

ROULE MA POULE SANS NID-DE-POULE

---

## **CAHIER D'ANALYSE**

---

3 mars 2019

Bastian Bouchardon  
Polytech Tours  
Département informatique industrielle

# Table des matières

1.1	Introduction . . . . .	2
1.2	Rappel du contexte de la réalisation . . . . .	2
1.2.1	Contexte . . . . .	2
1.2.2	Objectifs . . . . .	2
1.3	Rappel de l'architecture . . . . .	3
1.4	Électronique . . . . .	4
1.4.1	Interfaçage piézoélectrique . . . . .	4
1.4.2	Premier montage piézoélectrique . . . . .	5
1.4.3	Second montage piézoélectrique . . . . .	6
1.4.4	Troisième montage piézoélectrique . . . . .	7
1.4.5	Accéléromètre . . . . .	8
1.5	Informatique . . . . .	10
1.5.1	Fonctionnement général . . . . .	10
1.5.2	Classe Debug . . . . .	11
1.5.3	Classe Acl . . . . .	12
1.5.4	Classe main . . . . .	12
1.5.5	Séquences livrables partie 2 . . . . .	13
1.5.5.1	Premier démarrage . . . . .	13
1.5.5.2	Utilisation courante . . . . .	14
1.5.5.3	Boîtier autonome . . . . .	15
	References . . . . .	16

## **1.1 INTRODUCTION**

Ce document spécifie l'analyse du projet "Roule ma poule sans nid-de-poule", projet encadré par Emmanuel Neron et Nicolas Monmarche. Il met en évidence les schémas électroniques liés aux tests effectués sur les capteurs, les diagrammes UML liés à l'analyse du développement logiciel embarqué pour la partie 1 du projet expliquée ci-dessous.

## **1.2 RAPPEL DU CONTEXTE DE LA RÉALISATION**

### **1.2.1 Contexte**

Aujourd'hui, de nombreux moyens de déplacement existent pour faire les trajets quotidiens comme se rendre au travail, aller faire des courses, se déplacer à des événements culturels. Il y a les transports en commun comme le bus, le tramway ou le covoiturage et des transports personnels comme la voiture, le skateboard, les appareils électriques type mono wheel, hoverboard et le vélo. Ce dernier est de plus en plus utilisé pour les trajets quotidiens car il permet de garder une bonne forme physique, il est plus rapide que la voiture en ville, peu coûteux et facile d'entretien. Les cyclistes sont sensibles à l'état de la route, en effet, les routes abîmées, écorchées peuvent au mieux endommager, au pire les blesser. En France, d'après plusieurs études, les routes se dégradent, des motards sont décédés à cause de nids de poule, des accidents ont souvent lieu impliquant des cyclistes et des défauts de la route. C'est pour ces raisons que ce projet peut être apprécié des utilisateurs de vélo.

### **1.2.2 Objectifs**

Il s'agit de réaliser un système embarqué sur tout type de vélo permettant de capter les vibrations dues à la route et d'envoyer ces informations avec une position GPS sur un serveur. Cela permettra de cartographier l'état des routes et de prévenir les usagers et collectivités. Ce système sera constitué, au final, d'une carte électronique contenant l'intelligence, le capteur, le récepteur GPS, l'alimentation et la communication sans fil. Au départ, il faudra tester et choisir une technologie de capteur qui correspondra à notre utilisation. Il faudra également créer un serveur pour recevoir ces informations.

### 1.3 RAPPEL DE L'ARCHITECTURE

Voici un rappel de l'architecture du système : un boîtier composé d'un capteur de vibration, une alimentation, un système de communication (bluetooth, wifi), un système de géolocalisation et une mémoire permettant de fonctionner sans communication. Il transférera les informations à un serveur par le biais du téléphone du cycliste.

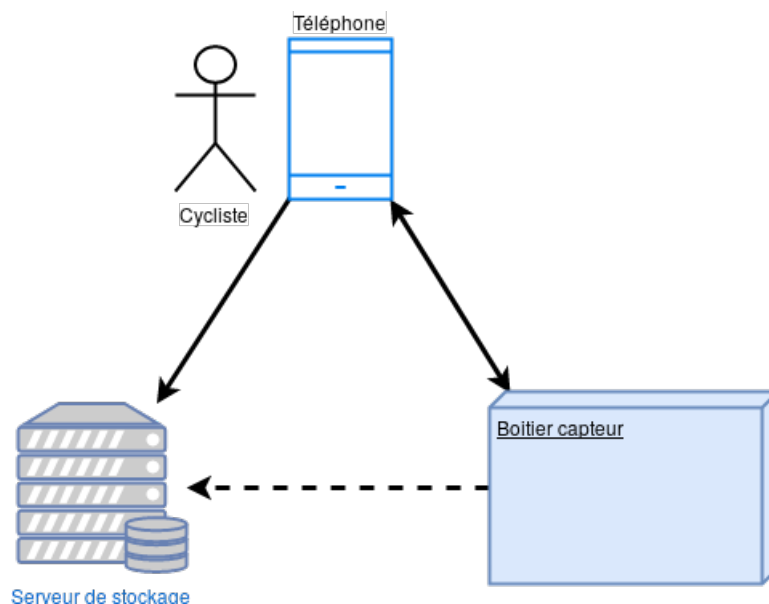


FIGURE 1.1 – architecture générale du système

Voici les blocs fonctionnels du boîtier qui sera fixé sur le vélo de l'utilisateur :

