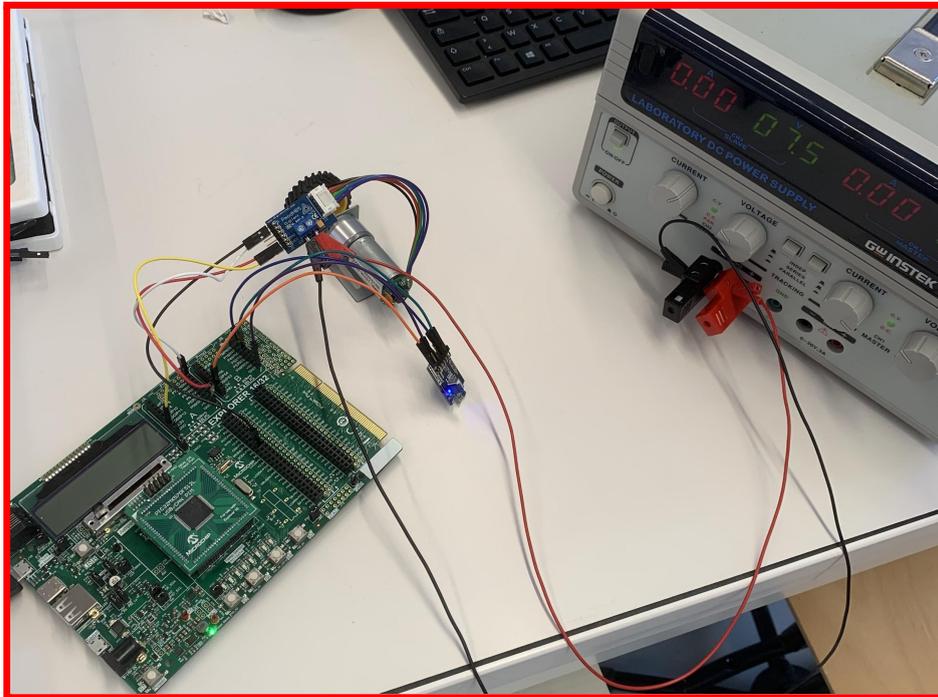


*Projet de Physique P6  
STPI/P6/2022 – 45*

**Initiation à l'informatique embarquée :  
Microprocesseur/Microcontrôleur et outils  
logiciels**



**Etudiants :**

**Noé PAQUIN**

**Hugo MATHIEU**

**Maxime LAURENT**

**Victorin TURNEL**

**Tristan GOUBIN**

**Nathan TESSADRI**

---

**Enseignant-responsable du projet :**

**Richard GRISEL**



## INFORMATIONS SUR LE PROJET

Date de remise du rapport : **11/06/2022**

Référence du projet : **STPI/P6/2022 - 45**

Intitulé du projet : **Initiation à l'informatique embarquée : Microprocesseur / Microcontrôleur et outils logiciels**

Type de projet : **Expérimental et simulation**

Objectifs du projet :

**L'objectif principal de ce projet est de nous familiariser avec la conduite de projets en groupe, ainsi que de réaliser des simulations et des prototypes fonctionnels comportant de l'informatique embarquée. Ici , nous tenterons d'analyser toutes les facettes de l'informatique embarquée en réalisant une maquette d'un robot suiveur autonome permettant de transporter une charge comme une valise.**

Mots-clefs du projet : **informatique embarquée – simulation – modélisation – électronique**

## TABLE DES MATIÈRES

<b>Notations et acronymes</b>	<b>5</b>
<b>Introduction</b>	<b>6</b>
<b>Méthodologie / Organisation du travail</b>	<b>7</b>
<b>1 Analyse des besoins et travail préliminaire</b>	<b>8</b>
1.1 Analyse des besoins	8
1.2 Difficultés observées	8
1.3 Mise en place des solutions	9
<b>2 Construction et Tests</b>	<b>10</b>
2.1 Partie mécanique	10
2.1.1 Nécessité de cette partie	10
2.1.2 Conception préliminaire	10
2.1.3 Dimensionnement et modélisation	11
2.1.4 Simulation	12
a) Présentation de la simulation	12
b) Mise en place de la simulation	13
c) Résultats de la simulation	14
2.1.5 Montage de la maquette	18
2.2 Partie électronique	19
2.2.1 Étude des composants	19
2.2.2 Étude des branchements	21
2.2.3 Étude énergétique et alimentation des cartes et des moteurs	22
2.3 Partie informatique	23
2.4 Test Global	26
<b>Conclusions et perspectives d'amélioration</b>	<b>27</b>
<b>Bibliographie</b>	<b>28</b>
<b>Annexes</b>	<b>29</b>

## NOTATIONS ET ACRONYMES

IE – Informatique Embarquée

U - Tension

I - Courant

VCC - Voltage Common Collector

GND - Ground

BLE - Bluetooth Low Energy

tr/min - Tour par minute

## INTRODUCTION

Dans le cadre de notre formation ingénieur INSA, il nous est donné de réaliser un projet supervisé au sein de l'EC P6. Les sujets proposés sont multiples et variés et nous nous sommes orientés vers celui portant sur l'informatique embarquée.

Par groupe de 6 étudiants, nous devons étudier et concevoir une maquette ainsi qu'effectuer des recherches, expériences, analyses et calculs par rapport à un sujet précis. Dans notre cas, l'IE fut un choix rapide, car nous sommes tous passionnés par l'informatique. De plus, la perspective de réaliser un projet concret s'opposait bien à nos cours plutôt théoriques. Nous avons donc choisi le sujet s'intitulant « *Initiation à l'informatique embarquée : Microprocesseur / Microcontrôleur et outils logiciels* ».

Au fil des séances, nous nous sommes fixés comme objectif à atteindre de réaliser un prototype fonctionnel de robot permettant de suivre un utilisateur en transportant une charge à l'aide de l'accéléromètre d'un téléphone.

Nous allons vous présenter dans ce rapport les différentes étapes de la conception de ce projet. Pour cela, ce document suivra le plan suivant. Dans un premier temps, nous décrirons notre projet, son utilité et ses fonctionnalités. Ensuite, nous nous intéresserons à la réalisation de ce projet. Enfin, nous conclurons ce rapport.

## MÉTHODOLOGIE / ORGANISATION DU TRAVAIL

Lors des premières séances, M.Grisel nous a présenté les bases de la programmation en C embarquée dont nous aurons besoin pour notre projet.

Avant même de choisir un projet, nous avons discuté de la définition d’"informatique embarqué" :

Tout d'abord, nous avons le terme "informatique" qui fait clairement référence à de la programmation, du code et du traitement de données, il est donc clair qu'il est nécessaire d'en faire son étude. Cependant, nous nous sommes fait la remarque que le code est inutile s'il n'a pas de support qui l'utilise, il est alors nécessaire de traiter une partie électronique.

Le second terme "embarqué" nous a demandé plus de réflexion, nous n'arrivons pas à trouver le bon secteur d'étude. Nous avons finalement adopté une approche mécanique, le terme "embarqué" fait référence au support mobile qui transporte toute l'informatique et négliger son étude manquerait à une analyse complète de notre thème d'étude qui est "l'informatique embarquée"

Par la suite, nous nous sommes mis d'accord sur la répartition du travail. Nous avons décidé de travailler sur un robot suiveur et nous avons divisé le travail en 3 pôles : Électronique, Informatique et Mécanique. Les différentes tâches sont représentées par pôles dans le tableau ci-dessous :

Partie Électronique		Partie Informatique		Partie Mécanique	
Tristan Goubin	Nathan Tessadri	Hugo Mathieu	Maxime Laurent	Victorin Turnel	Noé Paquin
- Ajustement des moteurs - Branchements - Montage		- Gestion du programme - Communication Machine - Utilisateur		- Modélisation - Simulation - Maquette	

Finalement, tout au long de ce semestre, nous avons suivi une organisation pouvant être modélisée par le diagramme de Gantt suivant :



# 1 ANALYSE DES BESOINS ET TRAVAIL PRÉLIMINAIRE

## 1.1 Analyse des besoins

En vue du projet que nous avons entrepris, nous avons établi plusieurs objectifs qui sont les besoins que nous devons satisfaire. Les voici listés ci-dessous :

- Programmer en C embarqué
- Utiliser une source d'énergie électrique
- Pouvoir transporter des charges comme une valise
- Établir les modules et les moteurs que nous allons utiliser
- Récupérer les données de l'accéléromètre du téléphone de l'utilisateur par bluetooth pour le suivre
- Dimensionner correctement chaque élément
- Permettre la "transportabilité" de l'électronique

## 1.2 Difficultés observées

Une fois cette liste mise au point, il nous a fallu réfléchir à la faisabilité de chacun de ces points.

