

Rapport de mi-parcours

Projet de Fin d'Etudes

Conception d'un périphérique USB de type « gadget »

Florian Giovannangeli



IMA

Tuteur projet : Xavier REDON

Introduction

Dans le cadre de ma 5^{ème} année en Informatique Microélectronique et Automatique à l'école d'ingénieur Polytech Lille, j'ai l'occasion de travailler sur un Projet de Fin d'Etudes sur l'ensemble du premier semestre. L'objectif de ce projet est la création d'un périphérique USB de type « gadget » ludique à l'aide d'un microcontrôleur. Le gadget est composé d'une partie mécanique utilisant des servomoteurs et d'une carte électronique de contrôle.

Pour ce faire, j'ai choisi de réaliser une catapulte dirigeable de façon rotative avec un servomoteur afin de permettre à l'utilisateur de viser la zone souhaitée, ainsi que d'un bras pour lancer un projectile. L'utilisateur utilise des touches du clavier pour diriger la catapulte. Ces commandes sont envoyées à une carte de contrôle, contenant un microcontrôleur ATmega16u2, via la communication USB, puis la carte traduit ces données et transmet des positions aux servomoteurs à l'aide de sorties PWM (Pulse Width Modulation).

Les principaux objectifs de ce projet sont donc d'utiliser un microcontrôleur reconnu en tant que périphérique USB (USB device) sur un bus USB géré par un contrôleur USB (USB Host), d'échanger des données via ce bus, et de les traduire en commandes pour des actionneurs, ici des servomoteurs.

Ce rapport présente l'état actuel de mon projet à mi-parcours. Vous trouverez dans ce rapport une présentation générale du projet avec le contexte et le cahier des charges. Dans une deuxième partie sont montrées mes avancées actuelles avec notamment les recherches préalables, des premiers tests et les débuts de la réalisation. En dernière partie est exposé le travail qu'il reste à réaliser pour aboutir ce projet, ainsi qu'un planning des étapes à venir.

Contenu

Introduction.....	1
Partie 1 – Présentation du Projet.....	3
1. Contexte	3
2. Cahier des charges :.....	3
a) Partie électrique	3
b) Partie mécanique	4
c) Partie informatique	4
Partie 2 – Présentation du travail effectué	4
1. Recherches préalables.....	4
a) Reconnaissance en périphérique USB.....	4
b) Commande des servo-moteurs.....	5
2. Réalisation de la carte	6
Partie 3 – Travail à effectuer et planning	6
1. Etapes à venir	8
a) Impression de la carte et câblage.....	8
b) Premiers tests de reconnaissance USB	8
c) Programmation du microcontrôleur	8
d) Réalisation de la catapulte et tests finaux	8
2. Planning	8
Conclusion	10

Partie 1 – Présentation du Projet

1. Contexte

Le but de ce projet est de réaliser un gadget USB constitué d'une partie mécanique et d'une carte électronique de contrôle. La carte de contrôle doit permettre au gadget d'être reconnu par l'ordinateur comme un périphérique USB (USB device) sur un bus USB géré par un contrôleur USB (USB Host).

Le premier travail est d'étudier les chips d'un Arduino UNO et plus précisément celui qui gère l'USB : l'ATMega16u2. L'objectif est de le programmer pour le faire apparaître non pas comme un convertisseur USB/Série (ce qui est le cas pour le UNO) mais comme un USB-Gadget de Linux. Une fois l'ATMega16u2 reconnu ainsi par l'ordinateur, le deuxième travail sera de réaliser la carte électronique en rajoutant des fonctions permettant de lui faire gérer quelques servomoteurs par rapport aux commandes reçues du contrôleur USB. Le périphérique doit donc présenter des points d'accès en écriture, par exemple pour commander la rotation de l'objet, mais aussi des points d'accès en lecture, par exemple pour savoir si la rotation est bloquée en fin de course. Pour la version de production, il est demandé de programmer l'ATMega avec *avr-gcc*. Pour finir, le dernier travail sera de réaliser la structure du gadget voulue en contre-plaqué usiné à la découpeuse laser.

2. Cahier des charges :

a) Partie électrique

La partie électrique sera composée d'une carte de contrôle comprenant :

- Un microcontrôleur ATMega16u2
- Un connecteur USB
- Un connecteur d'alimentation DC
- Un connecteur AVR SPI 2x3 broches pour programmer l'ATMega
- Des connecteurs femelles permettant le branchement des servomoteurs
- Un bouton poussoir « Reset »
- Une LED indicatrice de fonctionnement

... ainsi que tous les éléments nécessaires au bon fonctionnement du microcontrôleur et de la carte en général (Quartz, résistances, capacités, régulateurs, diodes etc...).

