

Julien ALLART | Paul VERNIER | Tanguy VIEUBLED  
Projet IMA3 - IMA4

## Rapport de projet

# Robe et pantalon pour lutter contre le harcèlement



Alexandre BOE | Xavier REDON | Thomas VANTROYS  
2019 - 2020



# Sommaire

<b>Introduction</b>	<b>3</b>
<b>I- Objectif du projet</b>	<b>4</b>
<b>II- Cahier des charges</b>	<b>5</b>
<b>III- Etat du projet aux semestres précédents</b>	<b>6</b>
<b>IV- Notre réalisation au S8</b>	<b>8</b>
Changement & reprise	8
Programmation	9
Algorithme arduino	9
Méthode de téléversement	10
Application	11
Création des cartes électroniques	13
Identification des besoins	13
Conception des schémas électriques	14
Création de composants	15
Routage des PCBs	16
Intégration sur le prototype	17
<b>Bilan &amp; Perspectives</b>	<b>19</b>
<b>Sources</b>	<b>20</b>

# Introduction

Dans le cadre de notre formation en Informatique Microélectronique et Automatique du cycle ingénieur à Polytech Lille, nous avons travaillé pendant trois semestres sur un projet. Ce projet a plusieurs objectifs. Un des objectifs est de nous permettre d'une façon ludique de mettre en application les connaissances que nous pouvons voir pendant nos cours. Aussi, nous avons pu développer les compétences en gestion de projets et en travail d'équipe ce qui est primordial pour notre vie professionnelle future.

Ce projet entre dans le cadre de la lutte contre le harcèlement, plus particulièrement contre le harcèlement physique ou bien contre les comportements qui peuvent mener à ce type de harcèlement. Le harcèlement est un phénomène très largement répandu dans notre société, particulièrement dans les soirées ou dans les transports en commun par exemple. Certaines personnes prétendent qu'il ne serait pas possible d'éradiquer ce phénomène. Nous souhaitons concevoir un dispositif permettant de supprimer ce phénomène ou tout du moins d'identifier les zones où il est plus fréquent pour apporter une réaction.

Le but de ce projet est de réussir à concevoir un système permettant de détecter sur n'importe quel type de personne les endroits où on la touche. Bien évidemment, notre prototype devra fonctionner en temps réel et être léger et aussi discret que possible.

# I- Objectif du projet

Ce projet a pour objectif de lutter contre le harcèlement physique à l'encontre des hommes ou des femmes. Après une étude sur l'état de l'art, nous avons vu qu'il n'existait pas réellement de système permettant de lutter contre le harcèlement. Une des solutions serait d'engager un garde du corps. Mais cette solution ne permet pas de faire de rencontre mais a pour avantages d'être efficace. Nous avons pu voir que Schweppes a ouvert une campagne pour dénoncer les attouchements des femmes lors de soirées. Leur système permet d'identifier les zones où il y a eu contact mais il n'y a pas d'intervention possible.

En combinant ces deux concurrents, nous pouvons discerner les différents objectifs de ce projet.

- ☐ L'un des premiers correspond à la conception d'un dispositif permettant de visualiser les zones du corps touchées par une tierce personne.
- ☐ Ensuite, nous devons être capable de traiter les données en temps réel.
- ☐ Et pour finir, le dispositif doit pouvoir gérer l'analyse et l'envoi de messages de danger.

Avant de commencer à programmer quoi que ce soit, nous allons définir notre cahier des charges et une liste de matériel nécessaire à l'élaboration de notre projet anti-harcèlement

## II- Cahier des charges

Etablir un cahier des charges est essentiel dans n'importe quel projet. En effet, ce document va nous permettre d'encadrer un projet en partant des besoins d'un client jusqu'à la remise du projet à ce client. Cet outil est nécessaire car il va pouvoir de définir clairement les besoins et les fonctions du projet.

Nous avons tout d'abord élaboré notre cahier des charges fonctionnel en se focalisant sur les besoins fonctionnels et comment notre solution y répond. Et à partir de ce premier cahier des charges, nous en avons développé un second qui s'intéresse aux exigences et aux contraintes techniques du dispositif.

Les besoins fonctionnels de ce projet sont les suivants :

- ☐ Dispositif sensoriel.
- ☐ Informations en temps réel.
- ☐ Développement d'une interface.
- ☐ Communication entre le dispositif et l'interface.
- ☐ Gestion d'un message ou d'un appel de danger et des numéros à prévenir.

A partir de ces besoins, nous avons pu affiner ce cahier des charges avec les contraintes techniques qui sont :

- ☐ Utilisation d'un grand nombre de capteurs.
- ☐ Programmation d'un algorithme de traitement des données.
- ☐ Conception d'une application en Java.
- ☐ Création d'un répertoire.
- ☐ Utilisation de la communication Bluetooth entre le dispositif et l'application.
- ☐ Légèreté et discrétion.

A l'issue de ce projet, nous serons capables de présenter un prototype fonctionnel et une application avec la capacité d'identifier les zones du corps touchées et aussi de prévenir quelqu'un du danger.

### III- Etat du projet aux semestres précédents

Nous avons commencé ce projet au S6 de notre cursus. Lors de ce semestre, nous avons établi un premier cahier des charges, analysé les concurrents possibles, décrit un scénario d'usage possible. Nous avons développé un premier prototype avec deux modèle de capteurs de force différents. Ces capteurs sont différents par leurs surfaces mais aussi par leurs sensibilités.