Au fil des séances, nous nous sommes rendus compte de la difficulté de récupérer les données de l'accéléromètre. En effet, malgré nos efforts, nous ne sommes pas parvenus à mettre au point un programme fonctionnel permettant de récupérer ces dernières.

Nous avons également réalisé qu'avec les matériaux et les moteurs dont nous disposions, transporter de lourdes charges ne serait pas possible. Nous avons donc dû réduire nos ambitions et limiter la charge maximale transportable par notre robot suiveur.

En termes de difficultés rencontrées durant ces semaines de travail, nous avons également dû composer avec des délais relativement courts par rapport à la charge de travail nécessaire afin de mener à terme notre projet initial. En effet, les 5 premières séances consacrées au projet ont été dédiées à la découverte et à la prise en main du langage C embarqué. Nous partions en conséquence avec un nombre de semaines réellement dédiées au développement même du projet réduit.

Notre manque d'expérience nous a aussi ralenti dans notre travail. En effet, nous avons a posteriori réalisé que nous avons manqué de communication. Les informations ne circulaient pas systématiquement dans l'ensemble du groupe, de ce fait, certaines personnes du groupe ont parfois gaspillé du temps à travailler sur des points du projet que nous avons modifiés, voire abandonnés.

Enfin, la dernière difficulté majeure que nous avons rencontrée était due au matériel ainsi qu'aux logiciels que nous avons à disposition. En effet, le logiciel MPLAB permettant de coder et de transférer les informations à la carte mère était relativement ancien, ce qui nous a par conséquent particulièrement compliqué la tâche du groupe en charge de la partie informatique du projet.

### 1.3 Mise en place des solutions

Afin de régler les problèmes apportés par ces difficultés, nous avons logiquement dû mettre au point des solutions adaptées et réalistes.

Pour ce qui est de la direction du robot, nous avons dû faire des concessions et opter pour une direction plus simple. La solution que nous avons adoptée repose ainsi sur l'envoi d'instructions de déplacement par bluetooth par l'utilisateur via son smartphone. L'approfondissement du système de direction sera détaillé plus tard dans ce rapport.

Concernant la limite de poids, nous avons réussi à l'augmenter légèrement en utilisant des roues plus conséquentes qu'un membre du groupe avait déjà en sa possession. Cependant, la puissance des moteurs étant toujours très limitée, nous n'avons quand même pas réussi à atteindre le poids d'une valise, ce qui était notre objectif initial.

À propos du manque de temps, nous n'avons pas eu d'autre choix que d'accorder plus de notre temps libre au projet afin de pouvoir le finaliser à temps.

Dans le but de régler le problème du manque de communication, nous avons créé un google drive sur lequel nous déposons chaque nouvelle avancée réalisée par un membre du groupe, et nous étions tenus de le consulter quand nous voulions travailler sur le projet à nouveau. En ce sens, nous étions sûrs que les informations étaient à disposition de tous.

Enfin, le dernier problème que nous avons, à savoir le manque de rapidité du logiciel MPLAB, nous n'avons logiquement rien pu y faire car c'était notre seul moyen de programmer le robot.

## **2 CONSTRUCTION ET TESTS**

### **2.1 Partie mécanique**

#### ***2.1.1 Nécessité de cette partie***

La présence d'une partie mécanique peut paraître surprenante dans un projet d'informatique embarquée. En effet, dans le cadre de ce genre de projet, on pense d'abord à l'électronique et au programme informatique. Néanmoins, cette partie mécanique est nécessaire à cause du terme **embarqué**.

Mais tout d'abord qu'est-ce qu'un système embarqué ? : "Un système embarqué est défini comme un système électronique et informatique autonome, souvent temps réel, spécialisé dans une tâche précise"<sup>1</sup>

Dans le cadre de notre projet, ce mot nous a amené à nous poser les questions suivantes :

1. Comment rendre mobile un système électronique ?
2. Comment dimensionner correctement chaque élément afin que chacun trouve sa place sur le véhicule ?
3. Le véhicule résiste-t-il à son environnement ?

Le but de la partie mécanique était donc d'élaborer des solutions pour répondre à chaque question.

#### ***2.1.2 Conception préliminaire***

Premièrement, nous devons clairement identifier les contraintes en répondant à la problématique 1. Nous devons ainsi imaginer un système permettant de transporter d'une part une valise et d'autre part un système électronique. Assez intuitivement, nous avons pensé à réaliser un chariot transportant sur la face supérieure la valise et le circuit électronique sur l'autre face.

Nous avons dû aussi avoir recours à d'autres éléments pour permettre la mobilité du chariot. Nous avons utilisé deux moteurs équipés de roues permettant au chariot d'avancer, mais aussi de s'orienter. Pour continuer, nous avons utilisé une roue folle dont le but était d'équilibrer la structure en créant trois points de contact sur le sol. Les autres éléments étaient des composants électroniques.

Une contrainte que nous avons rencontrée était de faire un système efficace tout en étant simple. En effet, nous avons à gérer le subtil équilibre entre réalisation technique et simplicité puisque nous devons par la suite être capable de concrétiser ces idées à l'aide

---

<sup>1</sup>Se référer à la source 1

d'une maquette. Pour cela, nous avons simplement limité le nombre d'éléments au strict nécessaire ainsi que limiter la présence de pièces intermédiaires.

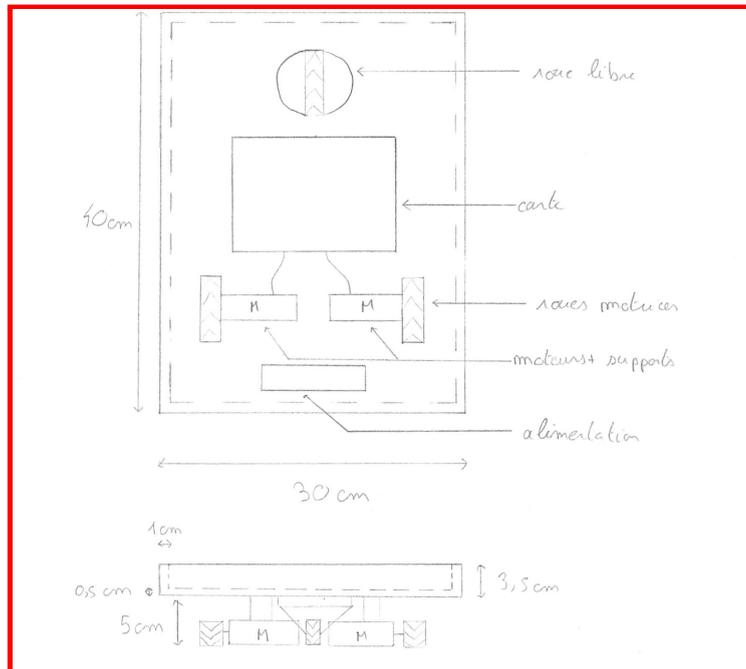


Figure n°1 : Premier schéma du véhicule

Nous avons donc abouti à ce schéma à la fin de la première phase de conception du système. Bien que ce schéma subira quelques modifications, comme expliqué par la suite, il représentera l'idée directrice de la partie mécanique.

### 2.1.3 Dimensionnement et modélisation

Il s'agit dans cette sous-section de répondre à la deuxième problématique. C'est-à-dire de s'assurer que chaque élément aura une place suffisante pour pouvoir être intégré sur le véhicule tout en ne surdimensionnant pas le chariot. D'autre part, nous avons besoin d'une modélisation précise pour :

- Faciliter la communication entre les différents pôles : le modèle en 3D facilitait la compréhension du système pour les membres du groupe ne s'occupant pas de la partie mécanique.
- Faciliter la prise de mesure : lorsque nous n'avons pas encore de maquette, nous avons tout de même besoin de prévoir la place entre les éléments.
- Faciliter le montage : le modèle 3D représentait un bon exemple visuel, ainsi nous n'avons plus qu'à le reproduire.