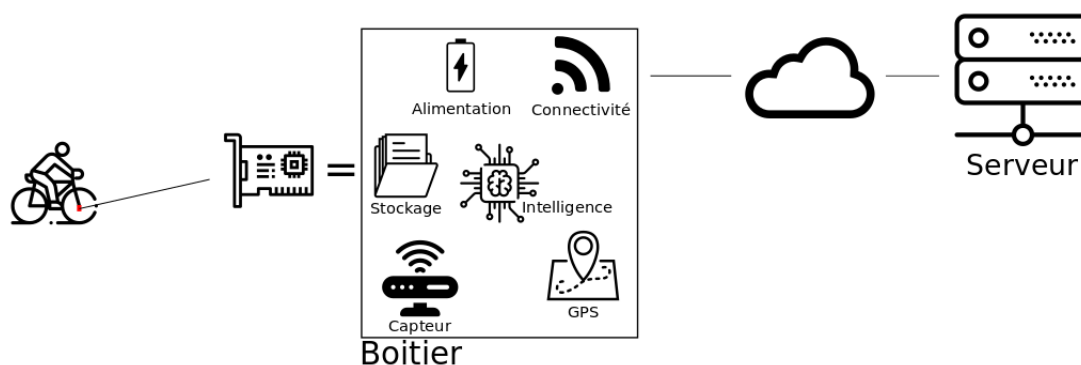


FIGURE 1.2 – schéma fonctionnel du boîtier

Ce document présentera dans un premier temps les différents montages électroniques permettant de tester des capteurs piézoélectrique et accéléromètre correspondant au premier livrable.

## 1.4 ÉLECTRONIQUE

### 1.4.1 Interfaçage piézoélectrique

Après des recherches sur les capteurs piézoélectriques, voici une sélection de deux schémas permettant d'interfacer le capteur à notre "intelligence", c'est-à-dire à une entrée d'un convertisseur analogique/numérique.

Un capteur piézoélectrique est un composant qui se polarise électriquement sous l'effet d'une contrainte mécanique. dans notre cas nous avons des capteurs piézoélectriques avec une masse permettant de se déformer lors de vibrations suffisantes.

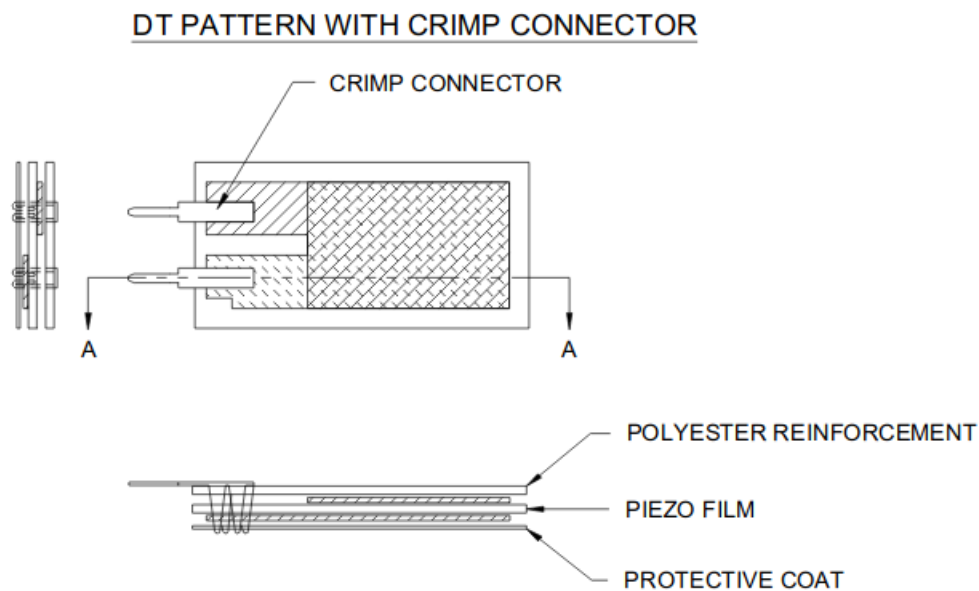


FIGURE 1.3 – Capteur piézoélectrique sans la masse

Dans notre cas, une masse permet d'être plus sensible aux basses fréquences donc plus sensible aux vibrations dues à la route.

### 1.4.2 Premier montage piézoélectrique

Le capteur piézoélectrique génère une tension en fonction des chocs subis par le capteur. Le premier schéma consiste simplement à mettre une résistance en parallèle du capteur permettant de le charger et de mesurer les variations de tension à l'aide d'un oscilloscope.

