW » pendant plus de 2 secondes) la lame se rétracte.

Pour éviter la confusion dans les ordres donnés par la carte Arduino, l'arbalète et la lame ne peuvent être activées simultanément.

Problèmes/rectifications :

Le capteur d'inclinaison peut envoyer des fausses données : en effet, des gestes brusques ou des vibrations peuvent perturber son fonctionnement. Il faudra donc le coupler à un gyroscope pour assurer le bon fonctionnement du gantelet.

Une LED sera rajoutée pour indiquer à l'utilisateur si le gantelet est en mode lame ou en mode arbalète (de couleur rouge pour le premier et verte pour le second et éventuellement une troisième couleur pour quand l'arbalète est chargée).

Spécificités du fonctionnement du bloc « lame » :

Premières pistes d'exploration :

Voici les vidéos qui nous ont servi d'inspiration :

[liens des vidéos]

<https://www.youtube.com/watch?v=VmvcirebyY4&list=ULarbNxONg88c&index=79>

<https://www.youtube.com/watch?v=n6sGkFsi7QY>

<https://youtu.be/6y1vNsBfAZY>

Ces mécanismes ne sont pas automatisés, il faut donc les adapter pour qu'ils puissent être commandés par la carte Arduino.

1^{ère} idée :

En partant du principe bielle manivelle et coulisse, il y aurait couplé à l'arbre du moteur, un gros engrenage sur lequel est fixé la manivelle elle-même reliée à la coulisse (le support sur lequel est attachée la lame. L'engrenage doit faire un demi-tour pour sortir et rentrer la lame. Pour éviter les frottements, le support fixé à la lame est coincé entre deux railles semblables à ceux utilisés pour les tiroirs.

Avantages :

- Le système sera facile à monter car il n'y a pas beaucoup de pièces et les mécanismes sont simples.

Inconvénients :

- L'engrenage sera volumineux et rendra le système plus fragile, moins esthétique et plus encombrant.
- Le moteur devra être très précis, car il devra fonctionner pendant un laps de temps donné pour effectuer exactement un demi-tour d'engrenage.
- Il y aura également un problème de position initiale, car il faudra connaître la position exacte de l'engrenage.
- Il faudra un système pour bloquer et débloquer la lame.

2^{ème} idée :

Le système s'inspire beaucoup de la première vidéo. C'est le fait de tirer sur la corde qui sera contrôlé par Arduino.

Réalisation du boîtier :