Pour cela, nous avons eu recours au logiciel SolidWorks, qui est un logiciel de CAO<sup>2</sup>. Ce logiciel nous a permis de faire toutes les tâches que nous avons eu besoin de faire par la suite dans cette partie mécanique.

<sup>2</sup> Conception Assistée par Ordinateur

D'abord, nous avons dû modéliser chaque élément<sup>3</sup> individuellement présent dans notre véhicule (pièces mécaniques et électroniques). La difficulté principale était de transcrire fidèlement la dimension réelle de l'élément dans le logiciel pour obtenir un modèle le plus proche de la réalité.

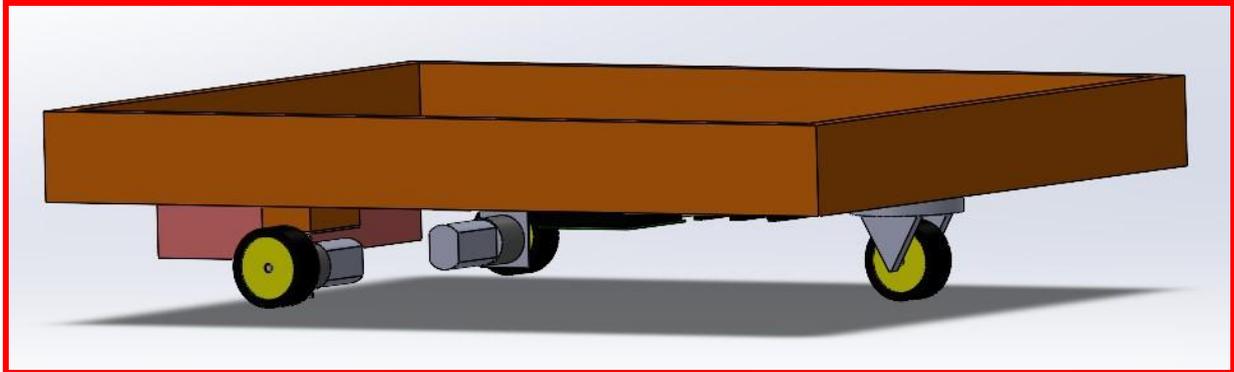


Figure n°2 : Modélisation sur ordinateur du véhicule

Une fois toutes les pièces ont été modélisées, il a fallu représenter les liens physiques qu'entretiennent les pièces entre elles et reproduire leur fonction technologique au sein du véhicule. Ceci a pu se faire en instaurant des contraintes sur les pièces, c'est-à-dire réduire leur degré de liberté (exemple : permettre seulement une rotation selon un axe pour les roues). On a ainsi obtenu la modélisation présentée ci-dessus qui a été la base de toutes nos mesures.

## **2.1.4 Simulation**

### **a) Présentation de la simulation**

Le but de cette étape était de répondre à notre troisième problématique. C'est-à-dire de s'assurer que la structure que nous avons imaginée résistera à son environnement. Lorsqu'on parle d'environnement, on qualifie les éléments avec lesquels le véhicule va interagir. L'élément principal qui représente un véritable obstacle est la masse que doit supporter le support.

Nous voulions donc être certains, avant la conception de la maquette, que notre structure serait adaptée à ce type de sollicitation. Nous avons donc eu l'idée de profiter de la représentation 3D du véhicule sur SolidWorks pour effectuer des tests de sollicitations statiques numériques.

La simulation des efforts a été réalisée seulement sur le support. En effet, il s'agissait de la seule partie que nous avons entièrement conçue. Les autres éléments de la structure étaient des composants standard (moteurs, roues) qui nous étaient fournis et sur lesquels nous ne pouvions pas agir. D'autre part, nous avons conçu le véhicule de manière que les éléments électroniques (alimentation, carte électronique) soient à l'abri des sollicitations. Ainsi, dans le but de simplifier la simulation, nous avons limité cette étude au support.

---

<sup>3</sup> Se référer à l'annexe 1 pour la modélisation des différents éléments

### ***b) Mise en place de la simulation***

La première étape de la simulation a été de compléter notre modélisation. En effet, il fallait choisir un matériau dans le logiciel pour représenter le support réel. Le modèle réel est constitué de la superposition de 3 couches de médium d'épaisseur de 3mm. Néanmoins, ce matériau n'était pas disponible, nous avons dû demander conseils à Mme. Hélène Rade qui nous a conseillé de choisir le bois de Balsa pour représenter informatiquement le véritable matériau. Le but était de représenter la caractéristique légère du matériau et ce type de bois était un bon compromis.

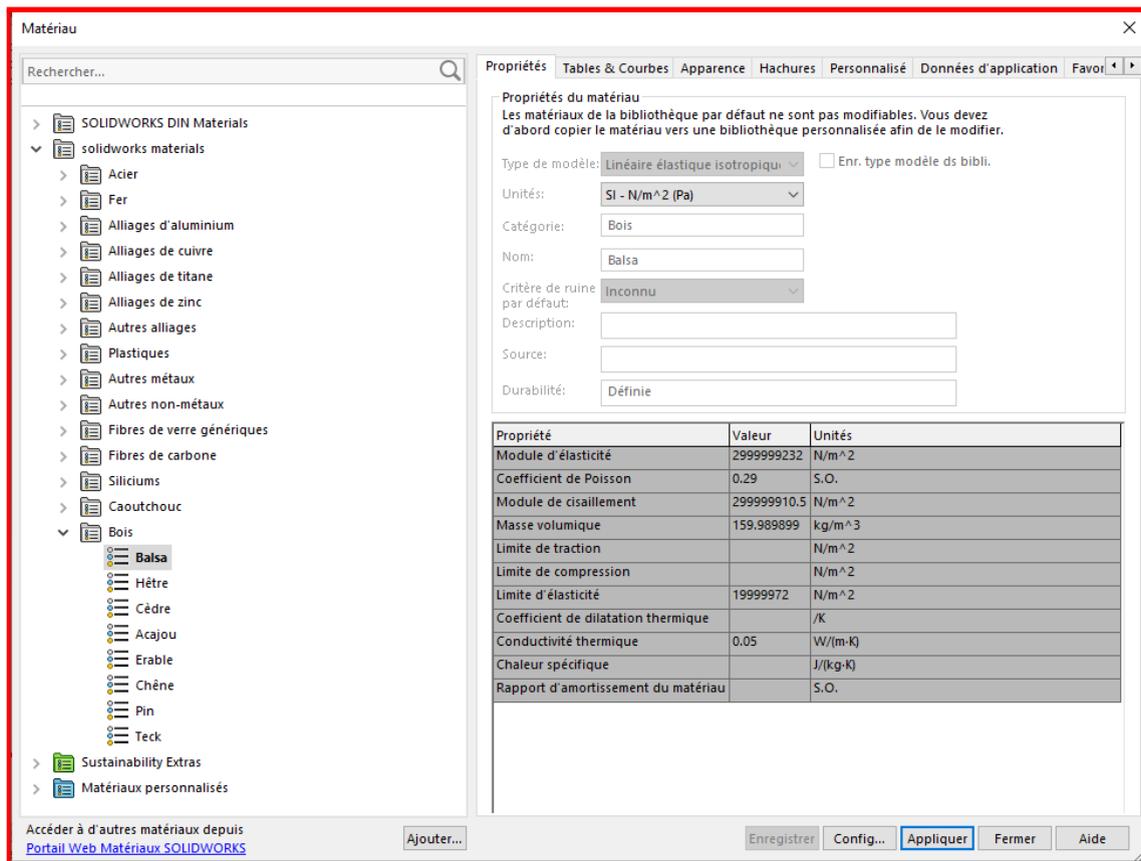


Figure n°3 : Choix du matériau sur SolidWorks

La deuxième étape de la simulation était de représenter les hypothèses sur la structures:

- On considère un effort de 50N<sup>4</sup> réparti sur la totalité de la face supérieure du plateau
- On considère que le plateau est parfaitement fixé sur les autres éléments (moteurs, roues...) et que ces mêmes éléments sont infiniment rigides (puisque l'étude ne porte pas sur eux).
- On considère que le support est un seul bloc de matière parfaitement continu, homogène et sans irrégularité. Contrairement au modèle réel où chaque épaisseur est constituée de trois couches de bois.