La carte sera donc alimentée par un câble d'alimentation DC basique 12V/5A comme pour les Arduino UNO. Par la suite, et pour se passer de tout câble d'alimentation pour un gadget, l'alimentation de la carte sera remplacée par une pile 9V. Cela apportera au gadget une meilleure mobilité.

b) Partie mécanique

Le choix qui a été fait comme gadget est une catapulte. Ce gadget comprendra donc 2 servo-moteurs : l'un pour orienter la catapulte et l'autre pour envoyer le projectile. Du côté mécanique, il faudra alors :

- Un socle de base supportant la catapulte, contenant la carte de contrôle, les connexions et le servomoteur 1
- Une plate-forme rotative dirigée par le servomoteur 1 et sur laquelle se trouve le servomoteur 2
- Un bras de projection rattaché au servomoteur 2

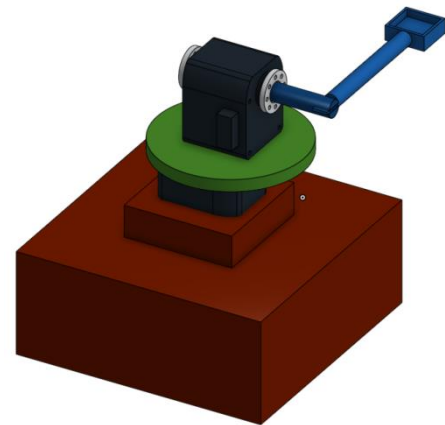


Figure 1 : Schéma subjectif de la catapulte

c) Partie informatique

Le programme informatique doit être écrit en langage C et devra être implémenté sur l'ATMega16u2 avec *avr-gcc* sous Linux. Il devra permettre à la carte :

- D'être reconnue comme un périphérique USB par l'ordinateur
- De recevoir et de transmettre des données à l'utilisateur via le câble USB (appuis clavier, état des servomoteurs etc...)
- De contrôler la position des servomoteurs en fonction des instructions reçues

Le programme devra évidemment aussi être capable d'initialiser l'ATMega16u2.

Partie 2 – Présentation du travail effectué

1. Recherches préalables

a) Reconnaissance en périphérique USB

Le premier travail consistait à tester la reconnaissance d'une carte en périphérique USB. Pour cela j'ai eu accès à 2 Arduino Uno.

La carte Arduino Uno ne communique que par le port série. Une fois installée, elle simule un port série qui permet de lui envoyer un programme. Elle n'est donc pas reconnue comme périphérique USB. Cependant celle-ci est dotée du microcontrôleur ATMEGA328, qui est le cœur même de la carte, mais elle a aussi le microcontrôleur ATMEGA16U2 qui a une contenance de 16Ko de mémoire flash. Ce composant traite les informations reçues par le biais de l'USB et l'ATMEGA328. L'objectif est donc de modifier le programme de l'ATMega16U afin de le faire reconnaître comme périphérique USB.

Remarque : Une fois le programme changé, le port série n'est plus actif, et il devient impossible, dans ce mode, de charger un nouveau programme dans l'ATMEGA328.

Les recherches effectuées m'ont permis de réaliser de premiers tests avec le programme UnoJoy qui permet de faire reconnaître l'Arduino comme un périphérique USB, et ainsi de pouvoir traiter des données échangées avec l'ordinateur via USB. Ici, l'idée est d'utiliser un joystick et de transmettre les actions effectuées sur celui-ci pour l'utiliser par exemple dans un jeu.

En revanche, cette option ne correspond pas au cahier des charges établi car la programmation de l'Arduino a été faite directement en série (télé-versement du programme direct) et non avec un Arduino en programmeur SPI. Mais les méthodes et le code utilisés dans ce programme me permettront de mieux comprendre le transfert vers un périphérique USB, et serviront de base à la programmation de ma carte de contrôle.