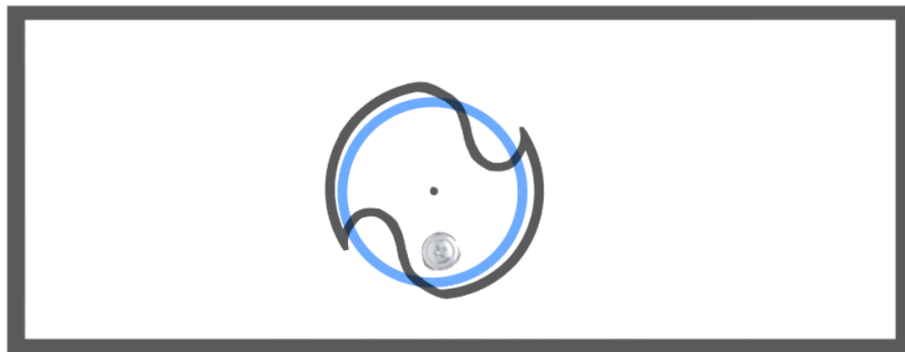


Figure 1 : vue du dessus du boîtier

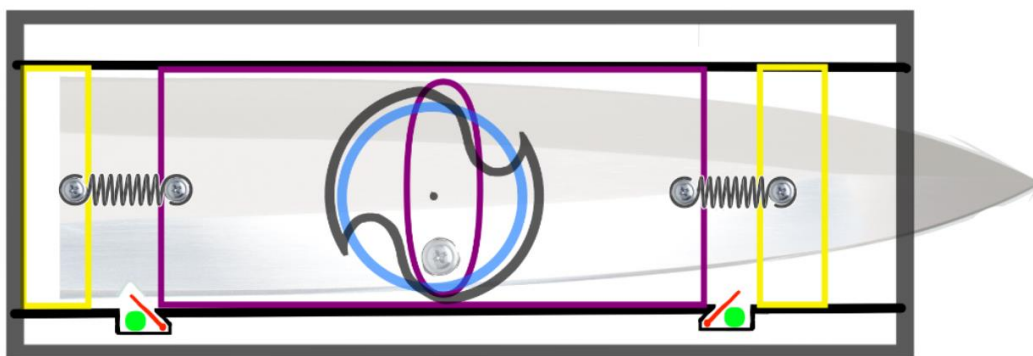


Figure 2 : vue du dessus en transparence du boîtier

Le boîtier est composé de plusieurs principalement de couches de métal formant une cavité dans lequel la lame peut se déplacer librement. Au-dessus de la lame, se trouve deux plaques fixées (en jaune dans la figure 2) sur la plaque supérieure, et une plaque flottante (en violet dans la figure 2) maintenue entre deux ressorts avec une fente ovale dans laquelle se place une vis plate mobile vissée dans la roue (en gris dans la figure 1).

Cette roue peut tourner et la vis fait donc un mouvement circulaire libre dans le trou de la plaque supérieure (en bleu dans la figure 1).

L'effet de cette rotation permet de faire déplacer la plaque libre dans un mouvement de translation (gauche-droite). Enfin la cette plaque poussera un des portillons (en rouge dans la figure 2) en alternant celui de gauche et de droite à chaque demi-tour de roue. Les portillons sont maintenus en position « ouverte » par des éléments spongieux (en vert dans la figure 2) qui après être écrasés peuvent retrouver leur forme initiale.

Ces portillons bloqueront la lame dans les deux cas suivants : « sortie » et « rentrée ».

Implémentation du moteur :

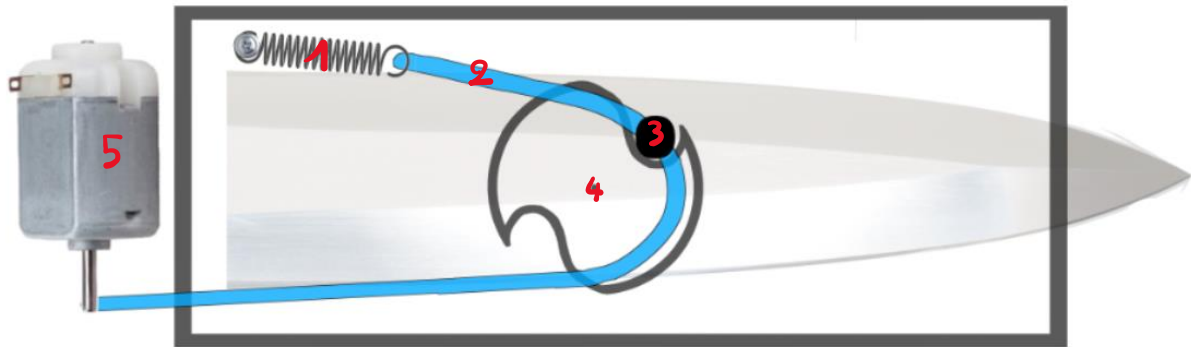


Figure 3 : schéma montrant l'emplacement du moteur par rapport au boîtier en vue du dessus

Le moteur sera placé à l'arrière le boîtier comme sur l'image ci-dessus (élément 5). La corde (élément 2) est fixée à un ressort (élément 1) d'un côté et à l'arbre du moteur de l'autre.

Quand le moteur s'active, la corde vient s'enrouler autour de l'arbre faisant ainsi bouger l'élément 3. Ce dernier se coince dans la forme en hameçon de la roue (élément 4), la faisant tourner.

Par effet de rappel quand le moteur s'arrête, le ressort va tirer sur la corde et faire revenir le système en position initiale (l'arbre du moteur n'opposant presque aucune résistance à tourner dans le sens opposé lorsqu'il est à l'arrêt). La forme en hameçon et l'élément 3 permet de simplifier le système en le faisant tourner que dans un sens et de favoriser le retour à l'état initial.