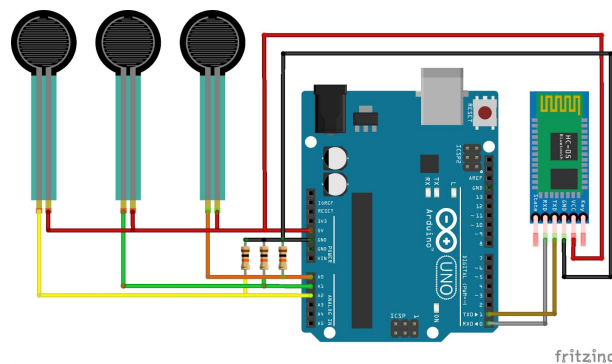


Fig. 1 - Schéma de montage

Nous avons adapté ce premier prototype à la détection des touches sur un avant-bras. Nous avons disposé deux petits capteurs sur l'intérieur du bras et le gros sur l'extérieur. Avec une carte Arduino, nous avons pu tester nos capteurs et déterminer le seuil de discrétion. Une application développée avec MIT, nous a permis juste après de voir les zones touchées sur un smartphone.

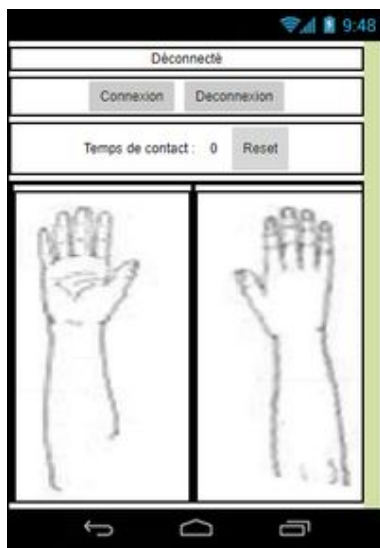


Fig. 2 - Vue de l'application avec MIT

En fin de compte, ce premier prototype ne nous a pas donné satisfaction. Le traitement des données par MIT est très lourd et les capteurs ne permettent pas la distinction lorsqu'on se touche et quand quelqu'un nous touche. De plus, pour des petits contacts discrets, le prototype ne les détecte pas toujours.

Nous avons donc commencé le semestre 7 à rechercher une nouvelle méthode de détection et aussi à concevoir une nouvelle application.

Ces recherches nous ont permis de créer des capteurs capacitifs dits "tactils". On utilise le même principe que sur nos smartphones où le corps humain vient de décharger dans une capacité. Grâce à cela, on arrive à avoir un système beaucoup plus précis qu'avec nos capteurs de force.

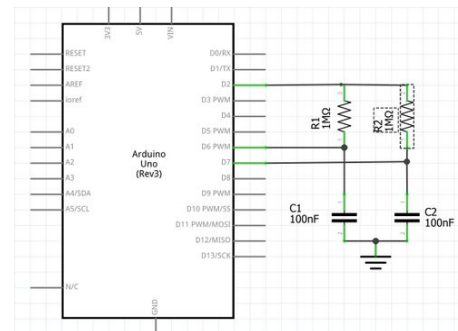


Fig. 3 - Schéma pour nos capteurs 'tactils'

Ensuite, on s'est demandé comment adapter cette méthode sur notre prototype sachant que notre corps peut aussi être considéré comme une touche avec un capteur. On a choisi de concevoir un nouveau prototype avec une double couche. Sur la première qui sera en contact avec l'extérieur, on va venir coudre un fils conducteur qui sera notre moyen de détection.



Fig. 4 - Rendu du prototype en fin du S7 avec application

Pour finir ce semestre, on a su développer une nouvelle application en Java grâce à la suite Android Studio. On a débuté par établir la connexion Bluetooth avec le module HC-05. Ensuite, on a cherché à visualiser les touches. Pour finir, on a continué en aménageant les vues et en ajoutant quelques fonctionnalités de l'application précédentes.

# IV- Notre réalisation au S8

## Changement & reprise

A la reprise du semestre 8, nous avons pris la décision de concevoir notre propre PCB. Nous avons choisi de garder en plus des capteurs capacitifs, les capteurs de force. En effet, ces capteurs vont nous permettre de savoir si on est en présence d'un contact amical ou d'un cas de harcèlement, voire d'agression. Nous allons donc disposer quelques capteurs sur notre prototype sur les zones qui nous intéressent.

En phase de conception, nous avons fait la liste des besoins de cette carte.

- ❑ 14 + 2 pins digitales pour les capteurs 'tactiles'.
- ❑ 6 pins analogiques pour les capteurs de force.
- ❑ 2 pins digitales pour la liaison bluetooth.
- ❑ 1 pin digitale pour le module vibrant.

Avec ses besoins, nous sommes partis sur l'idée d'utiliser un Atmega 2560. Après discussion avec nos enseignants, nous avons vu que la longueur des câbles pouvait nous poser des soucis sur les valeurs des impédances. Pour palier à ce problème, nous avons découvert qu'on pouvait mettre en place un système maître - esclave.

Cette solution va nous permettre d'une part de réduire les impédances et d'autre part de pouvoir augmenter considérablement le nombre de capteurs. La carte maître va avoir un rôle de récepteur. Il va gérer la liaison Bluetooth et l'envoi des messages avec l'application. Les cartes esclaves disposeront de tous nos capteurs (tactile + force). Chaque esclave transmettra une trame avec les informations sur l'état des capteurs en temps réel et le programme sur le maître va nous permettre de savoir quelle zone est touchée et stockera cette information pour savoir s'il s'agit de harcèlement ou pas.