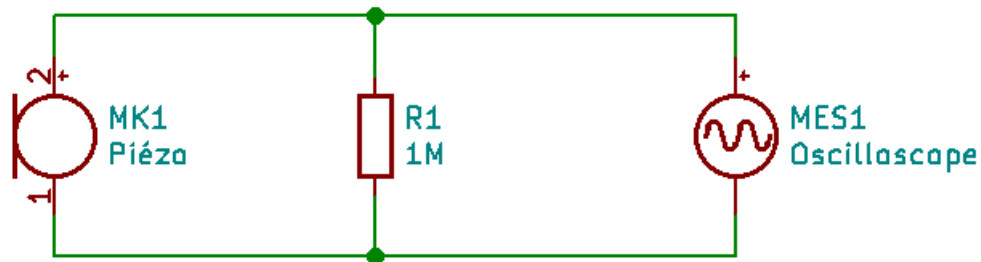


FIGURE 1.4 – Schéma piézo + résistance + oscilloscope

Nous pouvons remarquer sur l'oscillogramme ci-dessous qu'à chaque choc, une onde est générée. Cette onde oscille autour du 0V. Le piézoélectrique ne peut être connecté directement au microcontrôleur car des tensions négatives l'endommageraient.

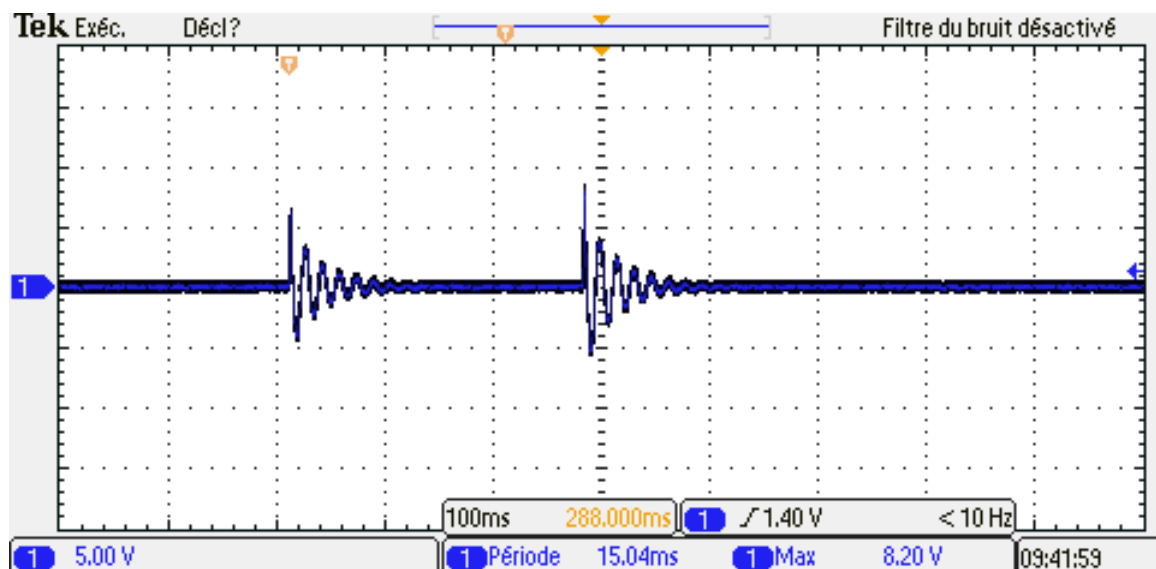


FIGURE 1.5 – Oscillogramme piézoélectrique + résistance

Pour pouvoir s'interfacer avec "l'intelligence", nous pouvons adapter le signal de deux façons différentes. Soit garder un signal analogique adapté qui sera traité par une entrée CAN ou bien commander un transistor de type MOSFET pour avoir des impulsions logiques

correspondantes aux chocs qui seront traitées par une simple entrée GPIO ou une entrée d'interruption.

### 1.4.3 Second montage piézoélectrique

Le capteur piézoélectrique génère une tension envoyée au convertisseur analogique / numérique mais ses variations de tension peuvent dépasser le seuil autorisé par le convertisseur et éventuellement le détruire. Ce signal est alternatif, il varie autour de la référence. Pour qu'il soit compatible avec une entrée CAN il faut passer par 3 étapes :

- Amplifier le signal pour convenir aux vibrations de la route
- Filtrer le signal avec un filtre détecteur d'enveloppe équivalant à un démodulateur AM.
- Enfin, adapter l'impédance du montage pour qu'il puisse "driver" le CAN.

Le TL072 est adapté à la première et dernière fonction. C'est un composant comprenant deux AOPs en un avec un très faible bruit et une grande gamme de tensions d'alimentation. L'un des AOP permettra d'amplifier le signal provenant du capteur piézoélectrique et le second permettra d'adapter l'impédance en le câblant en mode suiveur. Pour réaliser le filtre, il suffit d'une diode, une résistance et un condensateur.