Spécificités du fonctionnement du bloc « arbalète » :

Premières pistes d'exploration :

Voici les vidéos qui nous ont servi d'inspiration :

[liens des vidéos]

<https://youtu.be/HnekKnBrbMo>

1^{ère} idée pour le déploiement de l'arbalète :

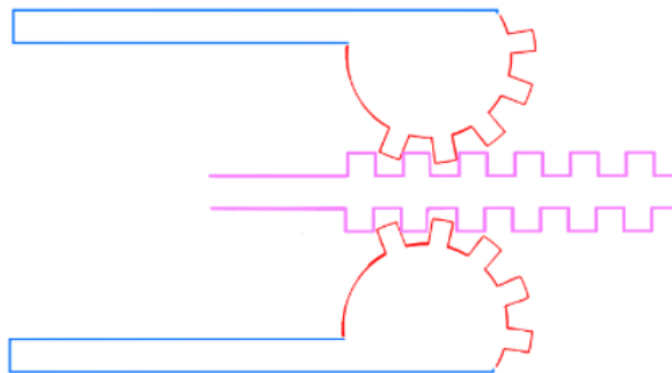


Figure 4 : vue du dessus de la partie centrale supérieure des bras de l'arbalète

Les deux branches (en bleu) se déploient grâce à des engrenage (en rouge) et une barre crantée (en violet). Cette dernière est reliée à un moteur qui en tournant dans un sens fait descendre la barre, ce qui déploiera l'arbalète, et dans l'autre sens fera monter la barre, ce qui rangera l'arbalète.

La corde sera indéformable et ce seront les branches qui se déformeront pour donner plus de détente à la fléchette.

Inconvénients :

- Il n'y a pas de réelle sécurité pour bloquer les branches ouvertes ou fermées.
- L'ouverture sera plus « lente » qu'avec des systèmes de ressorts.

2^{eme} idée :

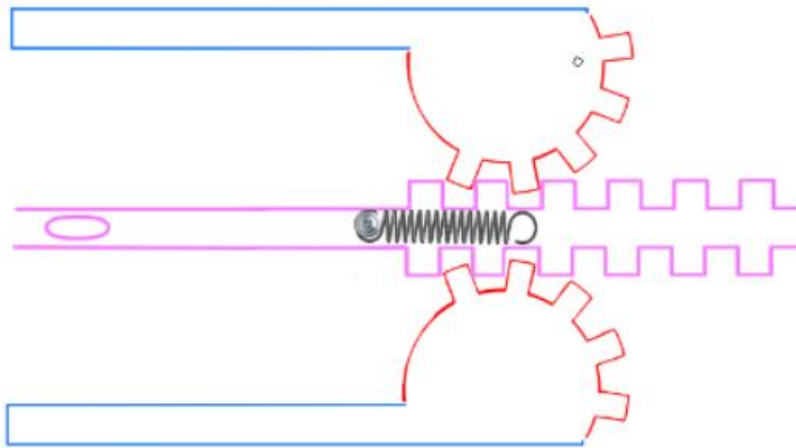


Figure 5 : vue de dessus de la partie centrale supérieure des bras de l'arbalète avec le ressort et la fente pour la gâchette



Figure 6 : vue de profile de la partie inférieure de la barre crantée couplée avec en vue de face l'engrenage fixé à l'arbre moteur

La barre crantée est reliée au support par un ressort. Celui-ci s'étire lorsque l'arbre du moteur, situé en dessous de l'arbalète, tire la barre crantée vers le bas. Une fente dans la partie basse de la barre crantée permet de bloquer la lame avec une gâchette. La gâchette sera certainement commandée manuellement par soucis de surencombrer le gantelet.

Une fois celle-ci baissée, le ressort refermera l'arbalète.

Avantages :

- Il n'y aura que deux états possibles pour l'arbalète (elle ne pourra pas restée semi-ouverte)
- La fermeture gagnera en nervosité
- La gâchette rajoute une sécurité et renforce le système

Idée pour le chargement et le tire de l'arbalète :

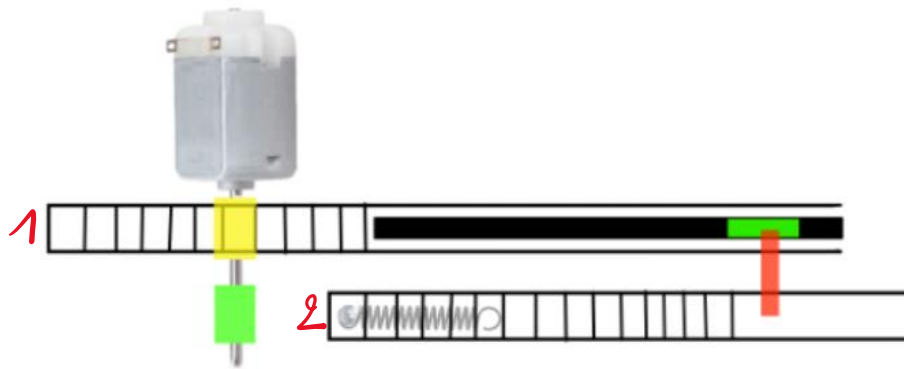


Figure 7 : vue du dessus de la partie inférieure de l'arbalète

Lorsque l'arbalète est ouverte, la barre crantée (1) est bloquée par le système de gâchette vu précédemment.