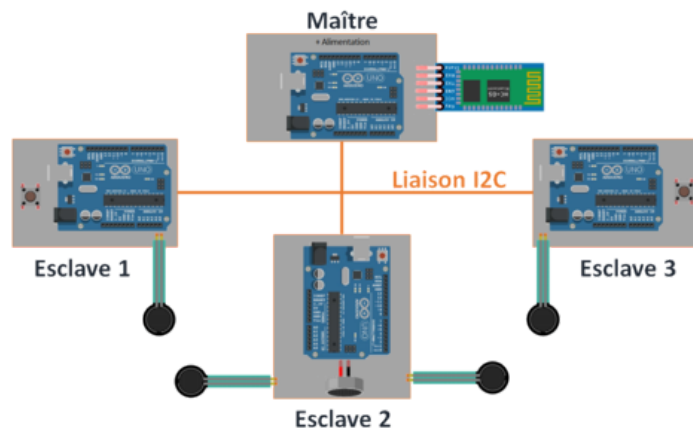


Fig. 5 - Représentation du système maître - esclave



Nous avons décidé d'assigner chaque esclave à une zone du corps et de centraliser la carte maître.

## Programmation

### Algorithme arduino

Nous avons eu comme idée d'implanter un système maître - esclave, pour cela on, connecte la carte maître et les cartes esclaves avec la liaison I2C disponible sur les atmega328p. Chaque carte esclave dispose de 1 à 6 capteurs tactiles ainsi que de 0 à 2 capteurs de force. Pour initialiser ce bus I2C, on utilise les fonctions de la bibliothèque I2C. La fonction `Wire.begin()` initialise la liaison et prend seulement pour la carte esclave son adresse en paramètre. Dans le code du maître, `Wire.requestFrom()` permet de demander aux esclaves l'état des capteurs. Et dans le code esclave, on retrouve la fonction `Wire.onRequest()` qui fait appeler à notre fonction de détection quand le maître le demande.

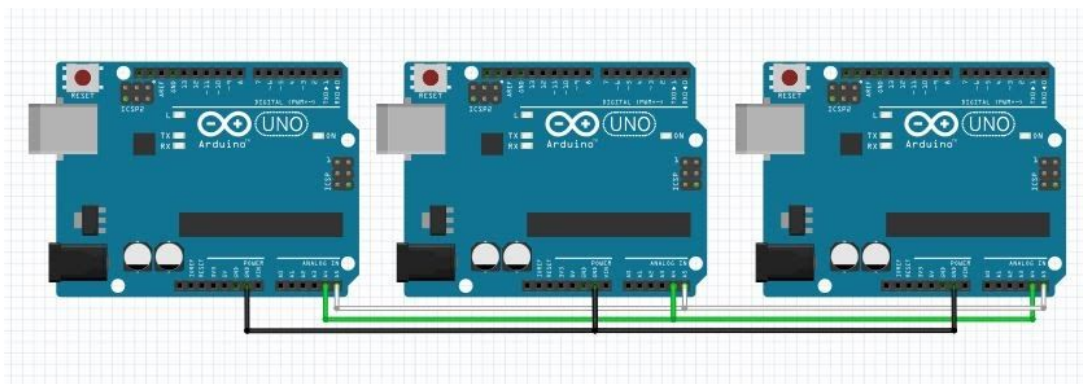


Fig. 6 - Représentation du système maître - esclave

Le maître envoie une requête à chacun des esclaves pour obtenir l'état des capteurs et il envoie ensuite les données via une liaison bluetooth vers le téléphone qui dispose de l'application. En détail, chaque esclave envoie une trame de 2 octets l'un après l'autre. L'octet de poids faible correspond la valeur des capteurs tactiles tout ou rien et l'octet de poids fort correspond aux valeurs analogiques des 2 capteurs de force allant de 0 à 15.

Ensuite le maître récupère les trames une par une et concatène les valeurs des capteurs tactiles entre-elles et fait de même pour les capteurs de forces. Ainsi nous avons deux trames, l'une avec les capteurs tactiles et l'autre avec les capteurs de forces.

