

# *POWER GAUNTLET*

*BY*

*ALEXIS H.*

*&*

*QUENTIN L.*

*RÉALISÉ DANS LE CADRE DU PROJET ÉLEC  
DE 42*

*PROJET ELEC 2019*

# *POWER GAUNTLET*

## INTRODUCTION

## PRÉSENTATION DU PROJET

*LISTE DES COMPOSANTS  
DESCRIPTIONS DES DIFFÉRENTES PARTIES  
SCHÉMAS  
SCRIPTS*

## PROTOCOLES

*PWM  
I2C  
UART  
DMA  
SCRIPTS*

## SPÉCIFICATIONS TECHNIQUES

*CONSOMMATION ÉNERGÉTIQUE  
COÛT  
CARACTÉRISTIQUES TECHNIQUES  
APPLICATIONS*

## BILAN DU PROJET

## DOCUMENTATIONS

*POWER  
GAUNTLET*

# *INTRODUCTION*

# POWER GAUNTLET

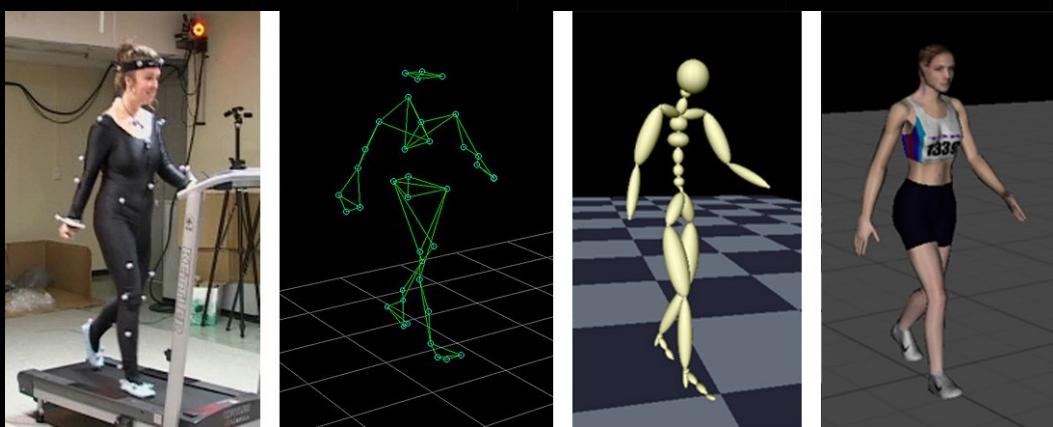
## INTRODUCTION

## ETUDE DE L EXISTANT

La capture de mouvement ou motion capture ou encore MOCAP est une technique permettant d'enregistrer les positions et rotations d'objets ou des membres d'être vivants, pour contrôler des bras robotiques, modèle 3D et bien d'autres chose. Cette technique permet et permettra d'améliorer grandement les interfaces homme-machine.

Nous allons à travers ce projet nous intéresser à la capture de mouvement d'une main humaine.

### Capture optique via caméra



L'image ci-dessus illustre bien les différentes étapes de ce procédé : une dame revêtue de différents capteurs qui permettent de modéliser son squelette via un algorithme de traitement d'images pour ensuite faire reproduire ses mouvements à un avatar.

Le point négatif de cette technique est qu'elle nécessite d'avoir une zone couverte par idéalement plusieurs caméras (comme dans un studio par exemple).

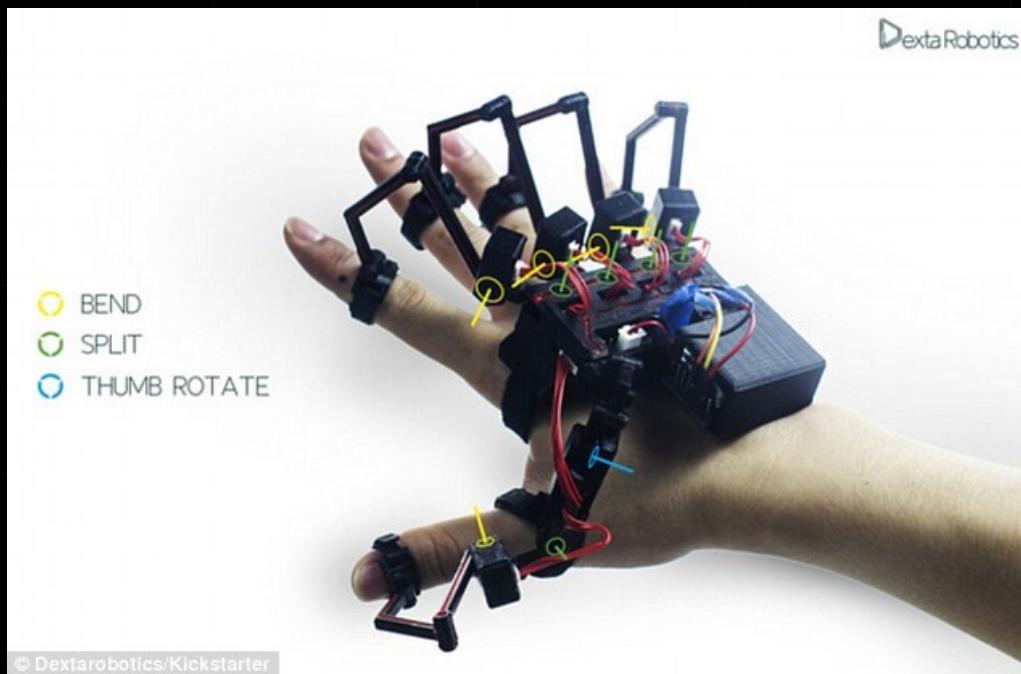
Le point positif de cette technique est qu'il est quand même possible d'obtenir des résultats avec juste une caméra.

# POWER GAUNTLET

## INTRODUCTION

## ETUDE DE L EXISTANT

### Capture mécanique



La capture mécanique se base sur un exosquelette sur lequel on fixe des potentiomètres aux différentes articulations afin d'en récupérer les mouvements.

Ci-dessus on peut en voir un qui utilise des codeurs numériques ( le principe reste le même) Chaque cercle coloré représente un axe de rotation dont on va récupérer l'angle.

L'avantage de cette technique est de permettre de récupérer avec précision le déplacement des doigts au millimètre.

L'inconvénient est l'encombrement par rapport à d'autre types de solutions.

# POWER GAUNTLET

## INTRODUCTION

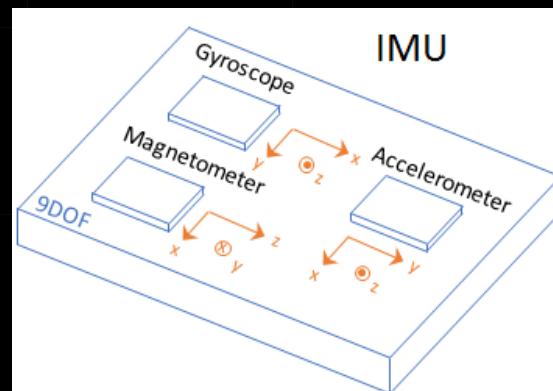
## ETUDE DE L EXISTANT

### Capture magnétique

Cette solution se base sur la création d'un champ électromagnétique et la distance et l'inclinaison par rapport à lui grâce à un capteur.

L'avantage de cette technique est la faible interprétation que l'on doit réaliser au niveau des calculs, cependant elle est très sensible à la moindre interférence ( aimant, appareil électrique etc...)

### Centrale inertielle



Une centrale inertielle est en réalité un assemblage de plusieurs capteurs, celui-ci comprend généralement un accéléromètre, un magnétomètre ainsi qu'un gyroscope. L'accéléromètre permet de calculer le déplacement dans l'espace. Le magnétomètre permet de déduire l'orientation dans l'espace. Le gyroscope permet de récupérer la vitesse angulaire.

Les différentes données sont inutilisables tels quels, on doit les fusionner pour obtenir des résultats convaincants, cependant les différentes étapes de calcul vont entraîner des pertes non négligeables sur le long terme rendant les mesures peu précises et donc peu utilisables.

*POWER  
GAUNTLET*

**PRÉSENTATION  
DU  
PROJET**

# *POWER GAUNTLET*

PRÉSENTATION DU PROJET

## *LISTE DES COMPOSANTS*

### Partie alimentation

- Piles LR6 x6
- LE33CZ-TR x1
- Fusible 200 mA x1
- Fusible 125 mA x1
- Diode Zener 5.6V x1
- Condensateur 100 nF x1
- Condensateur 2.2 µF x1

### Partie contrôle

- STM32f412RG x1
- IMU 6050 | BTW901 x1
  - HC06 x1
- Potentiomètres x5
- Transistor NPN2222A x5
- Resistance 20k ohm x5
- Condensateur 100 nF x5
- Condensateur 1 µF x1
- Condensateur 4.7 µF x1
- Condensateur 2.2 µF x1

### Partie moteurs

- Moteur vibrant x5
- Diode 1N4007 x5
- Condensateurs 100 nF x5

# POWER GAUNTLET

PRÉSENTATION DU PROJET

## ALIMENTATION

Cette partie regroupe :

- LE33CZ-TR x1
- Fusible 200 mA x1
- Fusible 125 mA x1
- Diode Zener 5.6V x1

L'alimentation est composée de deux parties :

2 fois 4.5V avec 3 piles LR6 dont les masses ont été reliée.

Nous avons modifier des boitiers afin de les équiper d'un meilleur interrupteur et pour plus facilement les intégrer dans le design de l'exosquelette.