D'autres tests ont été effectués avec deux Arduino en programmation SPI et en utilisant Linux avec les commandes *avr-gcc*, mais ces tests n'ont pour le moment pas été concluants.

b) Commande des servomoteurs

Le servomoteur a besoin de trois fils de connexion pour fonctionner. Deux fils servent à son alimentation, le dernier étant celui qui reçoit le signal de commande :

- **rouge** : pour l'alimentation positive (4.5V à 6V en général)
- **noir** ou **marron** : pour la masse (0V)
- **orange, jaune, blanc, ...** : entrée du signal de commande

Pour commander les servomoteurs, il est possible d'utiliser les signaux électroniques de type PWM (Pulse Width Modulation) disponibles sur l'ATMega16u. Deux caractéristiques sont alors indispensables pour que le servomoteur puisse comprendre la consigne : une fréquence fixe de 50Hz (comme celle du réseau électrique EDF) et une durée d'état HAUT, elle aussi fixée à certaines limites, qui permettra d'indiquer les positions souhaitées. A 50Hz, il y a une impulsion toutes les 20ms et les servomoteurs ont des valeurs limites pour l'état HAUT qui sont de 1ms au minimum et au maximum de 2ms pour les servos standards. En fonction de la durée de l'état HAUT, le servomoteur prendra donc un angle bien défini.

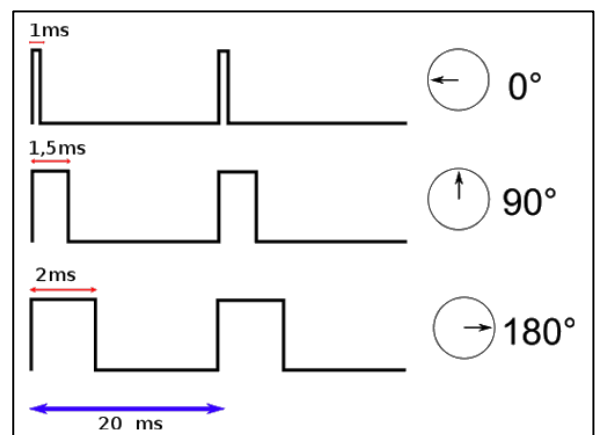


Figure 2 : Angle d'un servomoteur en fonction de la durée d'état HAUT

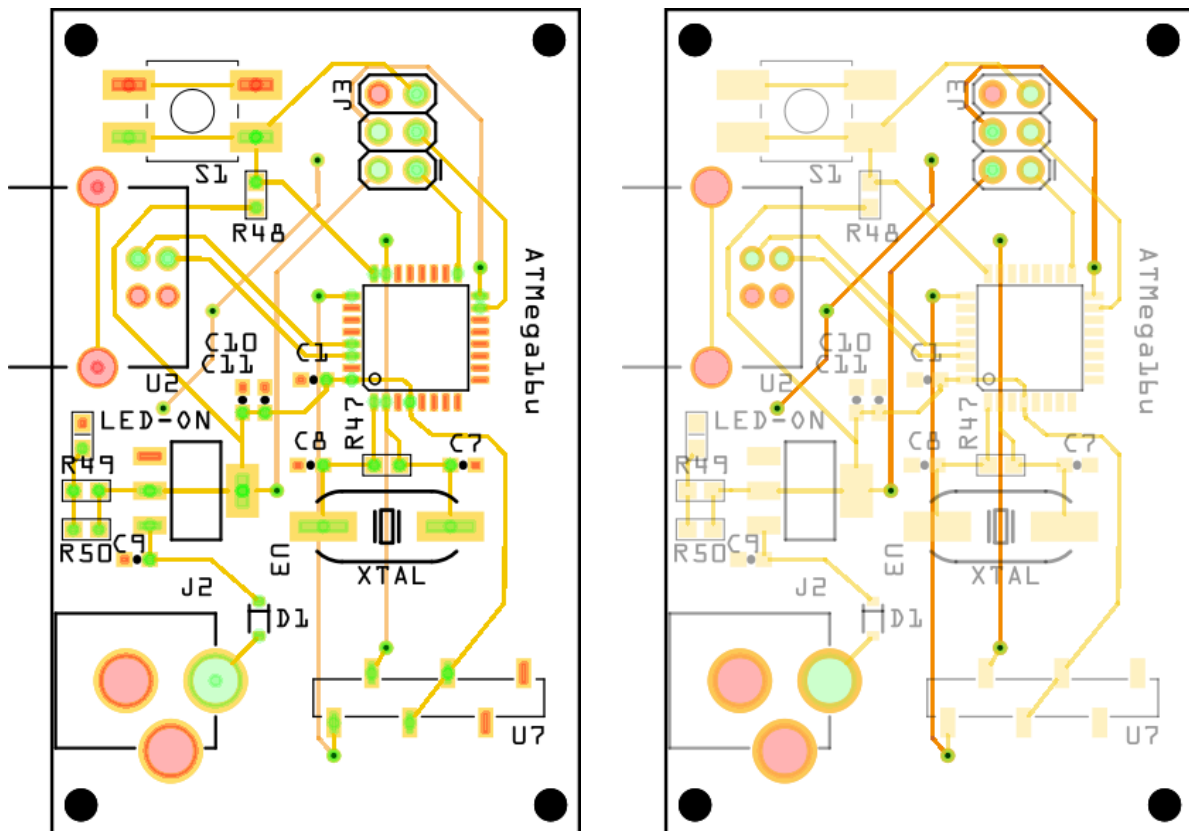
Pour pouvoir interagir pleinement avec le gadget, il faudra récupérer des données qui auront été calculées comme l'arrivée en fin de course des servomoteurs, et il faudra aussi lui envoyer des informations afin qu'il puisse les interpréter : les appuis clavier pour diriger la catapulte.