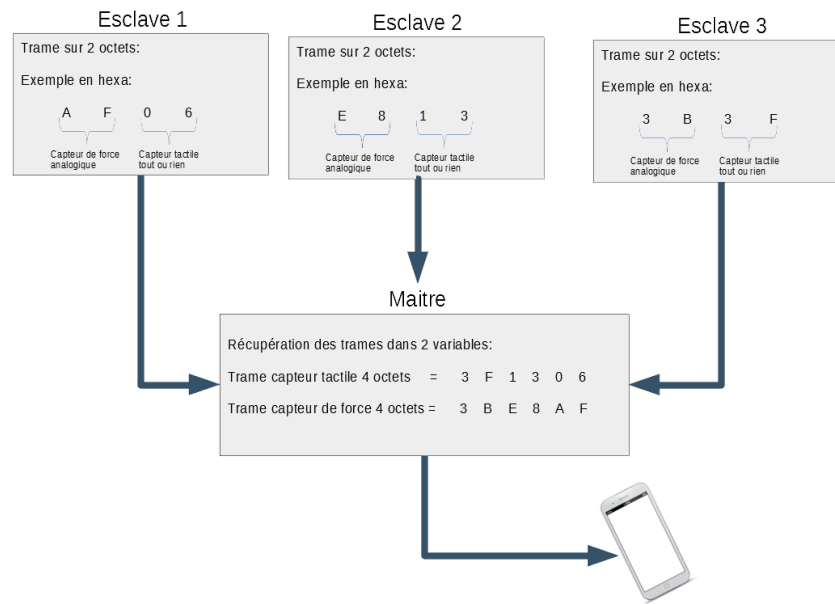


Fig. 7 - Trame Maître/Esclave

Nous envoyons ensuite le caractère 'f' qui est utilisé comme caractère de début de trame, ce caractère est suivi de l'état des capteurs de touche. Ensuite, on a une deuxième partie commençant par le caractère 'd' qui est suivi des valeurs des capteurs de force comme illustré à côté.

COM7	
340480	#0 : d10000101
340480	#0 : d10000101
340480	#0 : d10000101
340480	#0 : d10000101
340480	#0 : d10000101
340480	#0 : d10000101
340480	#0 : d10000101
340480	#0 : d10000101
340480	#0 : d10000101
340480	#0 : d10000101
340480	#0 : d10000101
340480	#0 : d10000101
340480	#0 : d10000101
340480	#0 : d10000101
340480	#0 : d10000101
340480	#0 : d10000101
340480	#0 : d10000101

Fig. 8 - Trame Maître vers téléphone

## Méthode de téléversement

Un point que nous n'avons pas encore abordé concerne la méthode à employer pour pouvoir téléverser nos programmes. En effet, nous avons pris le choix de concevoir nos propres cartes et donc nous devons trouver une solution pour uploader nos codes sur ces dernières afin de rendre le système opérationnel.

Deux choix se sont présentés à nous.

- ❑ Le premier consiste simplement à mettre en place une prise USB.
- ❑ Le second à utiliser un protocole spécifique.

Par souci de discrétion, nous avons décidé d'utiliser un protocole. Nous avons choisi le protocole ISP2 qui consiste à utiliser un arduino classique comme programmeur. Il suffit simplement de changer le type de programmeur dans l'IDE arduino en allant dans [Outils], [Programmeur] et sélectionner "Arduino as ISP".

Les seuls pins nécessaires à l'utilisation de ce protocole sont :

- ☐ MISO
- ☐ MOSI
- ☐ SCK
- ☐ Reset

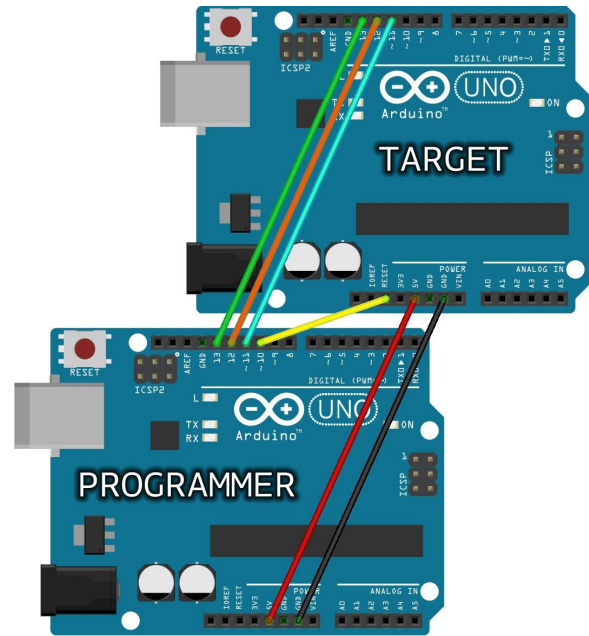


Fig. 9 - Schéma principe ISP

Par contre, nous avons remarqué que les pins Rx et Tx doivent être libre afin de pouvoir téléverser sans encombre. Pour cela, nous avons ajouté un commutateur deux positions [on, off]. Ainsi il suffit de le mettre sur [Off] quand on vient téléverser les programmes.

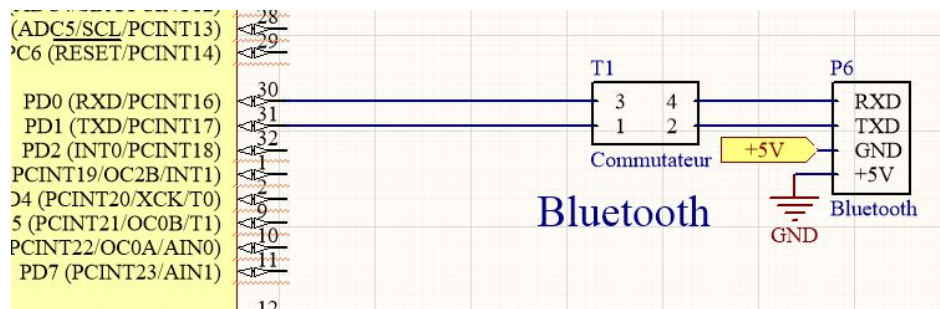


Fig. 10 - Vue schématique avec commutateur et connecteur Bluetooth

## Application

A partir du semestre 7, nous avons choisi de créer notre propre application android avec le logiciel Android Studio. Nous avons, dans un premier temps, créé une application capable d'afficher en temps réel les différentes zones du vêtement qui étaient touchées. Par la suite, nous avons ajouté diverses options, telles que la possibilité de contacter de l'aide via SMS, ou appels, ou encore le fait de pouvoir visualiser le temps de contact pour chaque zone de corps, ainsi que le temps de contact cumulé.