Si nous avons choisi d'avoir deux alimentations, c'est pour avoir deux parties distinctes : une partie contrôle avec notre microcontrôleur et une partie puissance avec des moteurs.

La partie contrôle est régulé en 3.3V et est protégée contre l'inversion et la surtension grâce à des diodes et fusibles.

La partie puissance quant à elle est alimenter en 4.5 V.

# POWER GAUNTLET

PRÉSENTATION DU PROJET

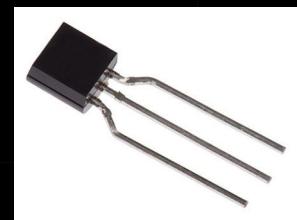
## ALIMENTATION

### LE33CZ-TR

Régulateur de type very low dropout afin de pouvoir supporter la chute de tension lorsque les piles faiblissent. Il nécessite pour stabiliser 2 condensateurs de 100 nF et 2.2 µF

#### Caractéristiques :

- Very low-dropout voltage (0.2 V typ.)
- Output current max : **150 mA**
- Output: **3.3V**
- Operating input voltage Max : **18V**



### 0242.200UAT1

Fusible pour protéger le régulateur, déclenchement rapide

#### Caractéristiques :

- Courant nominale : **200 mA**
- Tension nominale : **250V**

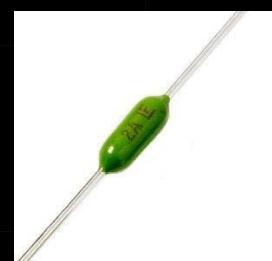


### 0251.125MAT1L

Fusible pour protéger le circuit

#### Caractéristiques :

- Courant nominale : **120mA**
- Tension nominale : **125V**



# POWER GAUNTLET

PRÉSENTATION DU PROJET

## ALIMENTATION

### BZX79C5V6

Afin de protéger le circuit contre de l'inversion, on utilise une diode Zener

#### Caractéristiques :

- Tension Zener : **5.6 V**
- Dissipation : **0.5 W**

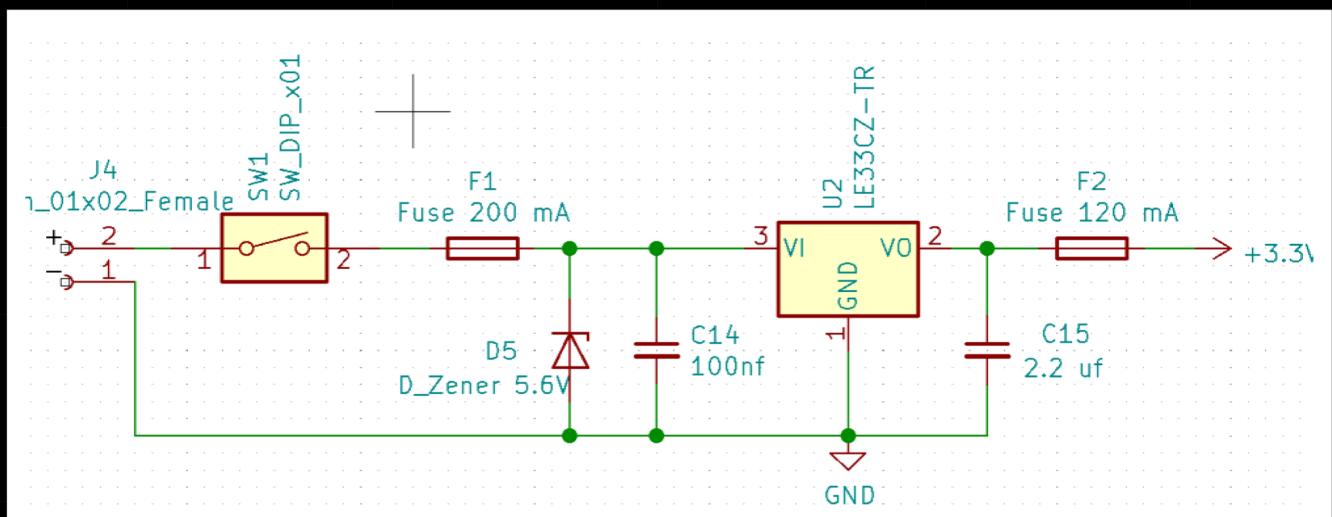


Schéma alimentation partie contrôle

# POWER GAUNTLET

PRÉSENTATION DU PROJET

## PARTIE DE CONTRÔLE

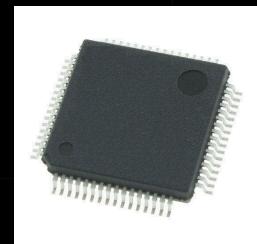
Cette partie regroupe :

- Stm32f412RG x1
- Module HC-06 x1
- IMU 6050 x1
- Potentiomètres x5
- Transistor NPN2222A x5
- Pin de programmation St-link x1

### STM32F412RG6 lqfp64

Nous avons opté pour ce modèle car il répondait à nos besoins :

- faible consommation
- Puissance de calcul élevée
- Floating Point Unit
- 5 PIN ADC
- 5 pin PWM
- TX / RX



Nous utilisons la clock interne à une fréquence de 16Mhz, cependant nous avons laissé un emplacement sur notre PCB pour pouvoir rajouter au besoin un oscillateur plus puissant.

Il se programme via ST-link grâce à des pin header.

#### Caractéristiques:

- Current voltage input : 1.7 to 3.6 V
- Data Bus Width: 32 bits
- Data RAM size : 256 kB
- Interface Type: i2c, UART, SPI, LIN



PRÉSENTATION DU PROJET

## PARTIE DE CONTRÔLE

### Module DSD TECH HC-06

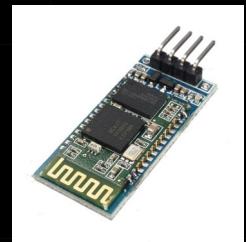
La communication avec l'ordinateur se fait par Bluetooth (via le protocole UART)

Pour cela nous avons utiliser un module HC-06.

Le baud rate utilisé est de 9600.

#### Caractéristiques :

- Input voltage : 3.3 to 6V



### Module IMU 6050

Cette centrale inertie comprend : un accéléromètre et un gyroscope.

Le protocole utilisé afin de communiquer avec le STM, est l'I2C.

Afin d'avoir une mesure le plus juste de l'inclinaison, nous avons implémenter deux méthodes différentes de calcul :

- un filtre complémentaire
- le filtre de Madgwick

#### Caractéristiques

- Résolution accéléromètre : 2 to 16g
- Résolution gyro : 250 to 2000 °/s
- Input Voltage : 3 to 5 V



### Potentiomètres :

Ils permettent de mesurer l'inclinaison des doigts, en modifiant la tension et en la mesurant grâce à une pin ADC. La valeur est ensuite stocké en DMA.

#### Caractéristiques :

- Resistance : 0—50 kΩ



# POWER GAUNTLET

PRÉSENTATION DU PROJET

## PARTIE DE CONTRÔLE

### Transistor NPN222A



Afin de contrôler les moteurs, tout en protégeant le circuit de contrôle d'un courant trop important, nous avons opté pour des transistors.

Afin que la vibration ne soit pas trop importante, nous avons choisi de mettre une résistance de 20 kilo ohm sur la base.

#### Caractéristiques :

- Collector-Emitter Voltage : 50 V
- Collector-Base Voltage : 75 V
- Emitter-Base Voltage : 6.0 V
- Collector Current – Continuous : 800 mA

$$3.3V / 20\,000 \Omega = 0.165 \text{ mA}$$

Selon la [datasheet](#), à cette intensité et à 25 °C, le HFE (current gain) est de 200, ce qui donne :  
33mA dans la partie puissance.