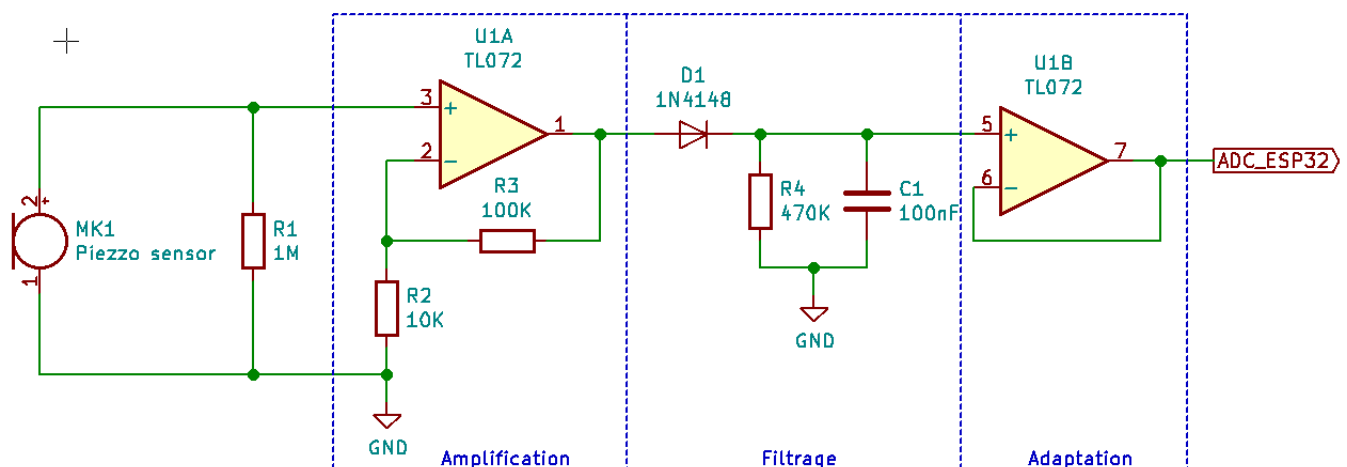


FIGURE 1.6 – Schéma adaptation analogique

Nous pouvons voir sur l'oscillogramme ci-dessous que nous avons transformé grâce au filtre détecteur d'enveloppe, l'oscillation par une rampe. Cela permet au CAN ensuite de détecter plus simplement les impacts dus aux routes. L'amplificateur opérationnel a joué son rôle d'adaptation d'impédance et d'amplification.

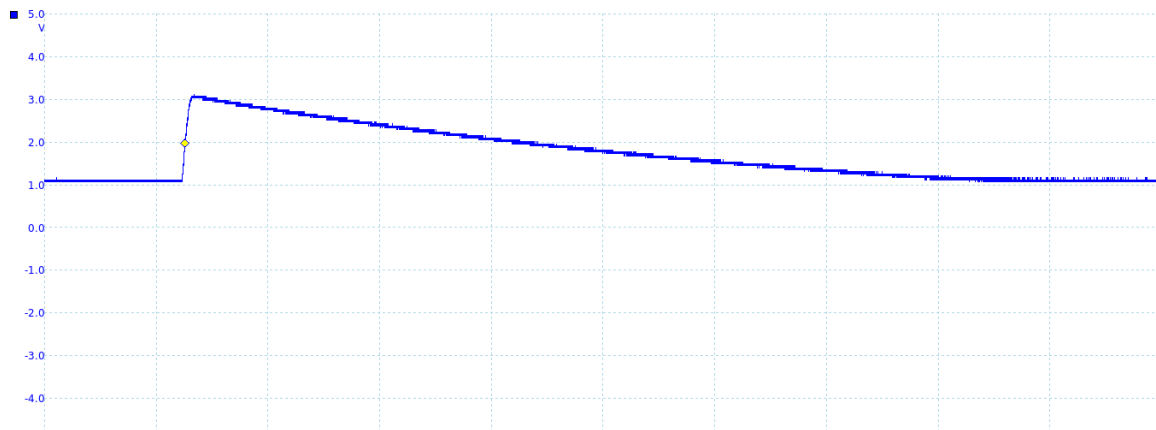


FIGURE 1.7 – Oscillogramme schéma n°2

#### 1.4.4 Troisième montage piézoélectrique

Un capteur piézoélectrique délivre une tension image de la déformation du matériau qui le compose. Dans notre application nous voulons pouvoir détecter des vibrations, des chocs. Une tension analogique n'est pas nécessaire à cette détection, "l'intelligence" peut se contenter d'une pulsation, d'un changement de niveau logique. Ce dernier montage permet de transformer la tension analogique en une pulsation logique compatible avec une entrée GPIO logique de l'ESP32. Sur le schéma ci-dessous, la première partie est le filtre détecteur d'enveloppe permettant de rendre positif le signal et d'annuler les oscillations. Puis le signal commande un transistor MOSFET canal N par la grille. Grâce à une résistance de pull-up sur le drain, une entrée GPIO pourra détecter les changements de niveau.