Comme indiqué précédemment, pour le semestre 8, nous avons choisi, d'ajouter, en plus des capteurs de touche, les capteurs de force. Ces derniers nous permettront de distinguer plus précisément, les contacts "amicaux" des contacts de harcèlement voire d'agression.

La carte maître du système maître - esclave envoie par liaison bluetooth, 2 trames, l'une d'elle contenant les informations en temps réel des capteurs de touche, et l'autre les informations concernant les capteurs de force. Une fois dans l'application, nous découpons ces deux trames et nous récupérons, par l'intermédiaire de quelques algorithmes, les valeurs de chaque capteur.

La valeur des capteurs de touche est une variable booléenne, la valeur 'vraie' indiquant que le capteur détecte une touche. Pour les capteurs de force c'est différent. Chacun d'entre eux a une valeur entre 0 et 15, plus cette valeur est élevée, plus la pression captée par le capteur de force est intense. Cela nous permet donc d'avoir en temps réel une information sur les zones touchées, ainsi que sur l'intensité de ce contact.



*Fig. 11 - Visualisation d'un contact sur l'application*

Pour l'affichage de ces informations nous avons décidé d'afficher, sur l'application, en rouge les zones à l'avant du vêtement et en orange celles à l'arrière, l'information des capteurs de force venant faire varier l'opacité de l'affichage de chaque zone.

Comme présenté sur le visuel ci-contre, lorsque l'on touche simplement le capteur de touche en appuyant très peu dessus, la zone s'affiche avec une couleur rouge plus faible que si l'on appuie fort.

Nous avons ensuite ajouté des fonctionnalités en lien avec l'option d'appel et d'envoi de SMS, la fonctionnalité principale est le fait de pouvoir réaliser un 'code' lorsque la personne se sent en danger. Ce code consiste à toucher le capteur de touche situé sur l'avant bras droit 5 fois de suite sans que le temps de chaque touche n'excède 0.5 sec. Cela permettrait à la personne de pouvoir appeler de l'aide en toute discrétion. Une fois le code effectué, le téléphone de la personne enverra un SMS, ou appellera le numéro préalablement enregistré dans la partie de configuration dans l'application. Pour avertir qu'un appel ou un SMS va bientôt s'envoyer, la téléphone va vibrer discrètement, afin d'être averti que le code a bien fonctionné, et, dans le cas

où la personne n'a pas fait le code de intentionnellement, cela permettra d'annuler l'appel ou l'envoi du SMS.

De plus, lorsqu'un capteur de force indique une valeur jugée trop élevée, l'application lancera automatiquement l'appel et l'envoi de SMS, en activant la vibration du téléphone pour avertir. Cependant, un problème se pose, si la personne n'a pas son téléphone directement avec elle, et que ce dernier est par exemple, dans un sac, elle ne pourra pas être avertie du lancement d'un appel ou d'un SMS. C'est pourquoi nous avons choisi d'intégrer un module vibrant directement sur le vêtement, ainsi dès qu'un appel se lance, ou qu'un SMS va s'envoyer, le module vibre sur le vêtement, de la même manière que le téléphone, prévenant ainsi l'utilisateur pour qu'il puisse agir en conséquence.

## **Création des cartes électroniques**

Pour concevoir nos PCB, nous avons utilisé le logiciel Altium Designer que nos enseignants nous ont présenté au semestre 7. Grâce à ce logiciel de CAO, nous avons pu réaliser schémas, les bibliothèques de composants personnelles ainsi que les PCB. Au préalable, nous avons commencé par identifier nos besoins. Puis, nous avons fait nos schémas électriques pour ensuite arriver à obtenir nos PCB.

### **Identification des besoins**

De manière générale, nous avons besoin pour les deux cartes d'une méthode permettant de téléverser nos programmes (ISP2), un protocole de communication maître-esclave (I2C) et surtout un microcontrôleur.

Pour choisir notre microcontrôleur, nous avons cherché à voir le nombre de pins digitales et analogiques nécessaires. Au total, nous avons besoin de six pins analogiques pour les capteurs de force et de 18 pins digitales pour les capteurs tactiles, le module vibrant et la communication Bluetooth et I2C. Grâce à cela, nous avons pu choisir un microcontrôleur simple et très répandu. Nous sommes partis pour utiliser un Atmega 328p qui suffit amplement.

Là où les besoins des cartes changent correspondent à leur utilisation. Pour la carte maître, il n'a pas besoin de capteur car elle va nous servir seulement de gestionnaire comme on a pu le décrire précédemment. Sur cette carte, on va trouver en plus la liaison Bluetooth, un régulateur de tension pour l'alimentation du système avec une batterie externe.