# POWER GAUNTLET

PRÉSENTATION DU PROJET

## PARTIE DE CONTRÔLE

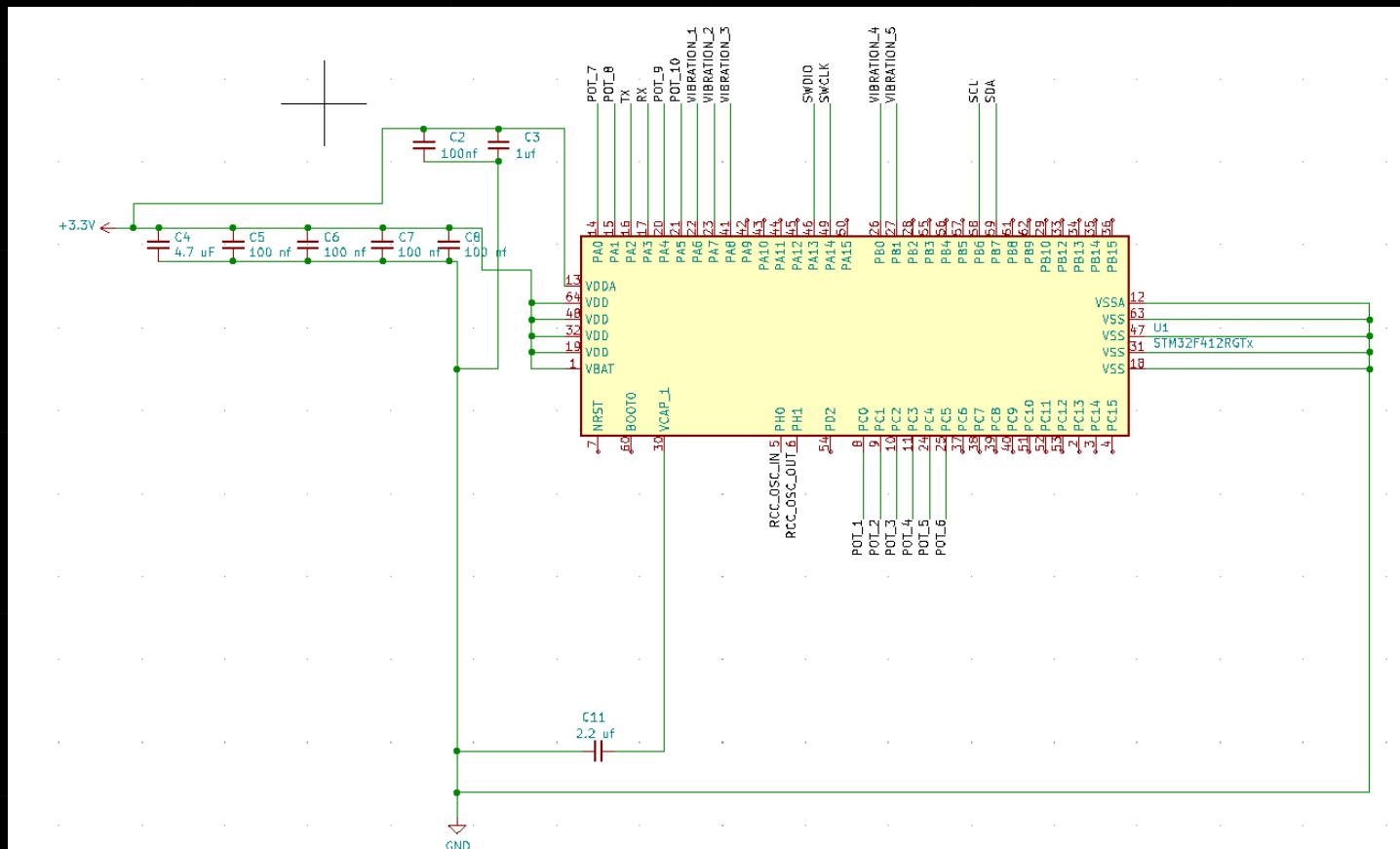


Schéma partie contrôle

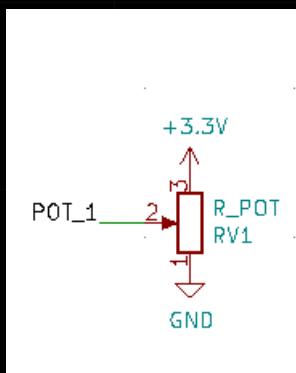


Schéma potentiomètre

# POWER GAUNTLET

PRÉSENTATION DU PROJET

## PARTIE MOTEUR

Cette partie regroupe :

- Moteur vibrant x5
- Diode 1N4007 x5
- Condensateurs 100nF x5

Vu que nous utilisons des moteurs, nous avons souhaité protéger les transistors des inversions de courants qui arrivent lors de l'arrêt des moteurs, pour cela nous avons rajouter sur notre montage des diodes pour que le courant puisse se déverser.

Nous avons également rajouté des condensateurs afin de lisser et de filtrer.

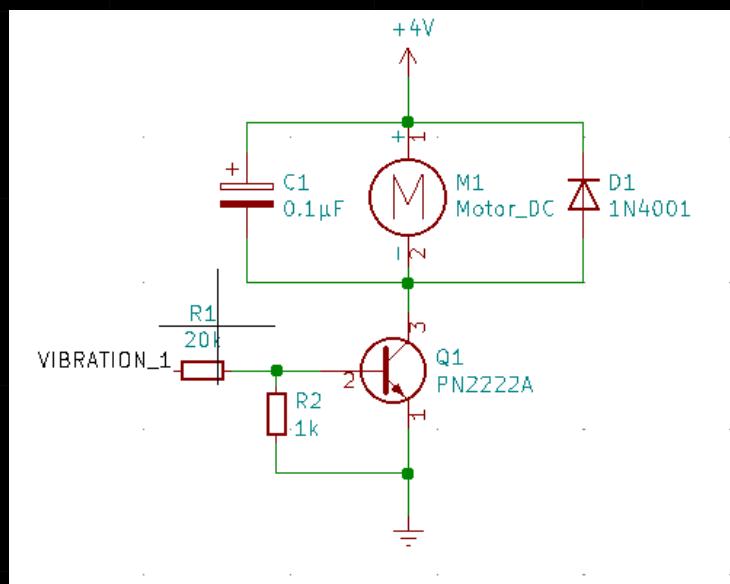
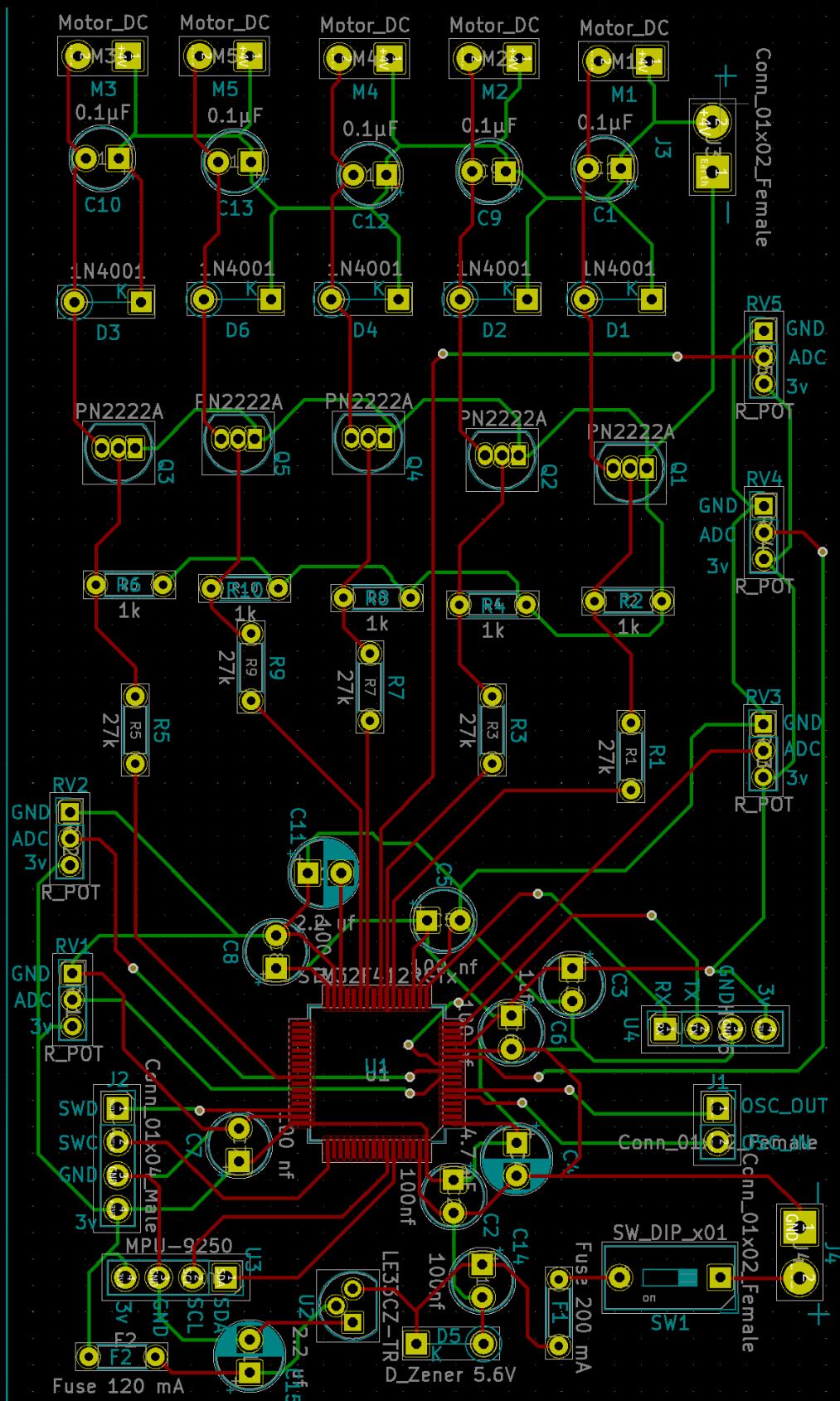


Schéma moteur

*Errata : la résistance de 1k(R2) est une erreur (elle n'est importante que si on utilise un MOSFET). De plus nous avons finalement choisi d'utiliser une résistance de 20k.*

# POWER GAUNTLET

## PRÉSENTATION DU PROJET



# POWER GAUNTLET

PRÉSENTATION DU PROJET

## PARTIE IMPRIMÉE

Cette partie regroupe :

- Exosquelette
- Boitier

Nous avons imprimé les pièces nécessaires à notre projet sur une imprimante 3D Ender 3. Tout a été imprimé en PLA à une température moyenne de 190 °C pour un remplissage moyen en 20% et 30%.

### Exosquelette



Composé de 34 pièces, nous avons récupérer le modèle sur Thingiverse, en réadaptant et en usinant parfois certaines pièces.