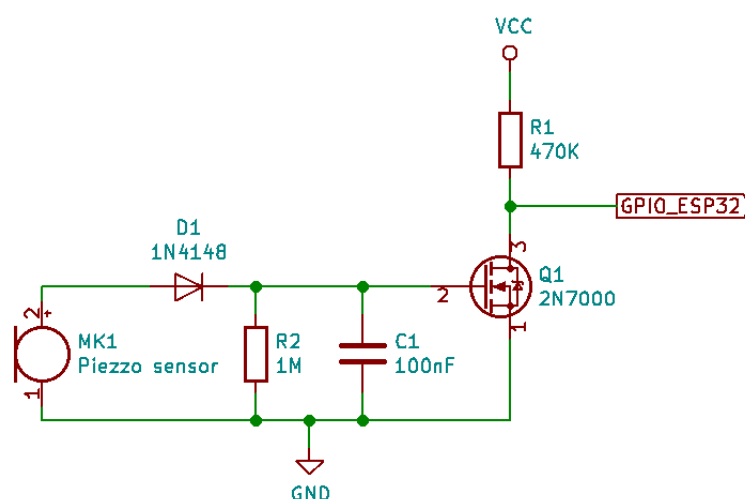


FIGURE 1.8 – Schéma adaptation numérique



Nous pouvons remarquer sur la figure ci-dessous que nous détectons bien les chocs. Nous transformons le signal analogique par un signal carré compatible avec une entrée numérique. De plus, plus l'impact est fort, plus la transition  $0V \rightarrow +VCC$  est longue. Nous pourrions détecter la force de l'impact en mesurant la durée du signal carré.

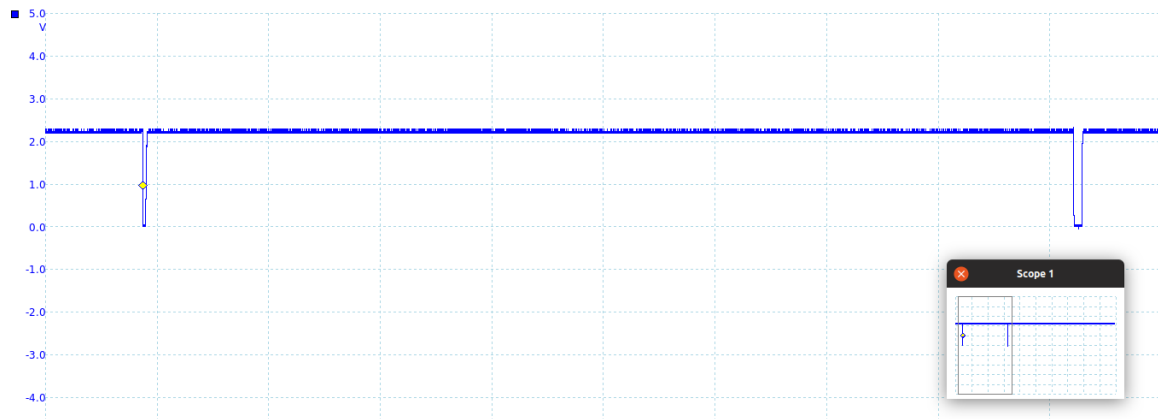


FIGURE 1.9 – Oscillogramme schéma n°3

Après ces schémas, on peut voir que le troisième montage pourra tout à fait convenir à notre besoin et il est plus économique en nombre de composants, complexité et coût. Nous n'avons pas besoin d'un signal analogique pour détecter un choc sur la route.

### 1.4.5 Accéléromètre

Un accéléromètre est un capteur, qui fixé sur un mobile, permet de mesurer l'accélération. La plupart du temps, un accéléromètre est constitué de trois capteurs, chacun sur un axe orthogonal (X, Y, Z) permettant de capter les vibrations dans toutes les directions.

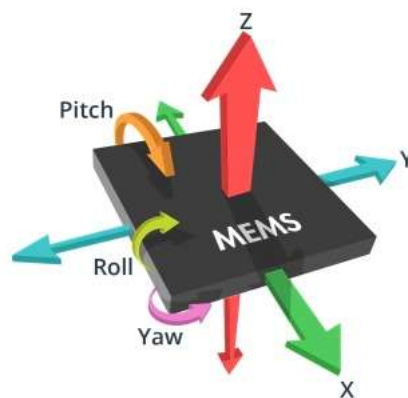


FIGURE 1.10 – Accéléromètre, principe

Nous pouvons l'utiliser pour capter des vibrations sur un axe orthogonal correspondant à un axe vertical par rapport au vélo du cycliste. Pour tester un accéléromètre, j'ai décidé d'utiliser un ADXL345 du fondeur Analog Devices. Il peut être connecté à un processeur par I<sup>2</sup>C ou SPI. On peut choisir la résolution en g<sup>1</sup> avec 2, 4, 8, et 16g. Cela nous permet de définir la résolution souhaitée pour notre utilisation. Voici ses caractéristiques :

- Ultralow power : as low as 23  $\mu$ A in measurement mode and 0.1  $\mu$ A in standby mode at VS = 2.5 V (typical)
- Power consumption scales automatically with bandwidth
- User-selectable resolution Fixed 10-bit resolution Full resolution, where resolution increases with g range, up to 13-bit resolution at  $\pm 16$  g (maintaining 4 mg/LSB scale factor in all g ranges)
- Embedded memory management system with FIFO technology minimizes host processor load
- Single tap/double tap detection
- Activity/inactivity monitoring
- Free-fall detection
- Supply voltage range : 2.0 V to 3.6 V
- I/O voltage range : 1.7 V to VS
- SPI (3- and 4-wire) and I<sup>2</sup>C digital interfaces
- Flexible interrupt modes mappable to either interrupt pin
- Measurement ranges selectable via serial command
- Bandwidth selectable via serial command
- Wide temperature range (-40°C to +85°C)
- 10,000 g shock survival

---

1. g : accélération de la pesanteur soit environ 9.81 m/s en S.I.