Pour la carte esclave, nous allons placer nos capteurs tactiles et les capteurs de force. En plus, on vient ajouter un module vibrant. Pour résumer, nous allons retrouver les éléments suivants sur nos cartes :

	Carte maître	Carte esclave
Alimentation extérieure	✓	X
Régulateur de tension	✓	X
Alimentation +5V	✓	✓
Bluetooth	✓	X

	Carte maître	Carte esclave
ISP2	✓	✓
I2C	✓	✓
Capteurs tactiles	X	✓
Capteurs de force	X	✓
Module vibrant	X	✓

## Conception des schémas électriques

Maintenant que nos besoins ont été identifiés, nous avons pu commencer à concevoir les différents schémas. Grâce à nos tests sur la breadboard, on sait déjà comment on va devoir câbler. Nous avons donc besoin des pins spécifiques suivantes :

- ❑ Le Bluetooth, nous avons besoin des pins Rx, Tx, +5V et GND.
- ❑ Le protocole ISP2, nous avons besoin des pins MOSI, MISO, SCK, +5V et GND.
- ❑ Le protocole I2C, nous avons besoin des pins SDA, SCL, +5V et GND.

Pour ajouter les capteurs, nous avons regardé les schémas du microcontrôleur afin de rester sur les mêmes ports. On ajoute ensuite des composants spécifiques comme le quartz et les condensateurs de découplage sur les différentes pins d'alimentation. Pour finir, après avoir câblé proprement et utiliser des labels pour rendre les schémas les plus lisibles possible, on obtient ces schémas électriques.

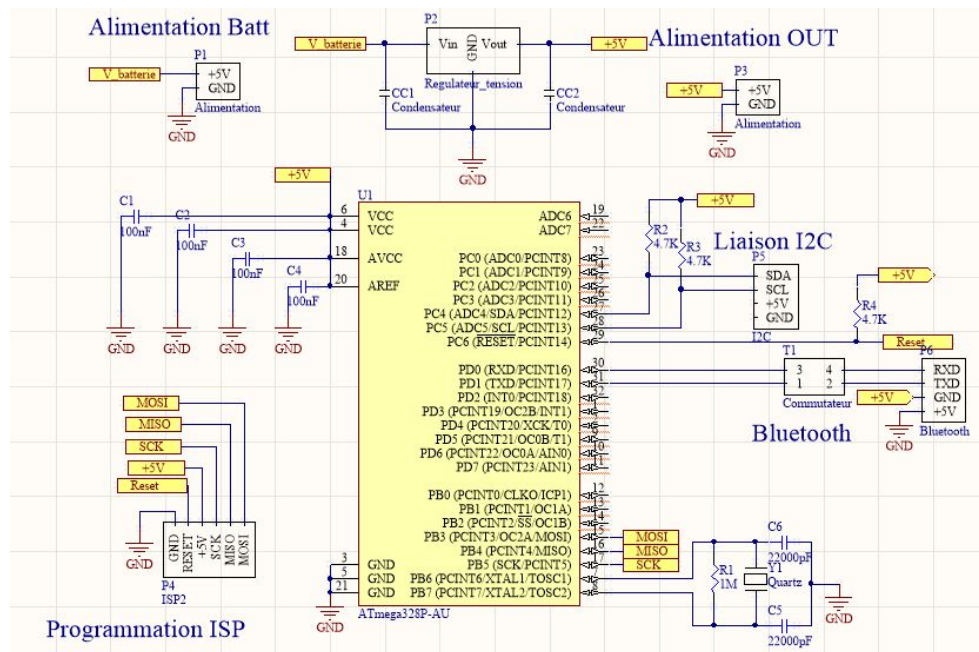


Fig. 12 - Schéma électrique de la carte maître



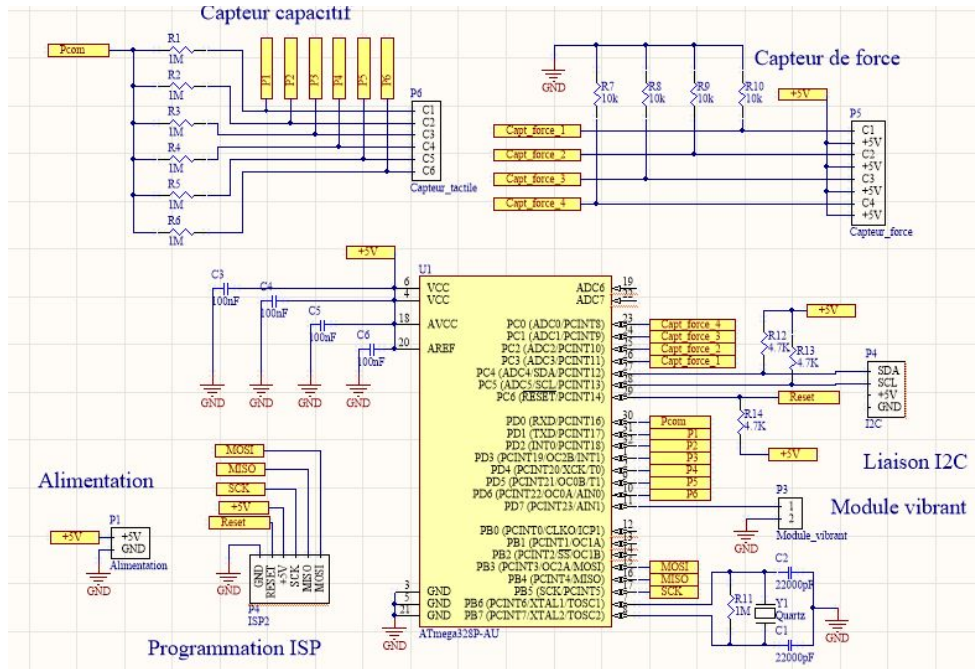


Fig. 13 - Schéma électrique de la carte esclave

## Création de composants

Certains composants que nous avons choisi dans notre liste de matériel n'étaient pas présents dans les bibliothèques du logiciel Altium Designer. Après quelques recherches et des bibliothèques incomplètes, nous avons pris la décision de concevoir nous-mêmes les composants tout en respectant les dimensions données par les fabricants dans les datasheets. On trouve le quartz, le régulateur de tension et tous nos connecteurs.