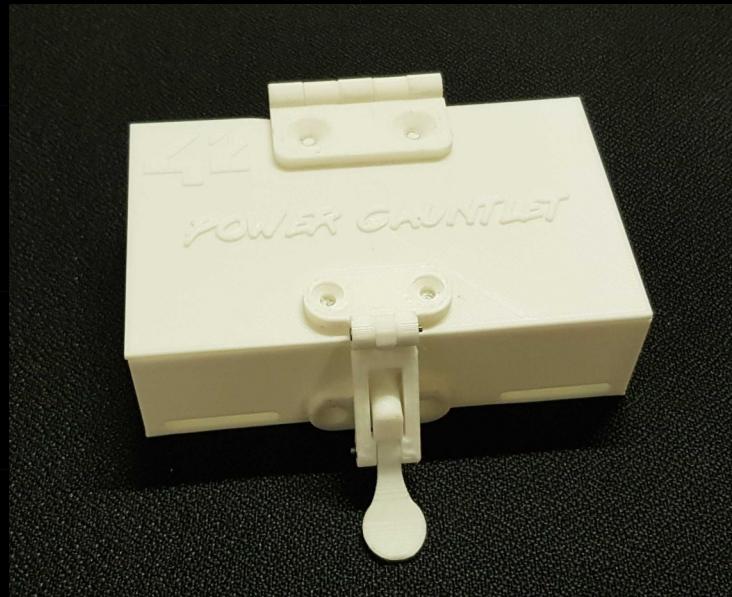
Pour les fixations, nous avons imprimer des fermoirs (également imprimée) ainsi qu'une bande élastique.

**POWER  
GAUNTLET**

**PRÉSENTATION DU PROJET**

## **PARTIE IMPRIMÉE**

### **Boitier**

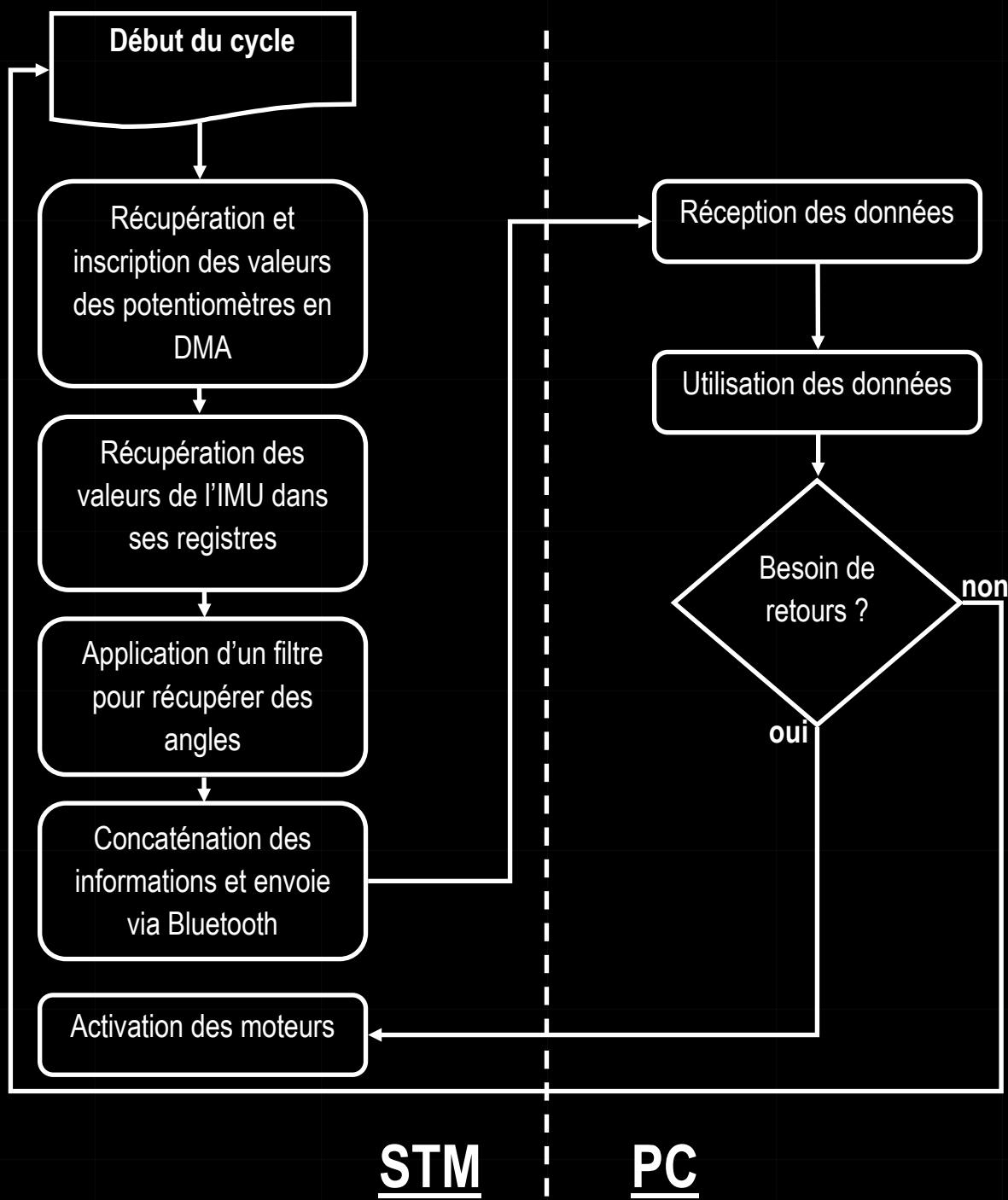


Nous l'avons modéliser dans fusion 360 le boitier et son couvercle, et nous avons récupérer un modèle de charnière et de loquet.

# POWER GAUNTLET

PRÉSENTATION DU PROJET

## SCHÉMA DE FONCTIONNEMENT



# **POWER GAUNTLET**

## **PRÉSENTATION DU PROJET**

### **COMMUNICATION ENTRE LE GANT ET LE PC**

**Le protocole de communication entre notre gant et le pc se sépare en deux parties.**

**1) La première partie consiste à envoyer l'état des différents moteurs à notre gant pour faire vibrer les moteurs ou non.**

**La requête envoyée à notre gant est de la forme suivante:**

Etat moteur 1 | Etat moteur 2 | Etat moteur 3 | Etat moteur 4 | Etat moteur 5

L'état d'un moteur caractérisé par la puissance de la vibration, comprise entre 0 et 255.

0 == moteur éteint

255 == moteur puissance maximum

*Exemple de requête envoyée : 255/000/255/100/050*

**2) La deuxième partie consiste à un envoi de notre gant vers le pc cible. La trame contiendra l'état des différents potentiomètres. Ici nous allons envoyer 24 requêtes les une à la suite des autres.**

**Requête envoyée par notre gant vers le pc**

Etat potard 1 | Etat potard 2 | Etat potard 3 | Etat potard 4 | Etat potard 5

L'état d'un potard 50k se caractérise par une valeur de retour comprise entre 0 et 1024.

0 == potard valeur min

1024 == potard valeur max

*Exemple de requête envoyée : 255/000/255/100/050*

# POWER GAUNTLET

PRÉSENTATION DU PROJET

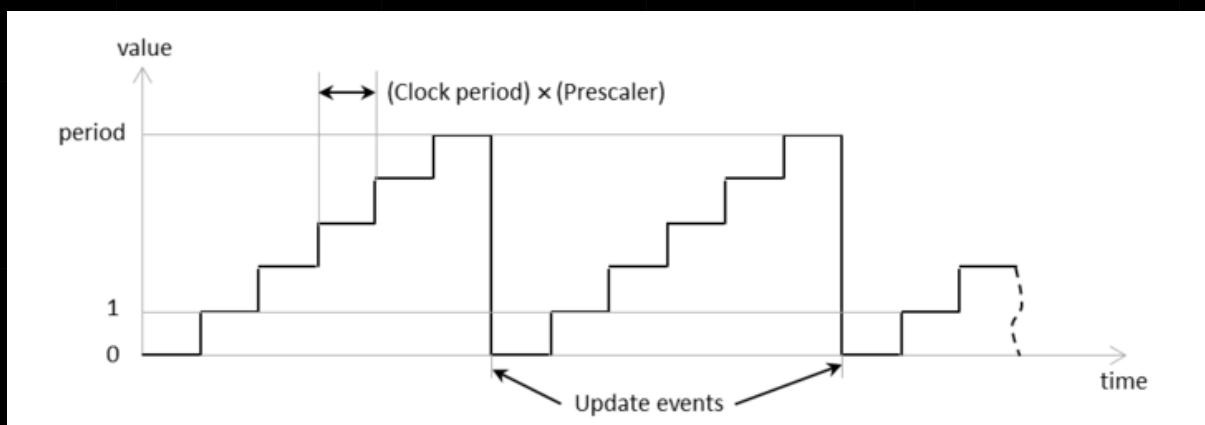
## PARAMÉTRAGE DU STM

Pwm:

Parametre de calibrage des timer sous cubemx:

**Prescaler:** le prescalaire designe un multiplicateur qui aura le role de multiplier par un facteur \* prescaler le temps pour que le timer se mette a jour.

**Couter Period :** Le counter period nous indique la valeur a laquelle le compteur va generer un "update events". On peut voir ca comme une frequence.



Les clock associees à nos timer sont calibrés a 16MHz.