Bien que nous ayons dû faire plusieurs hypothèses, il ne s'agit pas d'approximation totalement aberrante. Puisque d'une part, la masse à supporter n'est pas très importante et

<sup>4</sup> Poids d'une valise partiellement chargée de petite taille

d'autre part, comme le montrera la simulation, les déformations et déplacements engendrés ne sont pas suffisamment importants pour rendre ces hypothèses inutilisables.

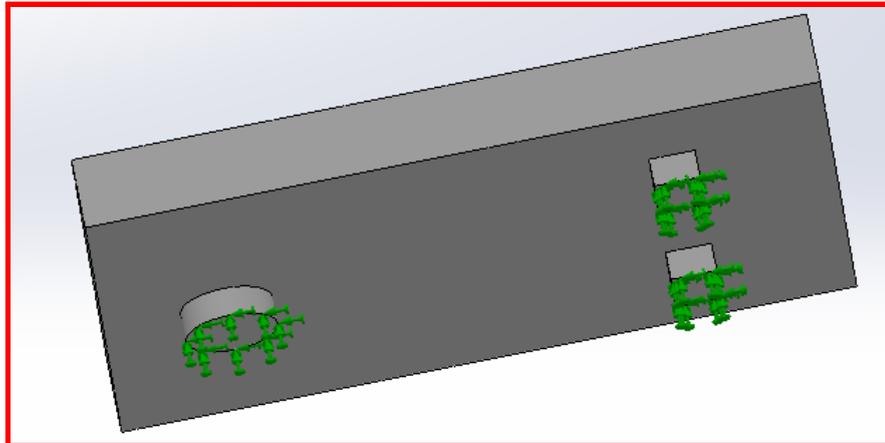


Figure n°4 : Modélisation de l'hypothèse n°2

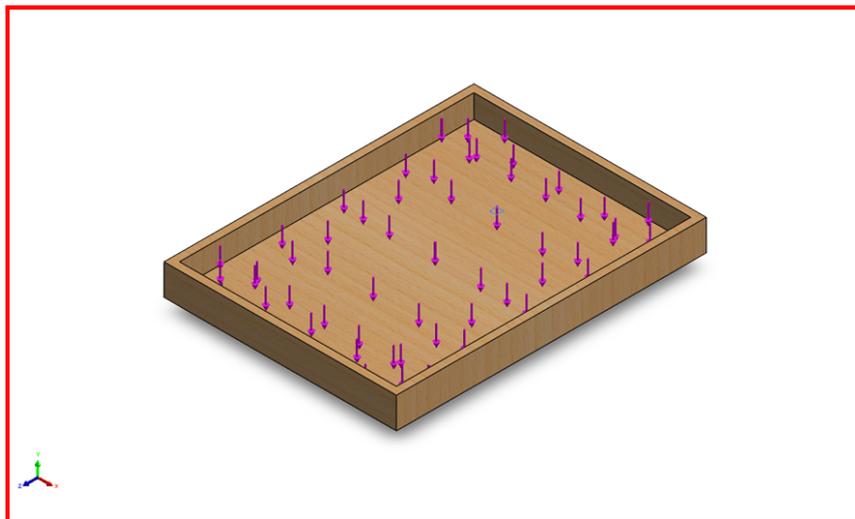


Figure n°5 : Modélisation de l'hypothèse n°1

### ***c) Résultats de la simulation***

Dans l'ultime étape de la modélisation nous avons obtenu les résultats de la simulation et nous avons cherché à les interpréter dans le cadre de notre projet. Cette simulation, aura permis d'étudier 3 points principaux :

- La contrainte : effort par unité de surface.
- Le déplacement : écart de position d'un point de la structure avant et après l'application de l'effort
- La déformation : variation relative des déplacements par rapport à la position initial d'un point de la structure

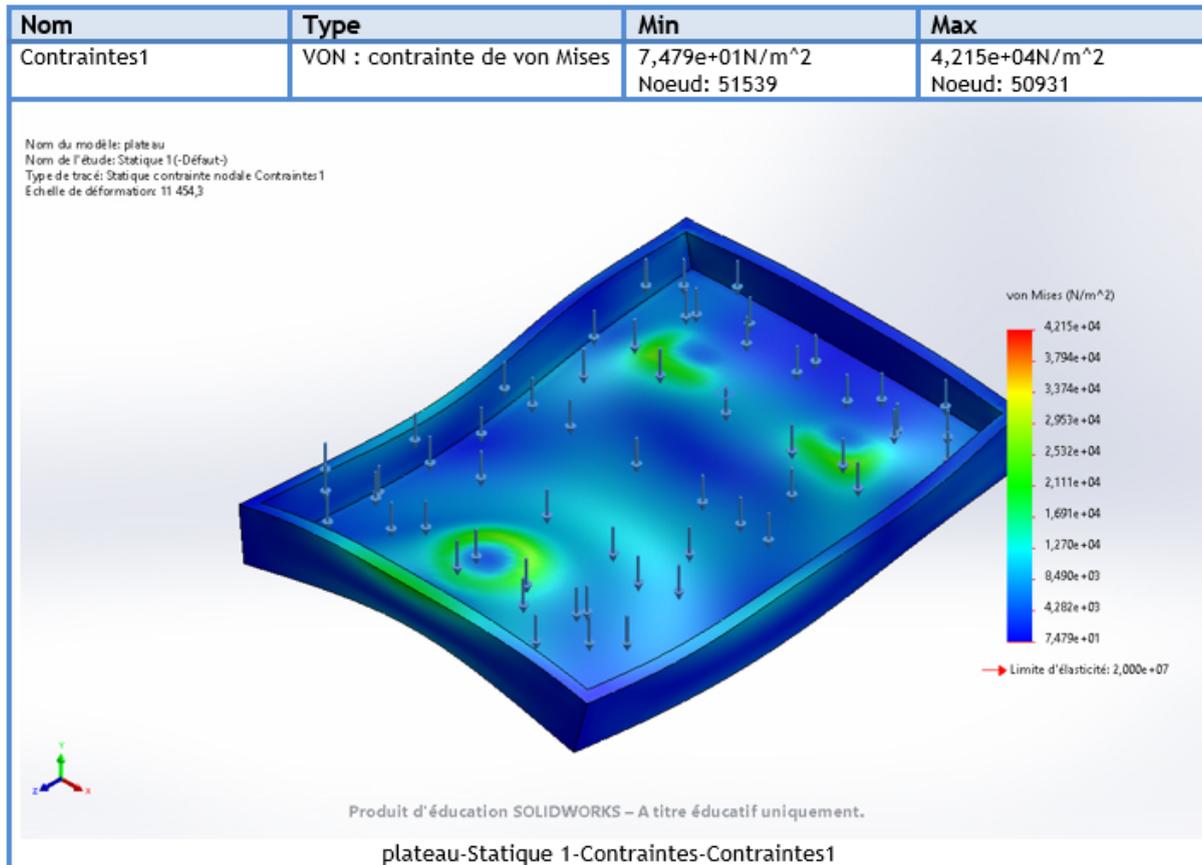


Figure n°6 : Résultats de la simulation pour l'étude des contraintes

Tout d'abord cette première simulation est très instructive, puisqu'on peut y apercevoir la déformée du support à cause de l'effort. Nous commenterons plus tard ce résultat. Pour continuer, on observe, sans surprise, que les plus grandes contraintes sont obtenues vers les supports du plateau. Ceci est tout à fait cohérent puisqu'il s'agit des endroits avec le plus d'irrégularités à cause de la jonction avec les autres éléments.

Un autre résultat intéressant est celui de la contrainte maximale. Le logiciel indique :  $4.215e+04 \text{ N/m}^2$  qui est très inférieur à la limite élastique du matériau qui est de :  $2.0e+07 \text{ N/m}^2$ . On en déduit que notre support ne sera pas du tout déformé définitivement par la masse puisque la limite élastique ne sera pas dépassée.