2. Réalisation de la carte

En parallèle des différentes recherches et des premiers tests sur Arduino, j'ai réalisé un premier prototype de la carte de contrôle sur **Fritzing**. C'est ce premier prototype que je vais pouvoir imprimer et sur lequel je réaliserai les premiers tests généraux.

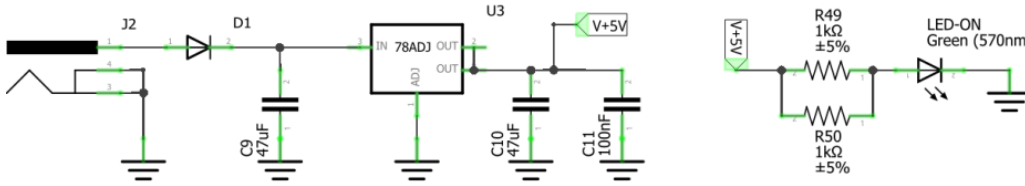
Dans ce prototype nous pouvons donc retrouver l'alimentation avec le connecteur DC 12V/5A dont la tension est ensuite régulée avec un régulateur 5V à découpage (78ADJ) de façon à alimenter le microcontrôleur, les servomoteurs et les autres composants, ainsi qu'une LED rouge indicatrice de fonctionnement. Nous avons la partie microcontrôleur ATmega16u avec ses différentes entrées/sorties et un Quartz de 16MHz. La fonction RESET est contrôlée par un bouton poussoir S1. Et nous retrouvons enfin le connecteur USB avec ses signaux D+ et D- permettant le partage de données, le connecteur ICSP pour pouvoir programmer le microcontrôleur, et le connecteur femelle 6 broches (2x3) qui servira à connecter les deux servomoteurs.

Remarque : Pour une meilleure visibilité du circuit, le PCB ci-dessous ne contient pas le plan de masse, mais il sera évidemment ajouté pour relier tous les pins nécessaires à la masse.

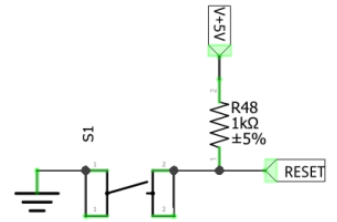


*Images 1 et 2 : PCB du premier prototype de la carte de contrôle
vue supérieure (gauche) et vue inférieure (droite)*