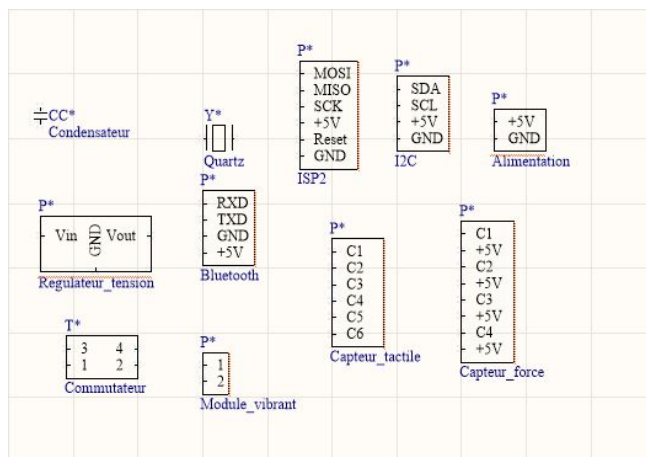


Fig. 14 - Schéma électrique de la carte esclave

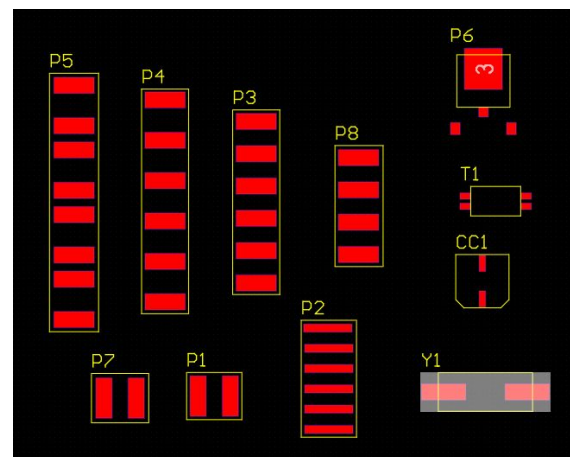


Fig. 15 - Schéma électrique de la carte esclave

## Routage des PCBs

Pour commencer le routage, nous avons commencé par établir les règles de design. Cela va nous permettre d'ajouter des contraintes à notre dessin qui vont nous permettre de respecter les règles de conception en suivant le document présent sur le wiki de l'intranet.

Ensuite, on vient positionner les composants de sorte à limiter les routes. Pour les condensateurs de découplage, on vient les ajouter aux plus près des pins d'alimentation de l'Atmega 328p. Pour le reste, nous avons tracé sur papier pour avoir une idée de la position des autres composants ainsi que les liaisons. On n'oublie pas d'ajouter un plan de masse afin d'éviter les parasites et de référencer ce plan avec la bonne désignation. Pour la face Top, on choisit GND et pour le bottom, on sélectionne +5V.

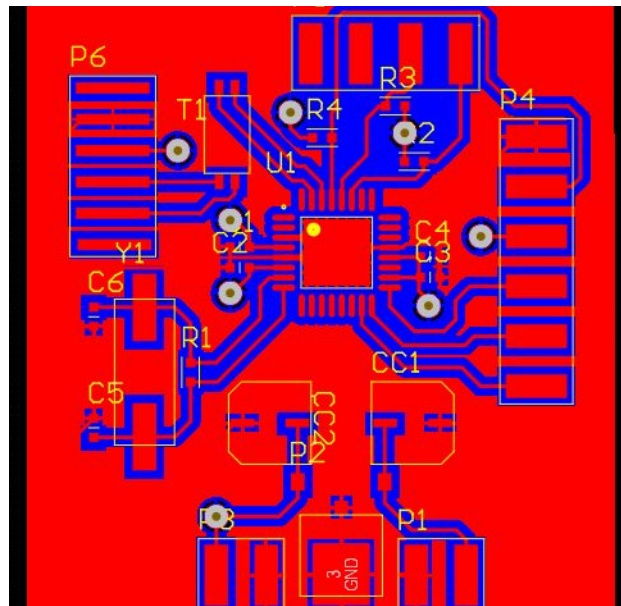


Fig. 16 - Vue PCB de la carte maître



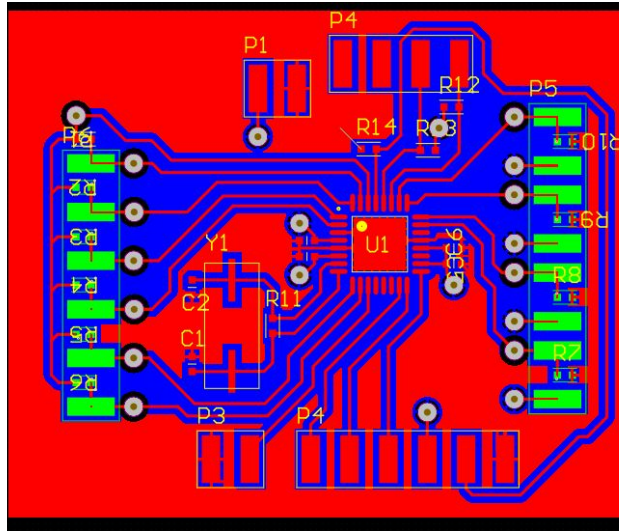


Fig. 17 - Vue PCB de la carte esclave

Pour les réaliser, nous avons deux options soit les faire par gravure chimique soit avec une gravure mécanique.