**Pour calculer le temps d'update:**

$$(1/\text{MHz}) * \text{prescaler} * \text{count period} = (1 / 16\,000\,000) * 691 * 254 = 0.01 \text{ seconde}$$

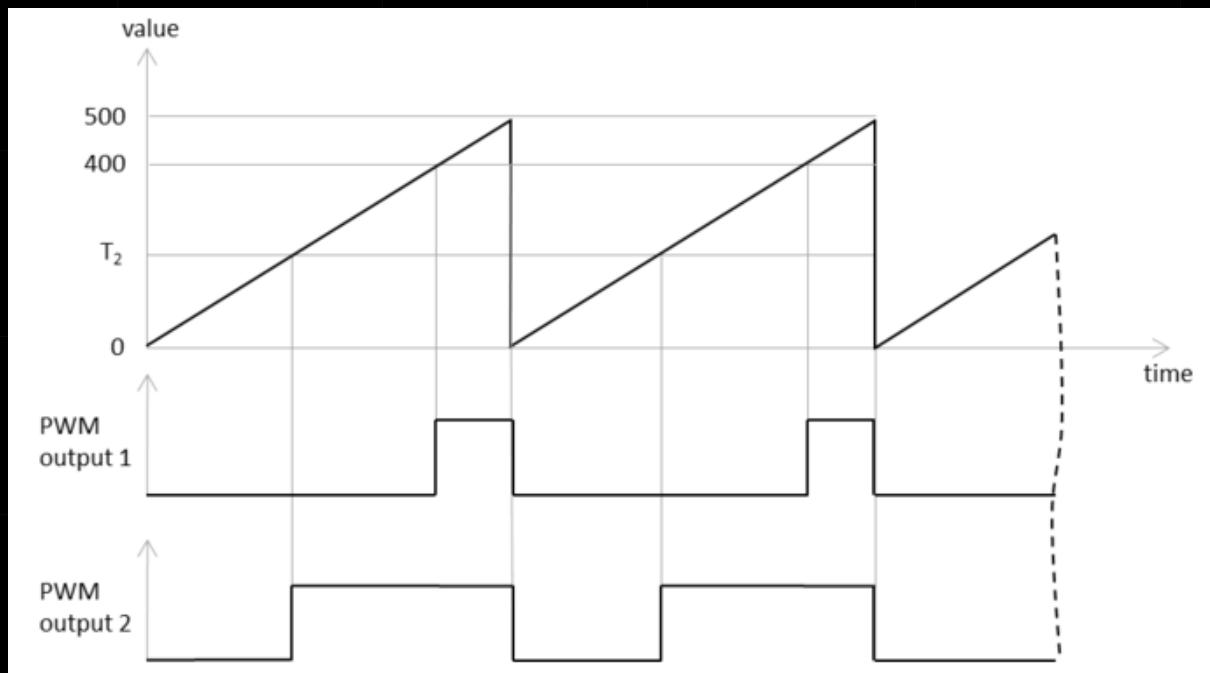
Donc notre timer genere un update event tout les 0.01 seconde.

**POWER  
GAUNTLET**

PRÉSENTATION DU PROJET

## PARAMÉTRAGE DU STM

Génération du PWM via les timers



# POWER GAUNTLET

PRÉSENTATION DU PROJET

## COMMUNICATION ENTRE LE GANT ET LE PC

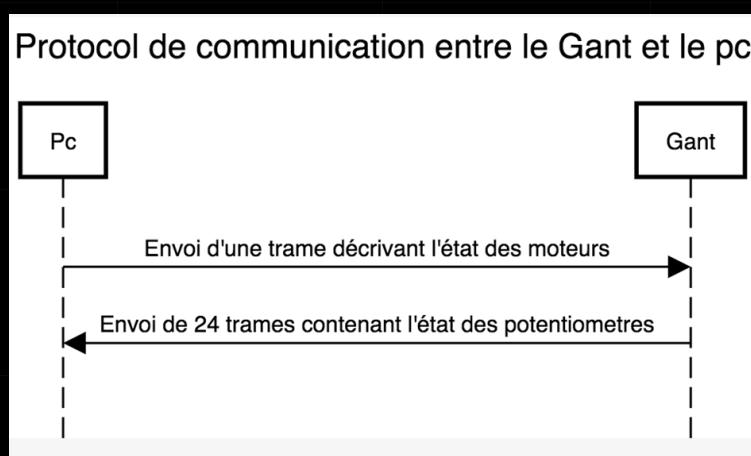
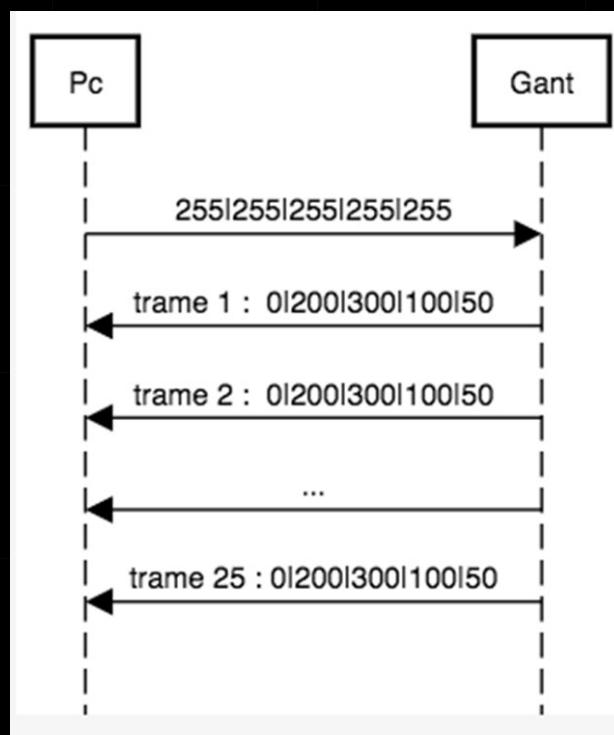


Schéma communication



Exemple



PRÉSENTATION DU PROJET

## PARTIE SOFTWARE

### //Script pour récupérer la valeur des potentiomètres

```
import serial
import time

name_serial = '/dev/cu.father-DevB'
output = ""
ser = 0
turn = 0

def b2_connection():
    global ser
    ser = serial.Serial(name_serial) # open serial port
    print(ser.name)                # check which port was really used

def b2_ask():
    global output
    global turn
    bread = ser.inWaiting()
    output = ser.readline()
    print(output)
    turn += 1

def b2_end():
    print ("Close b2 connection.")
    ser.close()                    # close port

def main():
    b2_connection()
    for i in range(0, 5000):
        b2_ask()
    b2_end()

main()
```



PRÉSENTATION DU PROJET

## PARTIE SOFTWARE

### //Script pour contrôler les moteurs

```
import serial
import time
name_serial = '/dev/cu.father-DevB'
output = ""
ser = 0
turn = 0
s1 = "255|255|255|255|255\0"
s2 = "000|000|000|000|000\0"

def b2_connection():
    global ser
    ser = serial.Serial(name_serial) # open serial port
    print(ser.name) # check which port was really used

def b2_ask():
    global output
    global turn
    global s1
    global s2
    print("Reverse state motor")
    data = raw_input()
    if ((turn % 2) == 0):
        print("All motor activation !")
        ser.write(s1)
    else:
        print("All motor off !")
        ser.write(s2)
    turn += 1

def b2_end():
    print ("Close b2 connection.")
    ser.close() # close port

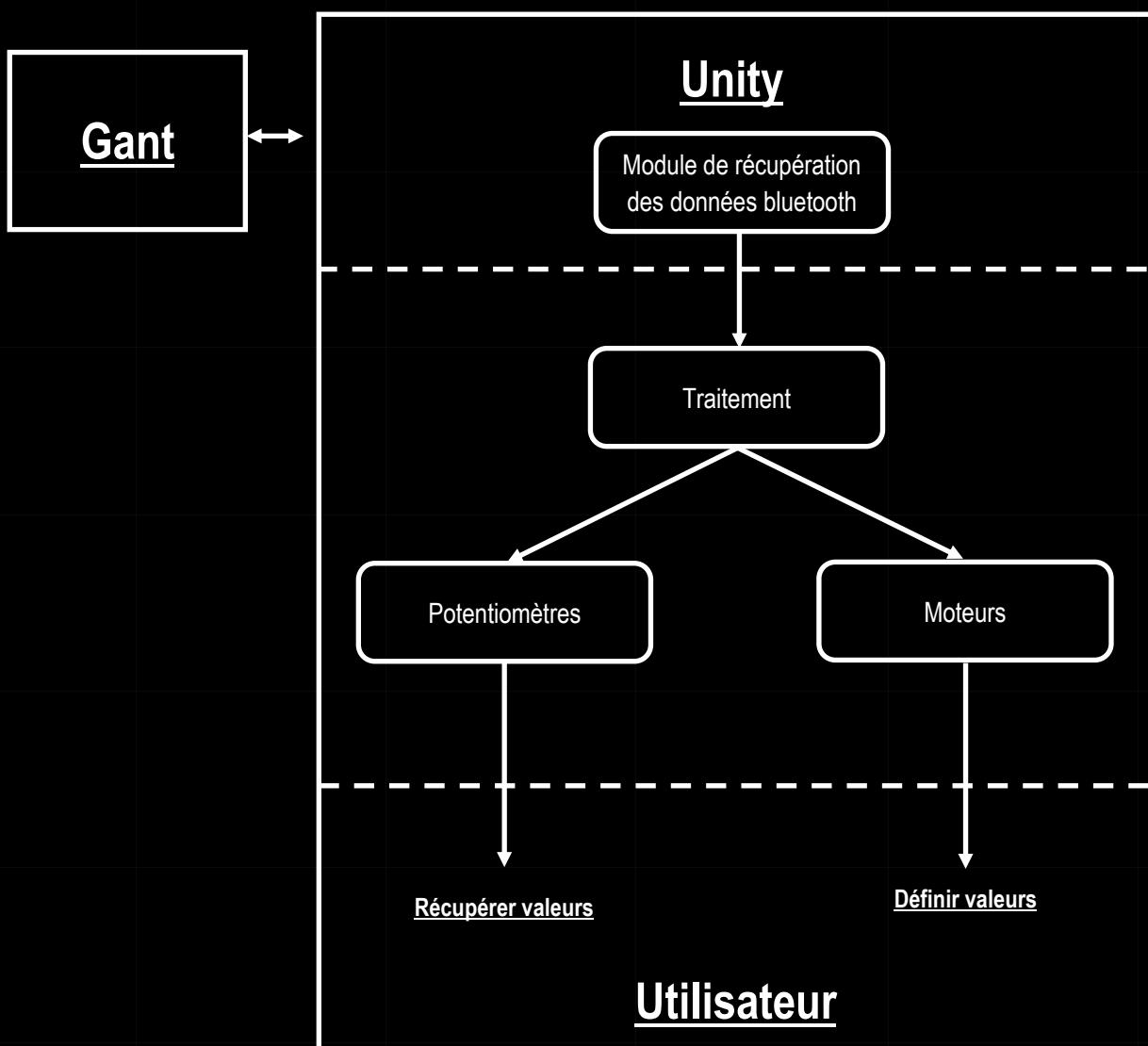
def main():
    b2_connection()
    for i in range(0, 500):
        b2_ask()
    b2_end()

main()
```