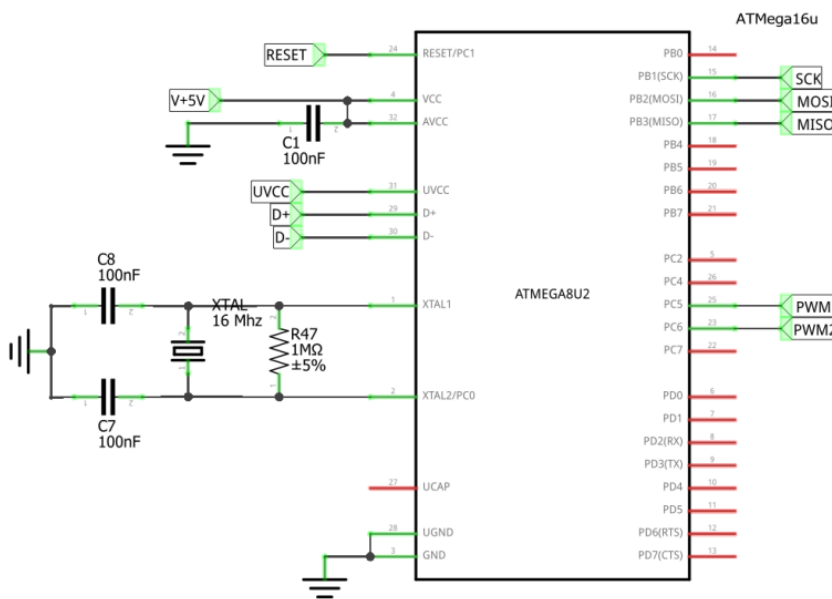
Alimentation et LED de fonctionnement



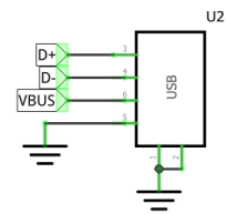
Bouton poussoir RESET



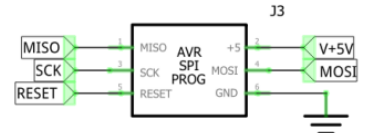
Microcontrôleur



Connecteur USB



Connecteur ICSP



Connecteur servomoteurs

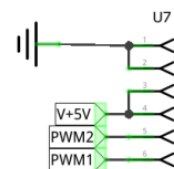


Image 3 : Vue schématique de la carte de contrôle réalisée sur Fritzing

Remarque : Des modifications vont néanmoins être apportées à ce PCB, comme notamment le changement de l'alimentation par une pile et la modification de certains composants

Partie 3 – Travail à effectuer et planning

1. Etapes à venir

a) Impression de la carte et câblage

La première étape à venir est d'imprimer et de câbler le PCB qui a été réalisé. Cela me permettra, avec ce premier prototype, de tester le fonctionnement global de la carte, les alimentations et les connexions. Suite à cela, il est possible qu'une nouvelle carte doive être imprimée si un problème est détecté.

b) Premiers tests de reconnaissance USB

Une fois la carte de contrôle câblée et fonctionnelle, je testerai la reconnaissance en périphérique USB. Pour cela j'injecterai un petit programme dans l'ATMega16u2 afin de vérifier dans un premier temps que le PC reconnaît bien la carte, puis que le PC peut communiquer avec la carte en utilisant des appuis claviers simples.

c) Programmation du microcontrôleur

Il faudra alors ensuite programmer entièrement l'ATMega16u2 afin qu'il puisse être reconnu en périphérique USB, qu'il puisse communiquer avec le PC (envoi et réception de données) et enfin qu'il soit capable de contrôler les servomoteurs en fonction des données envoyées par l'utilisateur.

d) Réalisation de la catapulte et tests finaux

Si la carte de contrôle est entièrement fonctionnelle, je pourrai alors réaliser au Fabricarium les différentes parties de la catapulte (socle, plate-forme, bras de projection) et les assembler autour de la carte et des servomoteurs. Je pourrai alors tester le fonctionnement de la catapulte assemblée et vérifier sa solidité, son équilibre et sa précision. Il sera sûrement nécessaire de faire plusieurs essais pour trouver la meilleure résistance, et pour finir la meilleure esthétique.

2. Planning

En fonction du temps restant ainsi que des prochaines étapes évoquées ci-dessus, j'ai pu élaborer un planning sous forme de diagramme de Gantt pour organiser mon temps jusqu'à la fin de ce projet. La partie « Gestion des servomoteurs » comprend le contrôle direct des positions par le microcontrôleur, mais également toute la partie échange de données avec le PC via USB, et le traitement des informations reçues. Concernant l'étape de réalisation mécanique du gadget, le créneau indiqué correspond à une deadline pour les éventuelles modifications et l'assemblage, mais les premières réalisations seront réalisées bien plus tôt.

Remarques : Ce planning est cependant subjectif et il est susceptible d'être mis-à-jour en fonction de l'avancée de certaines étapes, la gestion d'éventuels problèmes ou dysfonctionnements, ou d'autres facteurs extérieurs.