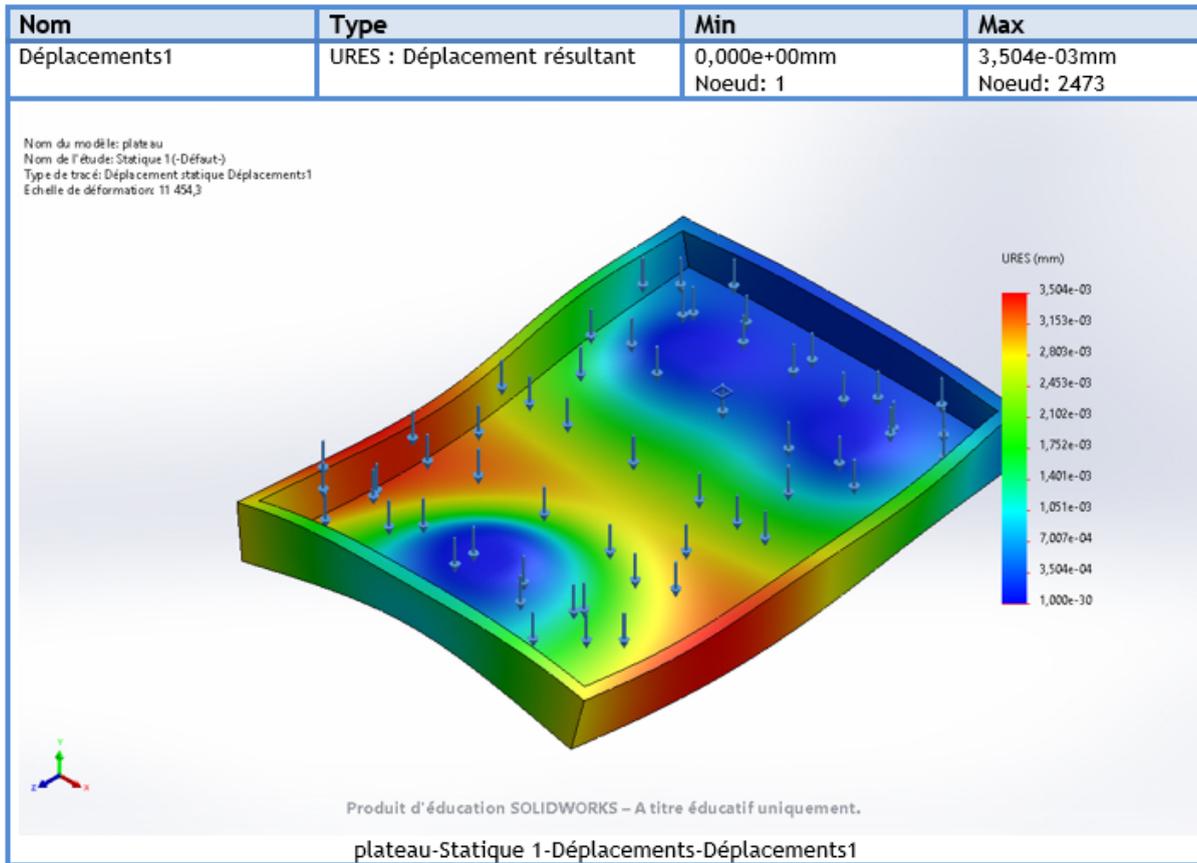


Figure n°7 : Résultats de la simulation pour l'étude des déplacements

Dans le cadre de cette étude nous allons revenir sur la déformée<sup>5</sup> du support. On remarque immédiatement que le support paraît totalement déformé par l'effort. Néanmoins, il s'agit que d'une représentation pour mettre en avant le phénomène de déplacement. En effet, on remarque dans le coin supérieur gauche de l'image, que SolidWorks a imposé un échelle de déformations de 11454,3. C'est pour cela que le support paraît aussi élastique.

On peut observer que les déplacements maximaux sont atteints aux centres des grandes longueurs (les rebords). Néanmoins, nous n'avons pas pu éviter de concevoir le support sans ces longs bords. En effet, il s'agit même d'une contrainte principale, puisque ce dernier doit être capable de contenir les dimensions d'une petite valise.

Comme pour l'étude précédente, le résultat important est celui du déplacement maximal. Puisqu'on constate qu'il est de 3.504e-03mm. Ce déplacement est tout à fait négligeable à l'échelle de notre support et permet de confirmer la validité des hypothèses sur les petits déplacements.

<sup>5</sup> Position des différents points d'un système après déformation

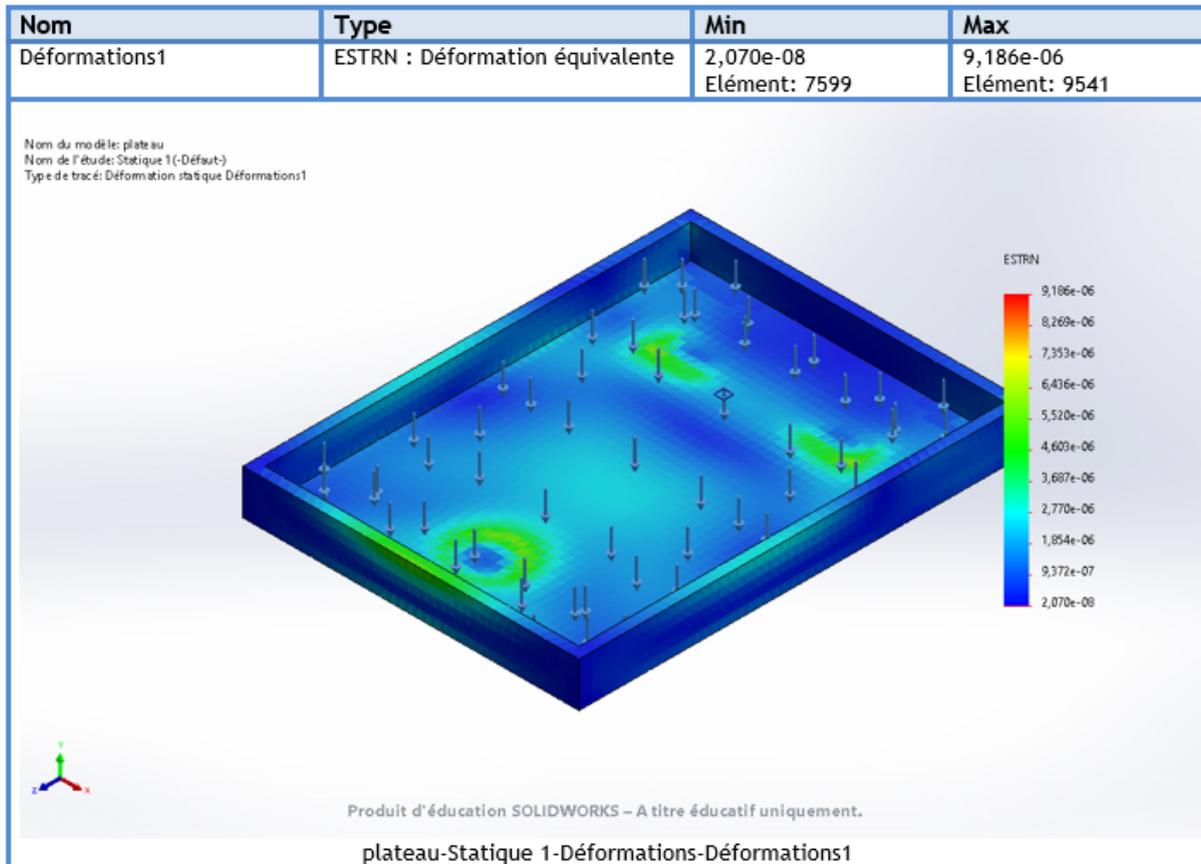


Figure n°8 : Résultats de la simulation pour l'étude des déformations

Enfin, le dernier résultat que nous avons obtenu avec la simulation est celui de la déformation aussi appelé allongement relatif :

$$\varepsilon = \frac{\Delta l}{l_0} = \frac{l - l_0}{l_0}$$

On peut noter que les déformations interviennent surtout au niveau du milieu des grandes longueurs et aux endroits où le support est soutenu, cela provoque des irrégularités au niveau des supports.