Une fois tous les éléments de notre commande reçue et nos PCB réalisés, il suffira de venir souder nos composants sur nos cartes, téléverser nos codes et connecter les différents capteurs sur les cartes esclaves.

## Intégration sur le prototype

Après avoir choisi de changer notre système au profit d'un dispositif maître - esclave, nous avons mis en place les différentes zones du vêtement. On a donc pu définir 16 zones comme on peut le voir sur le schéma suivant.



Fig. 18 - Schéma avec zone de détection

Grâce au fil conducteur que nous avons pu obtenir à l'école, nous avons pu coudre directement ce fils sur le vêtement test. Nous avons testé de coudre en suivant un motif en forme de spiral mais nous avons remarqué que ce motif n'est pas très optimal car il laisse de nombreuses zones vides. Or ces zones peuvent justement nous dire si on est en présence du harcèlement. Nous avons changé de motifs pour coudre avec une forme rectangulaire qui offre plus de surface.



*Fig. 19 - Vêtement avec les zones*

On a pu tester ce T-shirt 'test' sans le porter car on n'a pas réalisé de doublure dessus. On a constaté de légères perturbations mais rien de significatif car même lors de nos tests précédents, on a aussi observé des variations sur les valeurs des capteurs. Par contre, notre dispositif est opérationnel. Par ailleurs, il sera nécessaire de réduire l'espace entre les fils et surtout d'avoir un écartement homogène partout.

Pour l'identification du harcèlement, nous avons choisi de rajouter sur l'application une fonctionnalité permettant d'enregistrer par l'utilisateur des combinaisons de touches qu'il considère comme harcèlement en fonction de la tolérance de la personne. Il y a bien sûr des combinaisons préétablies pour des zones considérées comme intimes. Nous utilisons également les capteurs de force pour prévenir un harcèlement. Ces capteurs indiquent une force allant de 0 à 15, nous considérons un harcèlement violent à partir de 8 indiquant que l'utilisateur est en danger.

Il nous reste à récupérer notre commande pour réaliser nos cartes, coudre le fil et réaliser la doublure avec des poches pour les PCB, et tester la dernière version de ce prototype sur Paul.

# **Bilan & Perspectives**

Pour conclure ce rapport, nous avons pu développer une solution permettant la détection des touches sur une personne. Au cours de ce semestre, nous avons été amené à travailler sur différents aspects. Grâce à la pluralité disciplinaire de notre formation, cela ne nous a pas gêné mais au contraire renforcer notre travail de recherche et de développement.

Lors de ce semestre, nous avons pu prendre les dernières décisions de ce projet. La plus radicale de nos choix s'est portée non pas sur la façon de détecter mais sur l'aspect matériel. Nous avons décidé de mettre en place une méthode de maître - esclave pour réduire d'un part les risques de perturbations et pour augmenter grandement le nombre de capteurs d'autre part. Dès ce que ce système a été mis en place, nous avons tout fait pour que nos tests de détection soient concluants et qu'il n'y avait pas d'erreurs. Dans un temps, nous avons développé nos propres cartes électroniques afin de rendre notre dispositif plus pratique, adapté à notre besoin et exclusif. Pour finir, nous avons continué de développer notre application en ajoutant quelques fonctionnalités mais aussi lui donnant un aspect esthétique.

Malheureusement, suite à l'actualité, nous n'avons pas pu terminé de mettre en place la version finale de ce prototype de vêtement et de pouvoir le tester en condition réelle.

Pour conclure ce projet d'une façon plus général, nous avons particulièrement bien aimé à travailler dessus. Il nous a permis de renforcer nos compétences de recherches et de développement d'un point de vue électronique et informatique. D'autre part, nous avons bien su gérer le partage des tâches et le travail d'équipe en permettant à chacun de pouvoir s'exprimer et d'argumenter sur ses idées. Ce projet est un vrai atout dans notre vie universitaire car il nous a permis de voir comment se déroule dans son ensemble la conception d'un nouveau "produit". Nous avons débuté avec une problématique pour ensuite élaborer un cahier des charges et aboutir au développement d'un prototype fonctionnel.

# Sources

## Notre projet :

[projets-ima.plil.fr/mediawiki/index.php/IMA3/IMA4\\_2018/2020\\_P16](https://projets-ima.plil.fr/mediawiki/index.php/IMA3/IMA4_2018/2020_P16)

## Aide protocole ISP et I2C :

[forum.arduino.cc](https://forum.arduino.cc)

## Création des PCB :

[altium.com/fr](https://altium.com/fr)

[wikiportal.polytech-lille.fr/mediawiki/index.php/IMA#IMA4](https://wikiportal.polytech-lille.fr/mediawiki/index.php/IMA#IMA4)

## Choix des composants :

[farnell.com](https://farnell.com)

[Mouser.fr](https://mouser.fr)

## Application :

[developer.android.com/studio](https://developer.android.com/studio)

[www.youtube.com/watch?v=IVE5ETGEoA4&](https://www.youtube.com/watch?v=IVE5ETGEoA4&)