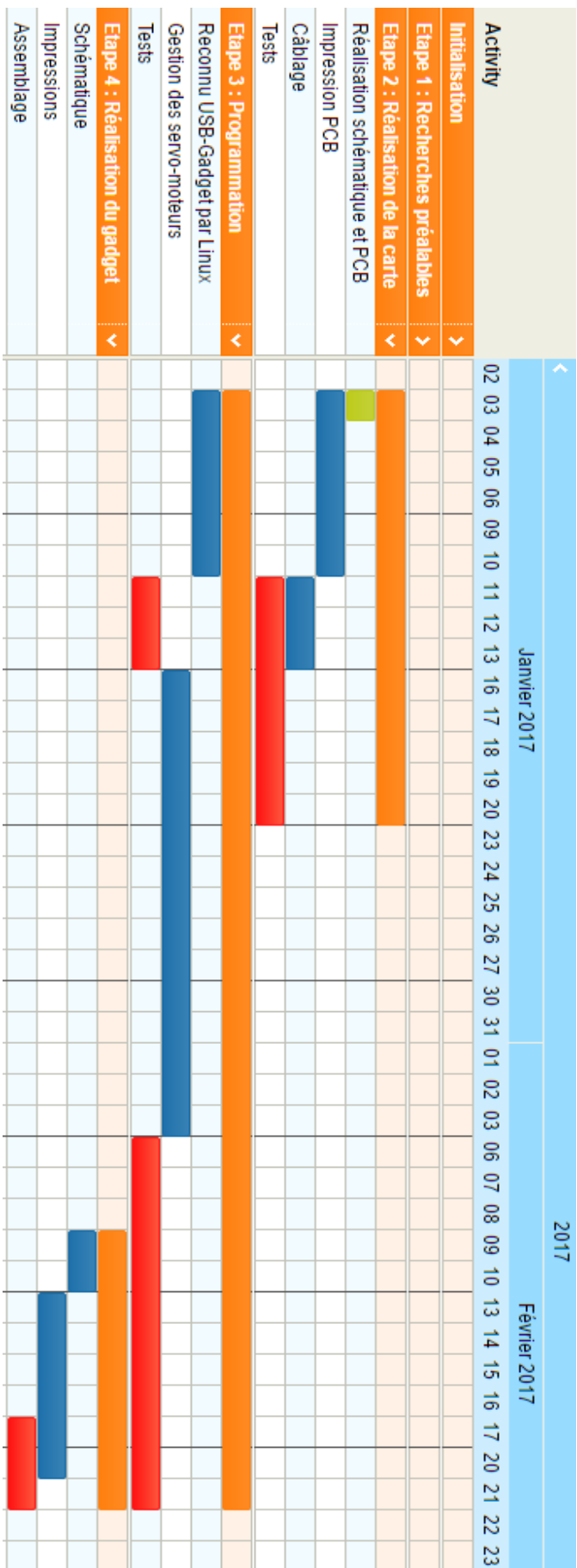


Image 4 : Planning des étapes à venir (diagramme de Gantt)

Conclusion

La première moitié du temps accordé à ce projet a donc principalement été utilisé pour définir le cahier des charges du gadget souhaité, rassembler de nombreuses informations sur les méthodes de contrôle et de fonctionnement du gadget et les démarches à suivre, effectuer quelques tests de périphérique USB et démarrer la réalisation de ma carte de contrôle.

J'ai pu ainsi mieux comprendre le fonctionnement d'un Arduino vis-à-vis de la liaison USB, et plus précisément de l'ATMega16u qui permet cette liaison. J'ai ainsi eu l'occasion de faire des tests sur un Arduino pour qu'il soit reconnu comme un périphérique USB et non comme une simple liaison USB-série.

Concernant le contrôle mécanique de la catapulte, j'ai pu bien assimiler le fonctionnement global des servomoteurs, et plus spécifiquement avec un signal de contrôle de type PWM disponible sur l'ATMega.

Du côté électronique, j'ai travaillé sur la carte de contrôle et j'ai réalisé une schématique et un premier prototype de PCB qui est sur le point d'être vérifié, imprimé et testé. Des modifications devront néanmoins être apportées avant la création de la carte finale afin de lui apporter plus d'efficacité et de pertinence.

La suite de ce projet sera donc de mettre à profit tous ces préparatifs dans la réalisation de ma carte de contrôle, la programmation de ma carte et la réalisation du gadget dans son ensemble. En parallèle de cela, je devrai également rédiger mon rapport final, tenir à jour le Twiki du projet, et entamer la préparation de la soutenance prévue pour février 2017.