Ce résultat est complémentaire à l'étude précédente puisqu'il confirme que les déplacements que nous avons vu précédemment sont négligeables. Puisque la déformation compare le déplacement d'un point par rapport à sa position initiale. Une fois de plus le résultat du logiciel est cohérent puisqu'on obtient un coefficient de 9.186e-6 (sans unité) très proche de 0. On a donc confirmation de la négligeabilité des déplacements.

Finalement, cette simulation et la validation des trois critères ont permis de confirmer que le support conçu résistera parfaitement à des masses jusqu'à environ 5Kg.

### ***2.1.5 Montage de la maquette***

Après avoir effectué une grande partie des calculs théoriques, nous nous sommes attelés au montage du prototype.

Afin d'avoir un modèle précis et fiable au modèle virtuel que nous avons créé sur solidworks ainsi qu'au modèle dessiné sur le plan industriel (fig.2.2), nous avons demandé de l'aide à une intervenante (Mme Rade) qui nous a permis d'avoir accès à du matériel de qualité et à une précision supérieure.

Suite à cette demande, nous avons pu récupérer les pièces initialement conçues sur le dessin technique précédemment effectué afin de pouvoir commencer à assembler le montage final en fixant nos pièces à l'aide d'une colle à bois.

Durant l'assemblage des pièces, nous avons été contraint de devoir redimensionner certaines pièces "sur mesure" à l'aide d'une scie sauteuse.

Une fois toutes les pièces fixées entre elles, nous devons respecter les dimensions très précisément afin de pouvoir positionner la carte électronique ainsi que les roues liées aux moteurs à leurs places respectives.

La précision du montage est une étape importante car elle nous permet de respecter au mieux les résultats que nous avons tirés de la partie précédente, et de prendre en compte le poids final de notre valise et son effet sur l'ensemble du prototype.



Figure n°9 : Photographie des planches lors du montage du support



Figure n°10 : Photographie de l'avancement de la maquette finale

Finalement, nous avons pu effectuer la complétion du modèle avec l'aide du pôle Electronique avec qui nous avons effectué les différents branchements notamment entre les moteurs et la carte électronique.

## 2.2 Partie électronique

Dans cette partie, nous aborderons l'aspect électronique de notre projet. Pour l'étudier dans son ensemble, il faudra étudier plusieurs points qui peuvent être vu comme indépendants, mais qui sont en réalité intrinsèquement liés.

Pour être plus précis, notre partie s'occupe de tout ce qui est lié de près à la carte mère :

- L'étude des composants
- L'étude des branchements et des connectiques
- L'étude énergétique et alimentation des cartes et des moteurs

Nous aborderons également les problèmes rencontrés, les solutions et les compromis que nous avons dû faire.

### 2.2.1 Étude des composants

La première chose que nous avons eu à faire dans notre projet est la familiarisation avec les différents composants. Voici la liste des principaux composants que nous allons utiliser pour notre projet ainsi que leur description, leur utilité et la justification de notre choix.

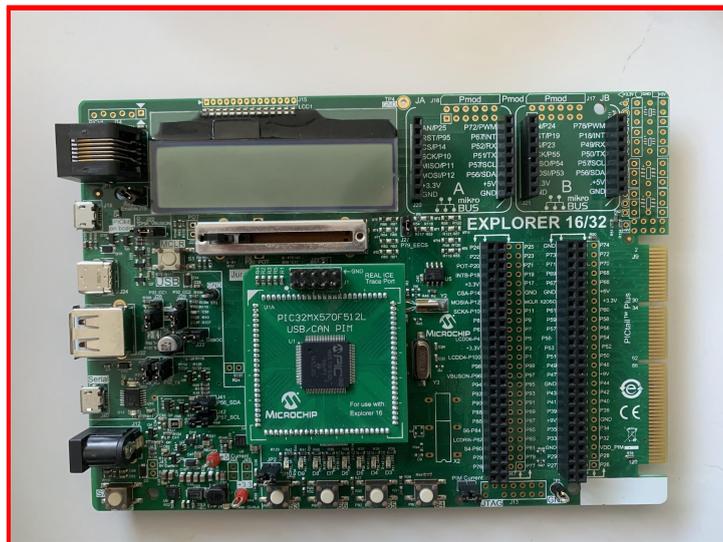


Figure n°11 : Photo de la carte mère Explorer 16/32

Une carte Explorer 16/32 était à notre disposition pour être utilisée comme carte mère. C'est elle qui va recevoir toutes les informations, les utiliser, les traiter et les rediriger vers les différents composants qui lui seront connectés tels que les moteurs et le module Bluetooth. Le choix de cette carte s'est fait plutôt rapidement, en effet elle nous était mise à disposition par l'INSA et elle remplissait parfaitement le travail qui lui est demandé.

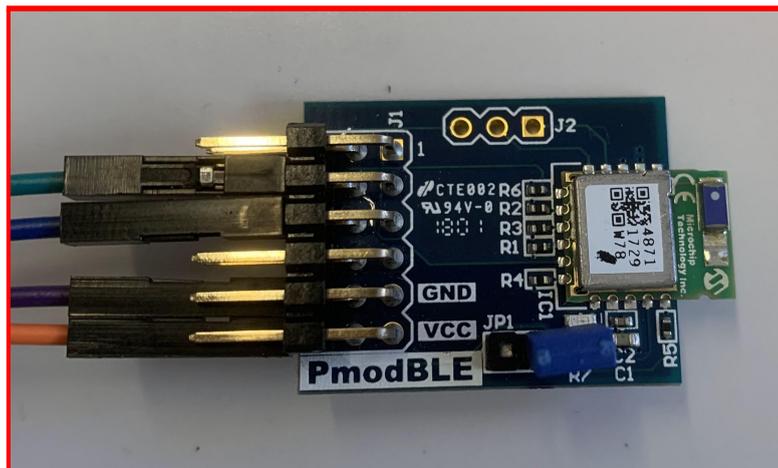


Figure n°12 : Photo de la carte Pmod BLE

Pour faire la liaison entre la carte mère et le téléphone de l'utilisateur, nous sommes passés par la technologie Bluetooth et plus particulièrement le BLE (Bluetooth Low Energy). Cette technologie est un standard dans la communication avec les appareils mobiles tels que les smartphones. En effet, elle est plutôt simple d'utilisation en comparaison aux systèmes GPS, WIFI, GMS, NFC, etc. et elle permet un échange de données plutôt rapide sans se soucier des problèmes de compatibilité IOS/Android.

Dans la même optique que la carte Explorer 16/32, nous avons choisi ce module BLE de la marque Pmod car l'INSA en disposait déjà et nous les mettaient à notre disposition.



Figure n°13 : Photo d'un des moteurs

Enfin, pour faire avancer notre robot, nous avons besoin de moteurs, nous nous sommes orientés vers le modèle IG220019X00015R, très souvent utilisé pour ce genre de petit projet, il remplit parfaitement son rôle qui est de délivrer un couple nécessaire pour faire avancer notre robot.

Pour contrôler les moteurs et ainsi modifier leur vitesse de rotation, il y a besoin d'un autre composant : le PmodHB5



Figure n°14 : Photo de la carte contrôleur moteur PmodHB5

Cette carte PmodHB85 nous permet de contrôler les moteurs. En effet, elle est capable de recevoir différentes informations comme le sens de rotation, la tension et la période d'activation. C'est grâce à elle que l'on pourra faire plusieurs vitesses de déplacement et des directions différentes.