# POWER GAUNTLET

PRÉSENTATION DU PROJET

## SCHÉMA UNITY



*POWER  
GAUNTLET*

*PROTOCOLES*

# POWER GAUNTLET

## PROTOCOLES

### PWM

#### Description:

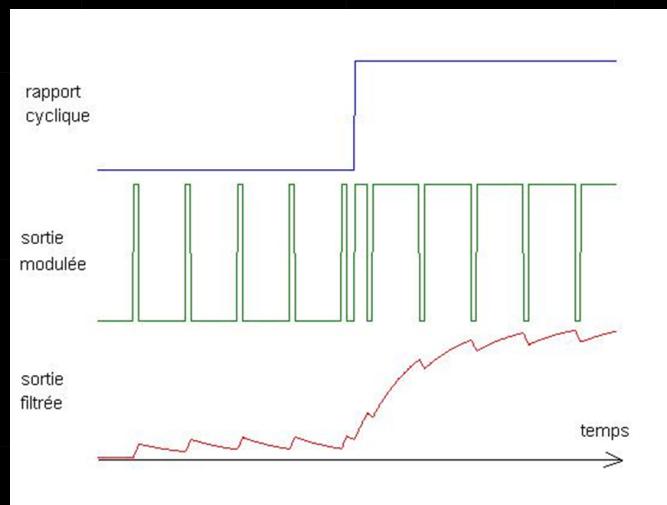
Pwm ou Pulse-width modulation est une technique permettant de simuler des signaux pseudo analogique à partir d'une sortie numérique.

#### Utilisation :

Nous nous servons de ce protocole pour moduler les vibration de nos moteurs vibrants.

#### Principe général :

Le principe général est qu'en appliquant une succession rapide d'états discrets avec des ratio de durée bien choisis, on peut obtenir *en ne regardant que la valeur moyenne du signal* n'importe quelle valeur intermédiaire.



Sur la figure ci-dessus on peut voir que la sortie filtré equivaut à la moyenne des états hauts au niveau de la sortie module.

# POWER GAUNTLET

## PROTOCOLES

### UART

#### Description :

UART ou ***Universal Asynchronous Receiver Transmitter*** est un mécanisme qui permet l'accès direct à la mémoire vive sans passer par le processeur permettant ainsi une accélération assez importante des performances pour les bus d'entrées/sorties (E/S ou I/O).

Pas besoin de clock, car asynchrone, cependant nous devons avoir le même baudrate sur les deux device. (1200, 4800, 9600, 19200, 38400, 115200).

Le Baud rate décrit la vitesse de transmission en bps(byte par second).

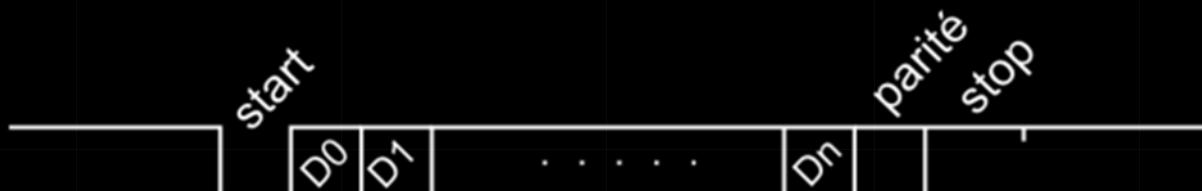
#### Constitution d'une trame UART

Bit de start toujours à 0 : sert à la synchronization du récepteur

Données : taille des data comprise entre 5 et 9bits

Parité : Paire ou impaire(controle d'erreur)

Fin : bit de stop, toujours à 1



#### Niveau de tension:

Niveau de tension de type TTL, soit 0V ou 5V

#### Vitesse de transmission(en baud):

110, 300, 1200, 2400, 4800, 9600, 19200, ..., 3686400 baud.

#### Utilisation dans le projet :

Nous utilisons ce protocole avec le module Bluetooth(HC-06) pour communiquer et recevoir différents types d'informations tels que la valeurs des potentiomètre ou l'activation des moteurs.

# POWER GAUNTLET

## PROTOCOLES

### I2C

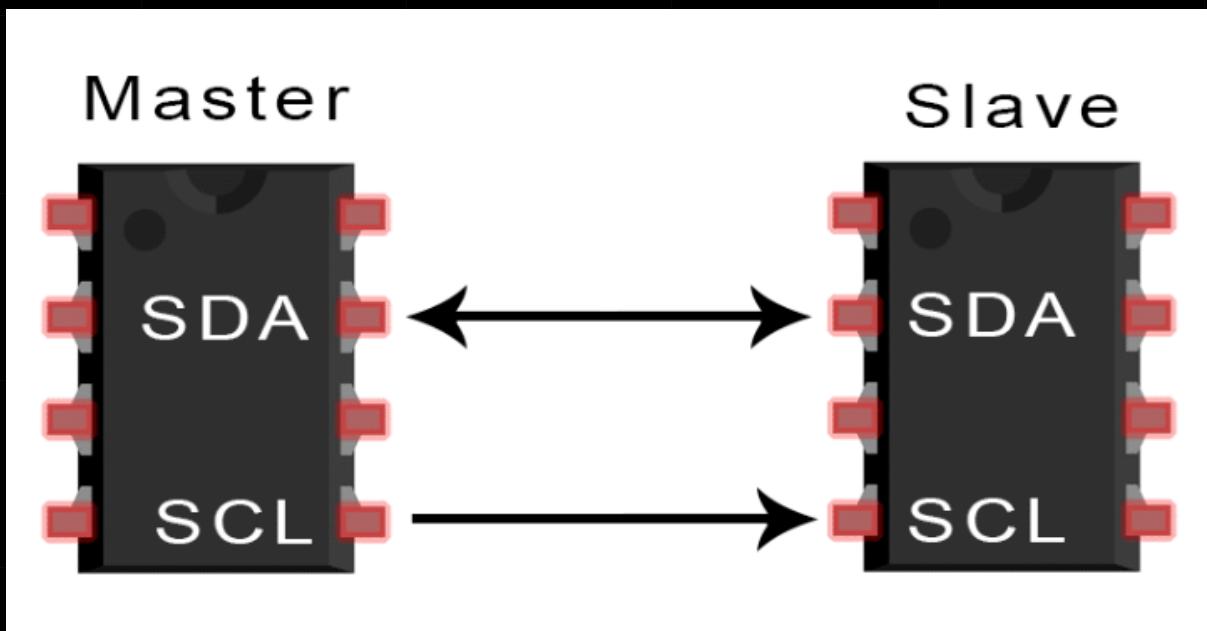
#### Description :

I2C est un bus série permettant de transmettre des informations de façon asynchrone entre divers circuits connectés.

L'i2c se compose de deux lignes.

- SDA (Serial data line) : ligne de données bidirectionnelles
- SCL (Serial Clock line) : ligne d'horloge de synchronisation bidirectionnelle

Ce protocole définit par Intel à l'avantage d'utiliser uniquement 2 pins pour connecter une multitude de composants divers et variés.



#### Temps et vitesse de transmission:

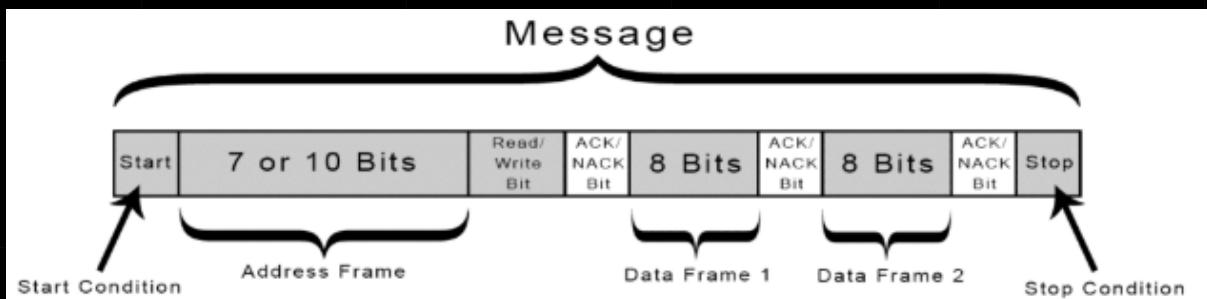
- « Standard mode (Sm) » ≤ 100 kbit/s,
- « Fast mode (Fm) » ≤ 400 kbit/s,
- « Fast plus mode (Fm+) » ≤ 1 Mbit/s,
- « High-speed mode (Hs-mode) » ≤ 3,4 Mbit/s,
- « Ultra-fast mode (UFm) » ≤ 5 Mbit/s, **unidirectionnel** uniquement.