J'ai choisi un module Digilent Pmod ACL Accelero permettant d'interfacer le ADXL345 plus simplement. Voici le schéma de câblage :

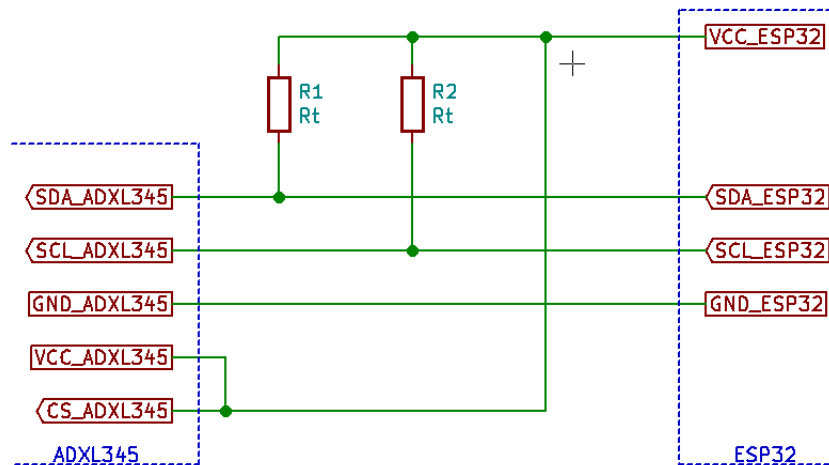


FIGURE 1.11 – Schéma connexion Pmod ACL Digilent

Pour terminer le premier livrable c'est-à-dire choisir le capteur et le montage qui conviendra à notre besoin, il faut tester sur une route test et un vélo de test ces montages. Il faut comparer les résultats et choisir le montage qui aura la bonne sensibilité face aux obstacles, aux chocs dus à des aspérités de la route et à des nids-de-poule.

## 1.5 INFORMATIQUE

Dans cette partie nous abordons les détails du logiciel embarqué dans l'ESP32 pour valider le premier livrable avec des diagrammes de classes, des diagrammes de programmation et des diagrammes de séquence.

### 1.5.1 Fonctionnement général

Lors des tests, il faudra pouvoir recueillir des données provenant des capteurs à intervalles réguliers ou les stocker pour les récupérer dans un second temps. L'ESP32 est pourvu d'une communication WiFi et Bluetooth ce qui permettra de pouvoir envoyer ces données sans fil sur un ordinateur pour ensuite les stocker et les visualiser.

Pour cela j'ai mis en place une classe "Debug" permettant d'initialiser une connexion série et/ou WiFi puis d'envoyer des données à travers ces communications.

J'ai également créé une classe "Acl" permettant l'utilisation de l'accéléromètre de façon simple.

Pour les capteurs piézoélectriques, je lis simplement une entrée analogique ou numérique dans la boucle principale dans la classe "main". Voici le diagramme de classes :

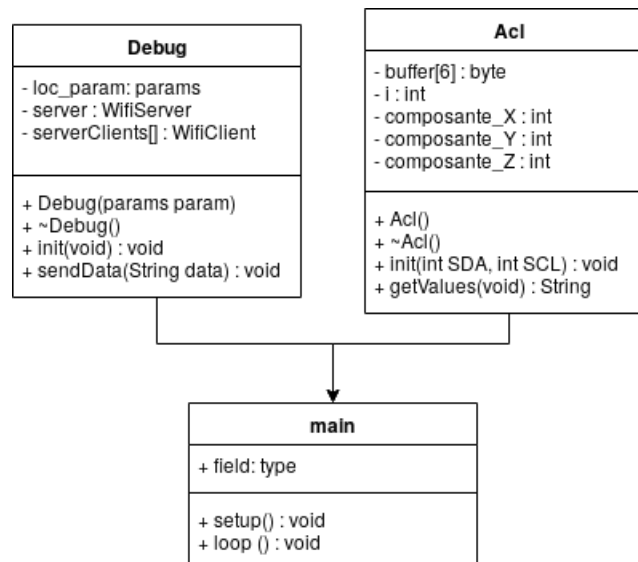


FIGURE 1.12 – Diagramme de classes - livrable 1

### 1.5.2 Classe Debug

Cette classe permet de faire le lien entre le microcontrôleur et l'extérieur. En paramètre de son constructeur, on peut indiquer si on veut que les données soient envoyées par WiFi, par liaison série ou les deux. Cela conditionne les initialisations et le canal sur lequel envoyer les données provenant des capteurs.

Ensuite la méthode `init()` permet d'initialiser la liaison série et/ou de créer un réseau WiFi et un serveur telnet permettant ensuite de pouvoir se connecter à celui-ci. Le hotspot est protégé par un mot de passe.

Une dernière méthode "`sendData(String data)`" permet d'envoyer une chaîne de caractères sur le port série, le WiFi ou les deux suivants le choix effectué auparavant.

### 1.5.3 Classe Acl

Cette classe fait office de pilote pour le composant ADXL345. Il permet d'initialiser ce dernier et de récupérer grâce à la méthode "getValues()" les accélérations en X, Y et Z sous forme d'une chaîne de caractères. Lors de l'initialisation, nous pouvons choisir quelle sensibilité doit avoir l'accéléromètre (2, 4, 8, 16g).