### 2.2.2 Étude des branchements

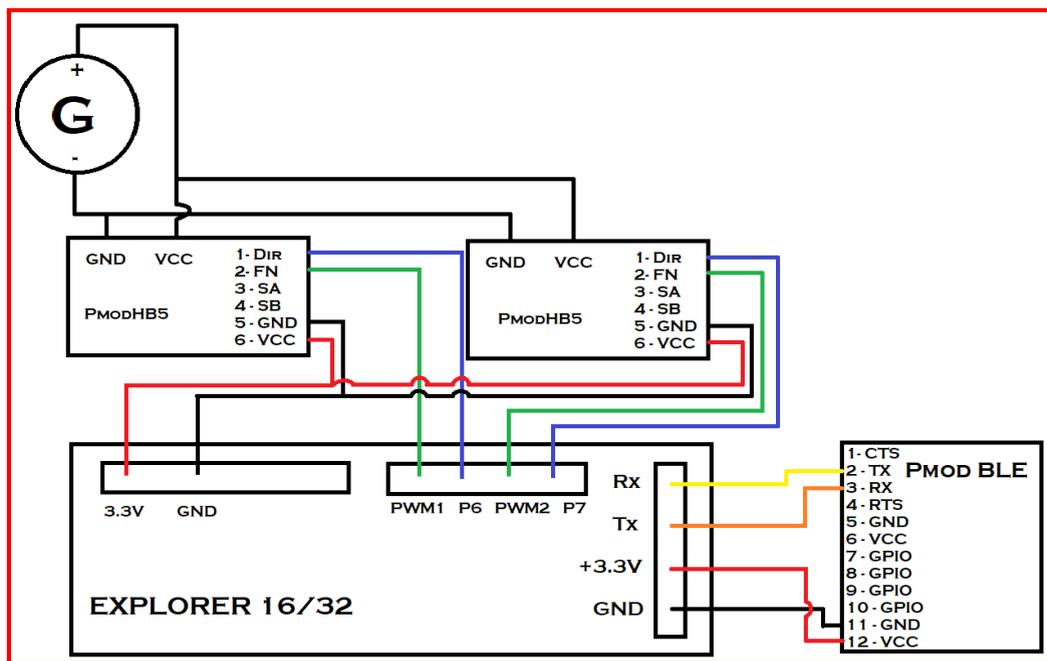


Figure n°15 : Schéma des branchements entre les différents composants électroniques

L'étude des branchements s'est faite plutôt facilement, nous avons travaillé avec la partie informatique pour savoir quelle broche d'accès utiliser. Pour nos tests, nous avons utilisé un générateur qui fera office d'alimentation, à terme, les piles le remplaceront.

Nous passons les informations par les broches RX et TX qui permettent, respectivement, la réception et la transmission des données au module bluetooth.

Le branchement des moteurs est plutôt simple, les modules qui les contrôlent possèdent une alimentation principale et des broches à plusieurs utilisations :

- DIR permet de recevoir la direction (avant ou arrière)
- FN reçoit l'information de la mise en fonctionnement ou non
- GND représente la terre
- VCC la broche d'alimentation

Quant au module BLE, nous n'utilisons que les broches principales :

- RX qui reçoit les informations
- TX qui émet les informations
- GND
- VCC

Pour la carte Explorer 16/32 nous avons eu quelques problèmes, la carte comporte beaucoup de ports différents, lire et comprendre la documentation fut très compliqué.

Le branchement global se fait suivant le schéma ci-dessus, il est important de noter que l'alimentation de la carte mère ne figure pas sur le schéma, car pour nos tests, elle était connectée par câble USB à l'ordinateur pour téléverser le programme et donc alimentée par ce même câble, encore une fois, à terme elle sera connectée à l'alimentation (les piles).

### ***2.2.3 Étude énergétique et alimentation des cartes et des moteurs***



Figure n°16 : Photo de l'alimentation finale

L'alimentation utilisée sera à terme deux piles AA qui délivreront **une tension de 3.3V**. Pour nos tests, nous utilisons à la place un générateur. Pour ce qui est des moteurs, nous allons utiliser des moteurs vus précédemment.

Pour choisir la vitesse des moteurs, nous allons utiliser la technologie PWM. Elle permet d'obtenir une sortie analogique (entre 0 et 12V) avec des moyennes numériques (0 ou 1). Pour ce faire, il faut jouer sur le rapport Haut/Bas (appelé cycle) de la tension. Plus il y aura de temps Haut, plus il y aura de tension et donc plus le moteur tournera vite.

Notre objectif est d'établir plusieurs vitesses de rotation dans le fonctionnement des moteurs pour que le robot puisse tourner à droite ou à gauche. Dans la documentation officielle, on trouve un graphique récapitulant l'évolution des différentes valeurs (annexe 2.5), ainsi que le rapport Tension / tour par minute qui est de **8,68 mV/tr/min**.

Nous aurons une alimentation de 3.3 V ainsi on obtient, avec un cycle de 100%, une rotation de  $3.3/0.00868 = 380 \text{ tr/min}$ .

Nous aimerions une vitesse maximale pour aller en avant et une vitesse intermédiaire permettant de faire tourner le robot, nous allons en conséquence prendre de manière arbitraire les deux cycles : 100% et 75% soit 380 tr/min et 285 tr/min.

Il est cependant nécessaire de faire des tests pour confirmer que notre choix de vitesse soit compatible avec la réalité. Ces tests seront réalisés durant la phase du test global, il est en effet nécessaire d'avoir le produit final pour ajuster correctement les cycles.

### **2.3 Partie informatique**

Dans cette partie, nous aborderons l'aspect informatique du projet. Dans ce projet, deux parties informatiques sont importantes. D'une part, la communication bluetooth entre un téléphone et la carte. D'autre part, le pilotage des moteurs afin de faire tourner les roues de la bonne façon.

Partant de zéro dans le domaine de l'informatique embarquée, nous avons donc eu besoin de beaucoup de séances dans le but de se familiariser avec le C et MPLAB, le logiciel utilisé pour transférer le programme dans la carte programmable. Nous avons donc commencé par aborder la partie théorique de la carte programmable avec toutes les fonctionnalités qu'elle avait et tous les différents ports qui pouvaient être utilisés.

Nous avons ensuite commencé la partie pratique, qui était le code. Nous avons d'abord dû analyser un premier code qui nous avait été donné, qui permettait d'allumer et d'éteindre une led à la vitesse de chaque itération de la boucle du programme, ce qui faisait qu'on ne voyait pas vraiment la led clignoter car la vitesse était trop rapide. C'est pour cela que nous avons ensuite abordé le principe de temporisation.

Une fois la compréhension de cette partie faite, nous avons pu ajouter à notre code de clignotement des led des temporisateurs, afin d'allumer et d'éteindre la led toutes les 4 secondes par exemple. Il fallait aussi se familiariser avec le logiciel afin d'éviter toutes erreurs qui ne seraient pas liées avec le programme mais avec la mauvaise initialisation de MPLAB. Tout l'apprentissage a bien évidemment été réalisé dans un premier temps en simulation puis avec l'utilisation de la carte électronique.

Par la suite, nous avons consacré la majeure partie de notre temps à l'apprentissage et à la compréhension des divers composants mis à notre disposition.