# POWER GAUNTLET

## PROTOCOLES

### I2C (SUITE)

Fonctionnement general:



Chacun des messages contient l'adresse de l'esclave à joindre ainsi que le message a transmettre.

La trame se compose de plusieurs partie:

Start condition : Debut de trame, debut de la communication.

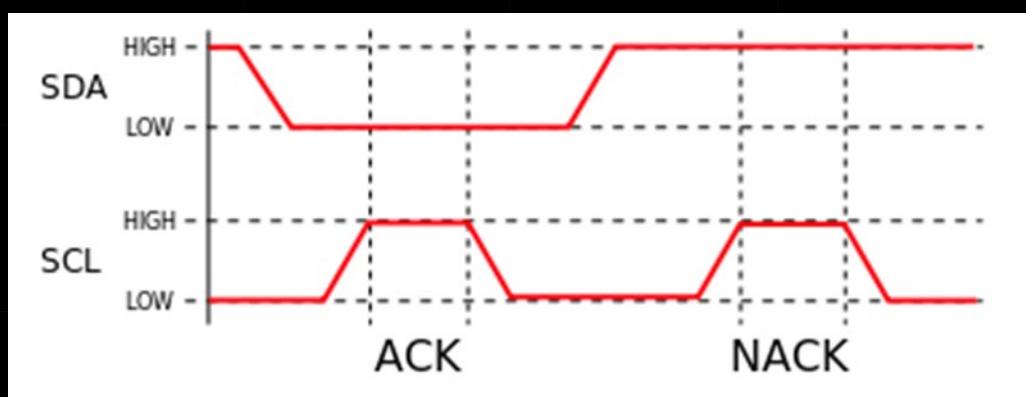
Stop condition : indique la fin de la trame à transmettre.

Address Frame : désigne l'adres i2c de l'esclave que l'on veut joindre.

Read/write bit :indique si on veut lire des données ou en écrire.

ACK bit : Forcer la ligne SDA au niveau LOW pour signaler la bonne reception de l'octet, equivalent a 0.

NACK bit: laisser la ligne SDA a un niveau HIGHT pour indiquer un défaut dans la reception de l'octet.



# *POWER GAUNTLET*

## *PROTOCOLES*

## *DMA*

### **Description :**

L'inconvénient des interruptions est qu'on oblige le processeur à 2 changements de contexte : un au début et un à la fin du traitement de l'événement.

Pour des processeurs hautes performances utilisant un pipe-line d'exécution le coût de ce changement de contexte est particulièrement sensible.

Un sous-programme d'interruption consiste souvent à lire un mot depuis un registre d'entrée puis à l'écrire en mémoire dans une zone tampon (ou inversement à lire une donnée en mémoire et l'écrire dans un registre de sortie).

Ce travail sans la moindre complexité algorithmique peut être réalisé par un contrôleur spécialisé appelé contrôleur DMA (DMAC).

Celui-ci organise le transfert de données entre la mémoire et un contrôleur d'entrées-sorties (périphérique), dans un sens ou dans l'autre. Le processeur continue donc à vaquer à ses occupations et son pipe-line n'est pas perturbé comme pour une interruption.

### **Fonctionnement du DMA :**

- Le sous-système de DMA va être configuré et démarré par le CPU
- Les données vont être échangées via des cycles de lecture ou d'écriture en mémoire initiés par le sous-système de DMA
- Lorsque les opérations seront terminées le sous-système de DMA va interrompre le CPU

### **Utilisation dans le projet:**

Nous nous servons du DMA dans notre projet pour récupérer les valeurs des potentiomètres.

*POWER  
GAUNTLET*

**SPÉCIFICATIONS  
TECHNIQUES**

# POWER GAUNTLET

## SPÉCIFICATIONS TECHNIQUES

### CONSOMMATION ÉLECTRIQUE

#### Partie contrôle

Hc06:

> 30 ~ 40mA

Stm32f412RG :

>  $112 \mu\text{A}/\text{MHz} * 16 \text{ MHz} = 1.792\text{mA}$

IMU 6050 :

> 3.9 mA

Totale partie commande : ~45 ma

#### Partie puissance

Moteur (individuel) :

~33 mA

Totale partie puissance (au maximum) : 165 ma

# POWER GAUNTLET

## SPÉCIFICATIONS TECHNIQUES

### AUTONOMIE

Si on considère que :

- Batterie 4.5V délivre 7900 mAH

Cela donne environ :

**170** heures pour la partie commande

**50h** pour la partie de puissance

# POWER GAUNTLET

## SPÉCIFICATIONS TECHNIQUES

### COÛT

Estimation des Cout de fabrication

Composants	Unité	Cout (en euros)
STM32F412RG	1	8
Regulateur	1	0.50
HC-06	1	5
IMU	1	2
PCB	1	3
Moteurs	5	3
Fusibles	2	5
Compartiment à piles	2	3
PLA (kilo)	0.50	8
Divers (diodes/ résistances / condensa- teurs / câbles		6
<b>Total :</b>		<b>35</b>

(En comparaison, le prix de lancement du Power Glove de Nintendo lors de son lancement était de 75 \$ soit 150 \$ actuellement)

**POWER  
GAUNTLET**

**SPECIFICATIONS  
TECHNIQUES**

**CARACTÉRISTIQUE  
TECHNIQUE**

Dimension (en cm) :

*Exosquelette : 14 \* 20 \* 8*

*Boitier : 8 \* 13 \* 3*

Poids :

*600g*

Température d'utilisation :

*0 - 40 c°*

Norme de protection :

*IP00*

Alimentation :

*6 piles LR6*

Compatibilité :

*Mac OSX / Windows 10 / linux*

Durabilité\* :

*200 heures*

*(\*usure des composants en PLA)*

# *POWER GAUNTLET*

## *SPÉCIFICATIONS TECHNIQUE*

### *POINT FORTS*

- Faible coût
- Facilement modifiable et améliorable
- S'adapte facilement aux morphologies
- Visuellement marquant

### *POINT FAIBLES*

- Précision inférieur comparé aux autres méthodes existantes
- Durabilité
- Encombrement

# POWER GAUNTLET

## SPÉCIFICATIONS TECHNIQUES

### APPLICATIONS

#### Robotique



Pouvoir contrôler à distance et s'interfacer avec une machine comme par exemple un bras robotique.

#### Jeu vidéo



Créer de nouvelles idées de gameplay et pouvoir interagir de façon encore plus poussée avec l'environnement.

#### Artistique



Pouvoir interagir de façon physique avec l'univers numérique avec des spectacles de musique / danse connectée.

#### Education



Permettre à des personnes de recréer de A à Z ce projet qui touche à un nombre important de discipline.



## ***SPÉCIFICATIONS TECHNIQUES***

# ***ÉTAPES AVANT COMMERCIALISATION***

### **Pourquoi commercialiser ce projet?**

Les techniques autour du motion capture ont commencé à prendre de plus en plus d'importance, notamment pour repenser la façon de concevoir des interfaces homme-machine. Les solutions proposées sont couteuses pour le grand public. Sans optimisation de cout a grande échelle nous nous retrouvons avec un projet pouvant être réalisés pour 35 euros ayant les capacités de proposer suffisamment de fonctionnalité pour être attractif.

Bien que le produit en l'état actuel ne peut être vendu, nous pouvons déjà imaginer une V2 qui permettra de résoudre les différents problèmes/améliorations (CF. partie améliorations).

Nous devrions également faire appel à des professionnels de l'électronique embarquée pour nous conseiller des améliorations/correction sur notre circuit actuel pour augmenter sa durée de vie et sa consommation électrique. Nous pourrions également faire appel à des ingénieurs spécialisés dans la conception d'exosquelette pour améliorer cette partie.

### **Modèle économique**

Nous pensons d'ailleurs à mettre l'ensemble de nos travaux sous Creative Commons / Open Sources pour permettre à n'importe qui de reprendre le projet ou de concevoir sa propre version.

Un modèle économique pourrait être la vente de Kit avec tout les composants nécessaires à réalisation du projet.

*POWER  
GAUNTLET*

**BILAN DU  
PROJET**



**BILAN DU PROJET**

## **MÉTHODOLOGIE DE TRAVAIL**

### **Etapes**

- 1) Documentation sur le fonctionnement d'un IMU **[mars – avril]**
- 2) Faire fonctionner un IMU sur Arduino **[avril]**
- 3) Documentation sur le STM32**[mai]**
- 4) Faire fonctionner les différentes parties du projet sur une board de développement **[mai – juin]**
- 5) Conception de la PCB **[juillet]**
- 6) Conception de l'exosquelette **[juillet]**
- 7) Tests et développement du code **[juillet – aout]**
- 8) Travail sur la version finale et sur les outils **[aout – septembre]**

### **Rythme de travail**

6 jours sur 7

### **Outils**

Gitlab  
Trello

# *POWER GAUNTLET*

**BILAN DU PROJET**

## ***DIFFICULTÉS RENCONTRÉES***

### IMU

Au tout début, nous souhaitions mettre une centrale inertielle par doigt et ainsi récupérer de façon très précise l'orientation ainsi que les déplacements.