A ce stade l'engrenage jaune attaché à l'arbre du moteur tournera dans le vide. En effet la section crantée de la barre sera derrière.

Enclenchement du deuxième système : lorsque que le capteur tactile est activé, le moteur se remet à tourner. Le deuxième engrenage (en vert) vient tirer la deuxième barre crantée (2) reliée au support grâce à un ressort.

Dans la première barre crantée, il y a une fente avec un mobile pouvant se déplacer transversalement. Par en dessous, ce dernier est relié à la deuxième barre crantée (élément en rouge), de telle sorte à ce qu'il suive les mouvements de celle-ci.

Lorsque que le ressort est au repos, le mobile est dans sa position la plus haute. La corde est alors prête à être tendue. Quand la deuxième barre crantée est tirée, la corde l'est aussi : l'arbalète est chargée.

Les ressorts s'occupent finalement de tout faire revenir à l'état initial.

Base du code arduino :

```
// if inclinaison d'arbalète==bonne position

    // allumer ledVerte pour le mode arbalette

    // if capteur tactile cliqué pendant 2 à 3 secondes

        //openAr=1

        //ouvrir avec arduino l'arbalette

        // while openAr==1

            //if capteur tactile activé pour pendant moins de 2 secondes

                //arbachargée=1

                //charge l'arbalette avec arduino

                //while arbachargée=1

                    //if capteur tactile activé

                        //tirer avec arduino

                        //arbachargée=0

            //if capteur tactile cliqué pendant 2 à 3 secondes

                //ferme l'arbalette avec arduino

//else

    // allumer ledRouge pour le mode lame

    // if capteur tactile activé

        //lameOut=1

        // while lameOut=1

            capteurTactile=HIGH

            //sortir avec arduino la lame

            //if capteurTactile=LOW pendant plus de 2secondes

                //lameOut=0

                //rétracte avec arduino la lame

}
```

Eléments contrôlés par Arduino

Ne connaissant pas le matériel exact disponible en présentiel, nous n'avons pas de code précis, mais plutôt des pistes d'explorations et des exemples.

Fonctionnalité gyroscope :

<https://circuitdigest.com/microcontroller-projects/mpu6050-gyro-sensor-interfacing-with-arduino#:~:text=MPU6050%20sensor%20has%20many%20functions,convertir%20hardware%20for%20each%20channel>

Le MPU6050 Gyro Sensor est un accéléromètre de précision permettant de donner en temps réel l'inclinaison selon les trois axes. Le site ci-dessus propose d'utiliser d'autres fonctionnalités du module et de les afficher sur un écran. Notre code s'inspirera du leur si l'on choisit ce type de capteur d'inclinaison.

Fonctionnement des 2 moteurs :

https://www.tutorialspoint.com/arduino/arduino_dc_motor.htm

Ce site nous a permis de mieux nous renseigner au sujet des différentes commandes du moteurs, nous aurons évidemment besoin de contrôler la vitesse de ceux-ci, le démarrer et l'arrêter grâce au capteur tactile et choisir le sens de rotation.

Fonctionnalité capteur tactile :

<https://create.arduino.cc/projecthub/arcaegecengiz/how-to-use-a-touch-sensor-81b7f4>

Pour le capteur tactile, le code est assez court et simple à implémenter au code général et le branchement est simple.

Pour le contrôle de la LED, nous allons utiliser un code similaire à celui vu en cours mais cette fois-ci pour une LED RGB.

Finalement la carte Arduino sera placée dans une pochette comme ci-contre :



<https://fr.aliexpress.com/item/32792683834.html>

Elle permettra également de transporter les batteries sans alourdir le gantelet et permettra de transporter les flechettes.

Les fils seront regroupés sous une gaine et se placeront le long du bras pour garder le maximum de mobilité.