### 1.5.4 Classe main

C'est la classe principale, avec la boucle principale. Elle permet d'initialiser les objets, de récupérer les informations des capteurs et de les envoyer soit au port série pour le débogage soit en WiFi pour les tests.

Le code a été développé en C++ avec des librairies Arduino sur l'IDE platformIO. Cela permet d'être adaptable facilement sur tout type de microcontrôleur ou SoC.

### 1.5.5 Séquences livrables partie 2

Dans cette partie, nous allons voir quelques diagrammes de séquences sur l'utilisation du produit final. Pour commencer, le démarrage et la première configuration du produit.

#### 1.5.5.1 Premier démarrage

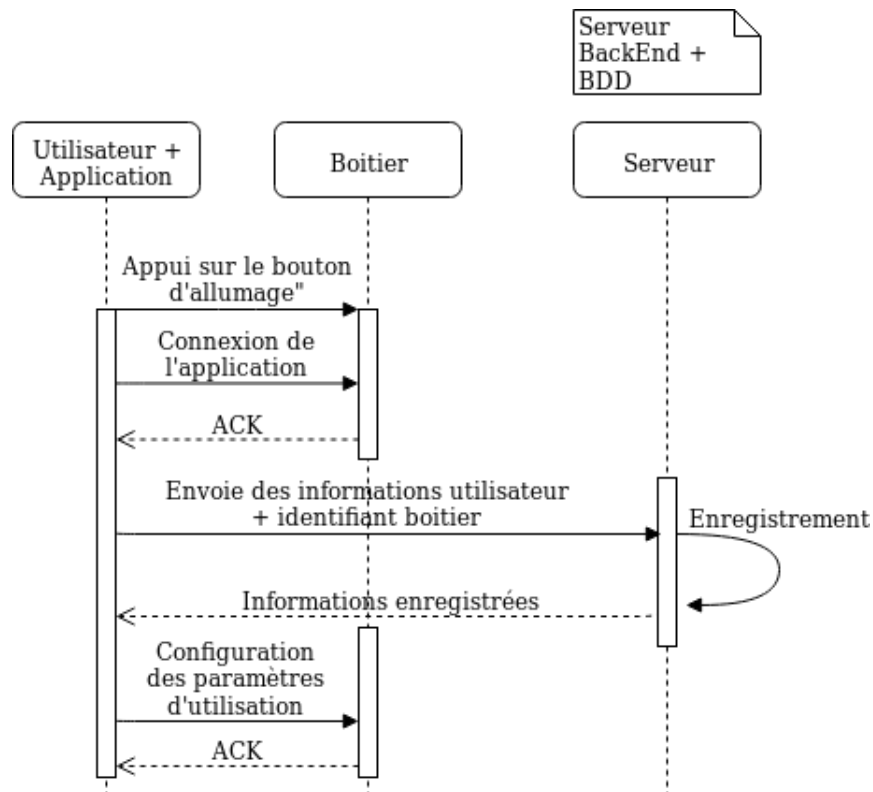


FIGURE 1.13 – Diagramme de séquence - premier démarrage

L'utilisateur aura une application sur son téléphone permettant de configurer son boîtier et ses paramètres utilisateurs ainsi que de visualiser ses parcours. L'application sera connectée en Bluetooth au boîtier. Il pourra configurer la confidentialité de ses parcours, le type de fonctionnement qu'il veut avoir avec le boîtier. S'il veut que les informations soient envoyées à intervalles réguliers, ou s'il faut stocker un certain nombre de données avant d'envoyer...

### 1.5.5.2 Utilisation courante

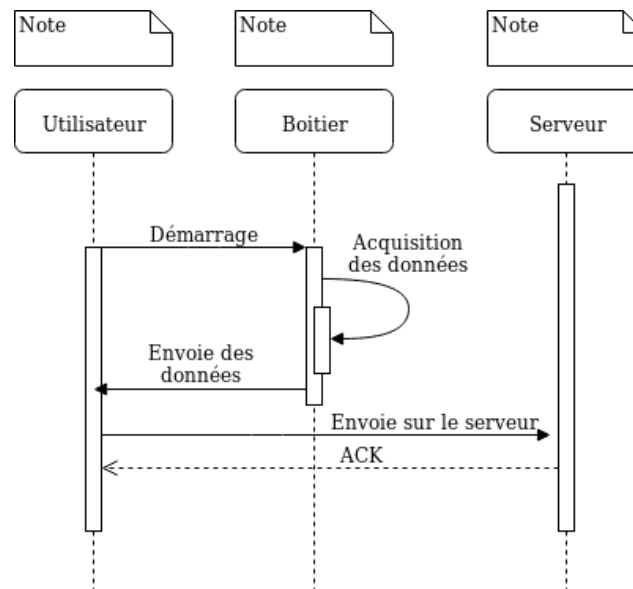


FIGURE 1.14 – Diagramme de séquence - Utilisation normale

Le boîtier peut envoyer ses données par le biais du téléphone et de sa connectivité à internet (3G, 4G) au serveur pour qu'il stocke le parcours et qu'il puisse partager l'état de la route aux autres usagers du système.