Cependant nous nous sommes rendu compte de plusieurs problèmes :

- 1) Bien que très intéressant mathématiquement et physiquement, nous n'aurions pas appris grand chose en électronique.
- 2) Il a été démontré que le calcul des déplacements en utilisant uniquement des IMU est beaucoup trop imprécis (à cause notamment de la double intégration pour récupérer la position à partir de l'accélération).

Par la suite nous utilisé un IMU 6050, pour avoir la position de la main, cependant la précision laisse à désirer et dérive beaucoup avec le temps.

# *POWER GAUNTLET*

**BILAN DU PROJET**

## ***DIFFICULTÉS RENCONTRÉES***

### **Erreurs PCB**

Notre PCB comporte actuellement plusieurs erreurs :

- Absence de masse commune entre les deux alimentations

*Notre solution :*

Nous avons modifier les boitiers des piles pour créer une masse commune entre les deux.

- Erreur de footprint sur le régulateurs

*Notre solution :*

On a tordu les pattes afin de les faire coïncider avec la PCB.

- Partie inutile (ancienne résistance de bleeding)

*Notre solution :*

Nous avons laissé vide ces parties là.

# POWER GAUNTLET

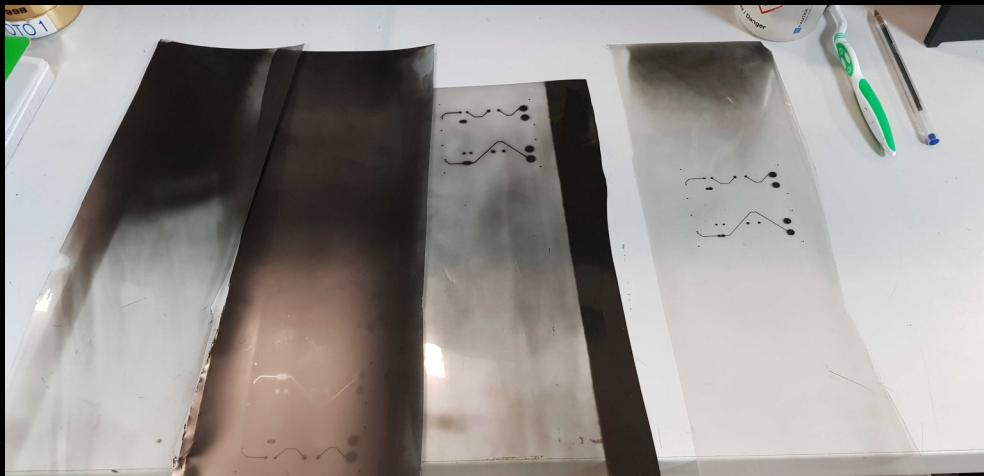
## BILAN DU PROJET

### DIFFICULTÉS RENCONTRÉES

#### Création de la PCB



Nous nous sommes formé à l'Electrolab de Nanterre afin de pouvoir réaliser nous même la PCB. Nous avons appris à faire un typon, à utiliser l'insoleuse ainsi que les différents produits. Cependant, la procédure permet difficilement d'avoir une qualité équivalente que si on passait par un professionnel. C'est pour cela que nous avons au final abandonné cette idée.



*POWER  
GAUNTLET*

*BILAN DU PROJET*

*AMÉLIORATIONS  
POSSIBLE*

**Voici quelques pistes d'améliorations possibles :**

- Ajouter 5 nouveaux potentiomètres pour autoriser un nouvel axe de liberté pour les doigts
- Prendre un microcontrôleur plus petit
- Tout faire en CMS afin de réduire la taille
- Rajouter des leds témoins
- Rajouter une diode pour protéger le régulateur lorsqu'on flash le STM
- Mettre un port USB pour le debugge
- Diminuer l'encombrement en terme de câbles ainsi que pour les batteries

# POWER GAUNTLET

## BILAN DU PROJET

### COMPÉTENCES ACQUISES

#### Savoir

##### Electronique / programmation

- Fonctionnement d'un IMU
- Protocole USART / I2C
- Architecture STM
- notions diverses en électronique et en électricité

##### Sciences

- Géométrie dans l'espace
- Dérivée / intégrale / Quaternions
- Physique du mouvement
- Magnétisme

#### Savoir être

- Travailler en groupe
- Coder propre
- Faire des commit compréhensible
- Eplucher des datasheets
- Se former par soi même

#### Savoir faire

##### Electronique

- Souder des composants
- Designer circuit avec Kicad
- Créer une PCB
- Debugger un circuit
- Utiliser le décalage et les registres
- Utiliser le DMA
- Flasher un microcontrôleur
- Faire un montage avec des transistors
- utiliser Nucléo

##### Programmation

- Utiliser de l'orienté objet
- Utiliser Unity pour créer des scènes
- Utiliser CubeMX pour paramétriser l'architecture d'un STM32

##### Conception

- Modéliser et réadapter des objets avec fusion 360

# **POWER GAUNTLET**

## **BILAN DU PROJET**

## **BILAN PERSONNEL**

### Alexis H.

*« Ce projet d'électronique dans le cas de l'école 42, m'a permis de découvrir un nouveau monde et une nouvelle façon d'appréhender les objets contenant de l'électronique.*

*J'ai pu ainsi faire le tour des différents techniques liés aux motions capture et apprendre beaucoup de nouvelle notion passionnantes.*

*Travailler sur un projet sur une longue période en équipe m'a permis de développer mon esprit peer 2 peers à son maximum.*

*Ce fut également le cas de faire une belle rencontre.*

*Merci à Fabrice L. de nous avoir permise le réaliser. »*

### Quentin L.

*« j'ai pris énormément de plaisir à faire ce projet : le côté transdisciplinaire était vraiment très stimulant . J'ai appris pleins de choses dans de nombreux domaines. J'ai également beaucoup apprécié le côté physique et manuel du projet : c'était très gratifiant de créer un objet de A à Z.*

*7 mois dessus, ça peut être long mais j'ai eu la chance de tomber sur un binôme très sympathique qui m'a énormément appris.*

*Merci à Fabrice pour son encadrement et ses conseils ainsi qu'à toutes les personnes qui ont pu nous filer un coup de main à un moment à un autre (L'electrolab et Flax notamment).*

*POWER  
GAUNTLET*

# *DOCUMENTATION*

# **POWER GAUNTLET**

## **DOCUMENTATION**

### **Datasheet:**

STM32f412RG :

<https://www.st.com/resource/en/datasheet/stm32f412ce.pdf>

HC-06 :

<https://www.olimex.com/Products/Components/RF/BLUETOOTH-SERIAL-HC-06/resources/hc06.pdf>

LE33CZ-TR:

<https://www.st.com/resource/en/datasheet/lexx.pdf>

Transistor NPN2222A

<http://web.mit.edu/6.101/www/reference/2N2222A.pdf>

Moteur vibrants :

<https://www.verical.com/datasheet/adafruit-motors-1201-4576014.pdf>

Fusible 200 mA

[https://www.tme.eu/Document/04e5b8c0f745bdf786e21d059a76c3d4/Littelfuse\\_Fuse\\_242.pdf](https://www.tme.eu/Document/04e5b8c0f745bdf786e21d059a76c3d4/Littelfuse_Fuse_242.pdf)

Diode Zener :

<https://www.onsemi.com/pub/Collateral/BZX79C2V4-D.PDF>

Fusible 120 mA:

<https://docs-emea.rs-online.com/webdocs/0f9e/0900766b80f9e8e9.pdf>

# POWER GAUNTLET

## DOCUMENTATION

### Modele 3D

Exosquelette :

<https://www.thingiverse.com/thing:2782111>

Hinge :

<https://www.thingiverse.com/thing:1083876>

Latch :

<https://www.thingiverse.com/thing:3283176>

Buckle for exosquelette :

<https://www.thingiverse.com/thing:1738270>

Buckle for box :

<https://www.thingiverse.com/thing:2934177>

### Liens divers

Thèse de Sebastian Madgwick :

[https://www.enib.fr/~kerhoas/RESEAU\\_CAPTEURS /documents/INVENSENSE/madgwick\\_internal\\_report.pdf](https://www.enib.fr/~kerhoas/RESEAU_CAPTEURS /documents/INVENSENSE/madgwick_internal_report.pdf)

Documentation sur les quaternions et Madgwick

[https://www.enib.fr/~kerhoas/rescapt\\_cours\\_mpu9250.html](https://www.enib.fr/~kerhoas/rescapt_cours_mpu9250.html)

<https://fr.wikipedia.org/wiki/UART>

[https://fr.wikipedia.org/wiki/Modulation\\_de\\_largeur\\_d'impulsion](https://fr.wikipedia.org/wiki/Modulation_de_largeur_d'impulsion)

<http://www.dicofr.com/cgi-bin/n.pl/dicofr/definition/20010101001435>

[http://www.gecif.net/articles/genie\\_electrique/cours/terminale/cours/le\\_bus\\_I2C.pdf](http://www.gecif.net/articles/genie_electrique/cours/terminale/cours/le_bus_I2C.pdf)

<https://circuitdigest.com/tutorial/serial-communication-protocols>

<https://fr.wikipedia.org/wiki/I2C>

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