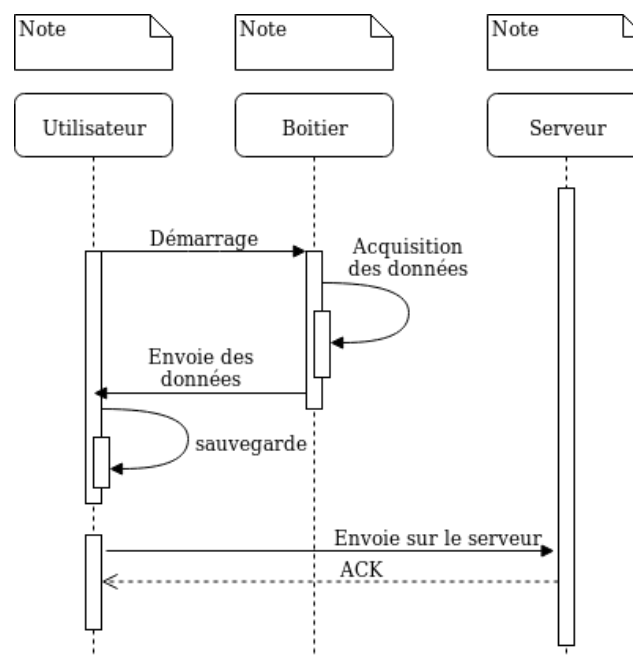


FIGURE 1.15 – Diagramme de séquence - Utilisation normale sans connexion

Si le téléphone n'a pas accès à internet alors l'application sauvegarde les données pour les envoyer lorsque la connexion sera rétablie ou si l'utilisateur se connecte à un réseau WiFi.

Un autre type de fonctionnement pourrait être possible où le boîtier serait autonome.

### 1.5.5.3 Boîtier autonome

En effet l'ESP32 est un module WiFi et Bluetooth. Si le boîtier scanne les réseaux WiFi et lorsqu'il détecte un réseau ouvert comme c'est le cas pour les freebox ou les réseaux publics, il se connecte et envoie directement les données au serveur. l'utilisateur n'a alors plus besoin d'avoir son téléphone sur lui ou d'avoir un forfait avec internet.

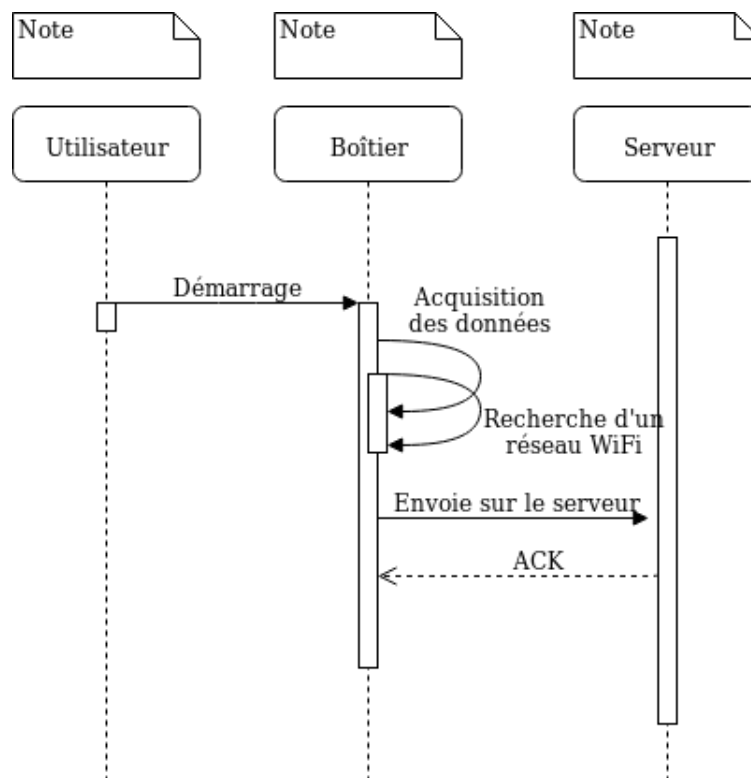


FIGURE 1.16 – Diagramme de séquence - Boîtier autonome

Tous ces diagrammes de séquences représentent au final dans le boîtier, une machine d'états pour acquérir les données de vibrations, la position GPS, la connexion Bluetooth ou la recherche d'un réseau WiFi, le stockage des données et la mise en forme de toutes ces données.



## REFERENCES

Measurement Specialties Inc, Piezo film sensors [document technique] disponible sur :<  
<https://www.sparkfun.com/datasheets/Sensors/Flex/MSI-techman.pdf>> [Consulté  
le 02 Janvier 2019]

Mechatronics Hub, Department Of Mechatronics Control Engineering, 2016. *Study of Piezoelectric Sensor Using Arduino & MATLAB*, [en ligne] disponible sur : Cliquez ici [Consulté  
le 15 Décembre 2018].

Dimension Engineering *A beginner's guide to accelerometers* [en ligne] disponible sur :  
<<https://www.dimensionengineering.com/info/accelerometers>> [Consulté le 10 Dé-  
cembre 2018].