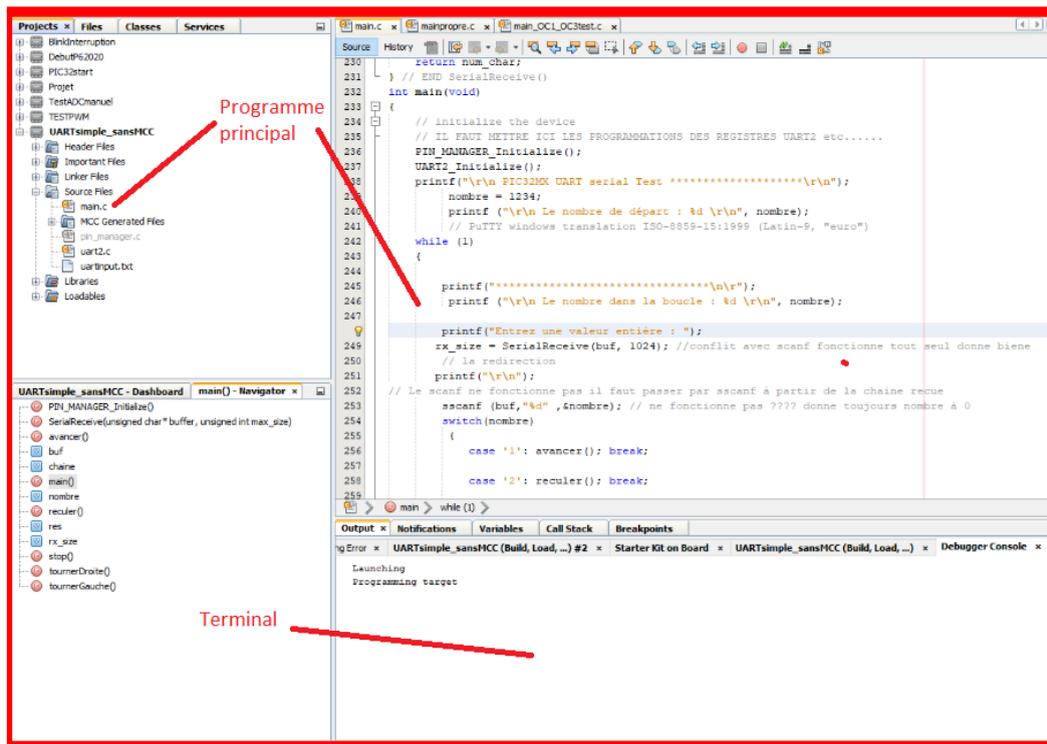


Figure n°17 : Interface MPLAB

Par la suite, après avoir décidé que notre projet avait pour but de transporter une valise de manière autonome avec notre téléphone, il fallait réfléchir sur les différentes parties du code à réaliser. Nous avons donc commencé par faire des analyses du problème et des analyses descendantes. La première étape était le bluetooth. Celui-ci a été particulièrement complexe à prendre en main. La première tâche a été de comprendre le fonctionnement du module Pmod BLE qui servait à la communication bluetooth avec le téléphone. Nous avons dû passer énormément de temps à faire des recherches dans les fiches techniques afin de bien comprendre le fonctionnement et la cohérence de ses caractéristiques avec notre programme (baud, branchement RX et TX, ...). Nous avons aussi cherché des informations sur internet à propos de la démarche pour réussir notre programme de communication avec un téléphone.

Une fois ceci fait, il fallait réussir à échanger des données avec un appareil, ce qui n'a pas été simple à cause de différents facteurs : la connexion entre les deux appareils, l'impossibilité d'envoyer un caractère avec notre application terminal ou encore des problèmes du Pmod BLE qui ne fonctionnait pas. Ces problèmes nous ont retardés sur l'évolution du programme puisqu'il fallait parfois tout reprendre de zéro.

Une fois ces problèmes résolus nous sommes passés à la partie contrôle des moteurs. Nous avons 5 fonctions à coder : avancer tout droit, tourner à droite, tourner à gauche, reculer et s'arrêter. Les fonctions avancer et reculer et s'arrêter ont été assez simples à coder car il suffisait de mettre les deux moteurs sur /on puis ensuite choisir le sens dans lequel les roues allaient tourner ou bien alors simplement mettre les moteurs sur /off. Cependant, pour les fonctions "tourner" à gauche ou à droite, nous nous sommes dit qu'il fallait faire une fonction qui ferait tourner la valise non pas sur elle-même, mais plutôt qui la ferait avancer en courbe afin de permettre à l'utilisateur de perdre le moins de temps possible. Nous avons donc eu recours aux personnes responsables de la partie électronique dans le but de nous fournir

une courbe, représentant la vitesse du moteur en fonction de la tension envoyée. Nous avons ensuite fait des recherches sur la manière de délivrer la bonne tension au bon moment.

Afin de rendre l'utilisation de notre programme plus facile par l'utilisateur, nous avons donc créé une application Android à l'aide de MIT App Inventor où nous avons juste à faire une interface facile d'utilisation où à chaque fois qu'un bouton est appuyé, un caractère lié au bouton est envoyé à l'aide du bluetooth à la carte programmable, qui réalisera la fonction associée.

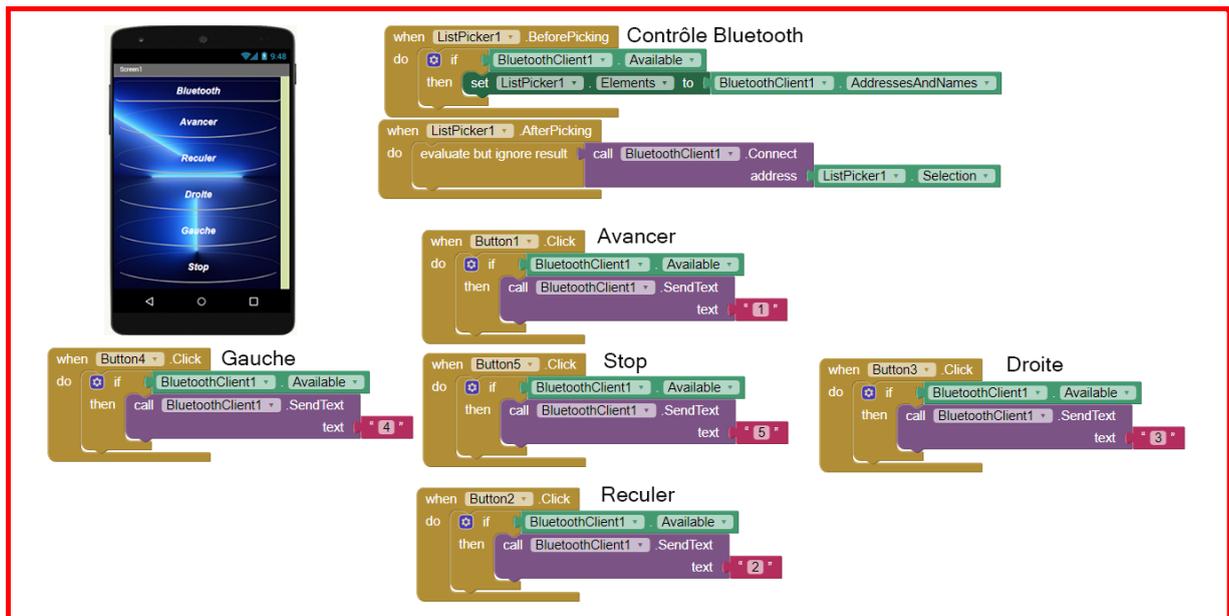


Figure n°18 : Application Contrôle Bluetooth

Le problème majeur que nous avons rencontré ici était l'implémentation du pilotage des ports PWM à travers notre programme. En effet, il a été complexe de trouver les bonnes commandes et les bons ports à contrôler. Cependant, nous avons réussi en coopérant avec le binôme de la partie électronique.

## 2.4 Test Global

Le test global consiste à réaliser la maquette finale et réaliser les tests nécessaires. En effet, elle comprend plusieurs étapes qui doivent chacune être effectuées et analysées. Par la suite, en fonction des besoins et des attentes, des ajustements seront réalisés.

Dans notre cas, nous devons vérifier plusieurs points : si les forces sont bien compensées via la structure, si les composants sont bien fixés et bien branchés et enfin si le robot avance correctement.

Il est plus ou moins simple de faire ces tests. La vérification de la structure et la fixation des composants consistent juste à contrôler la rigidité du robot, il faudra cependant plus d'analyse pour la vitesse et la rotation des moteurs. Pour ces derniers tests, le protocole restera simple : si le robot est trop lent, il faudra augmenter le cycle (le rapport Haut/Bas du port PWM) dans le code, à contrario, il faudra le baisser si le robot va trop vite.

Malheureusement, principalement par manque de temps et un manque d'efficacité, nous n'avons pas eu l'occasion de réaliser le test global tous les tests. Toutefois, nous avons toute la partie théorique de notre projet, de ce fait, nous pourrions monter la maquette et réaliser ces tests.

## CONCLUSIONS ET PERSPECTIVES D'AMÉLIORATION

En conclusion de ce projet, nous avons pu observer et se familiariser avec l'informatique embarquée. Nos études nous ont permis de se rendre compte que ce domaine n'est pas que du code et de la programmation, mais un ensemble de méthodes et de domaines d'application différents tels que l'électronique et la mécanique pour ne citer qu'eux. La partie mécanique nous a fait souligner l'importance de s'adapter à la structure que nous utilisons. La partie électronique, quant à elle, montre les limites des composants (tension, couple moteur, etc.) et par extension ce que l'on peut réaliser. Enfin, la partie informatique, le domaine principal de notre sujet, coordonne tout notre projet et permet de le faire vivre.

Leur étude fut une expérience très enrichissante, chercher des études similaires, travailler sur la théorie, préparer des simulations/expériences et analyser des données était notre quotidien pour ce projet de P6.

Nous regrettons le fait de ne pas avoir pu finir notre maquette fonctionnelle de robot à temps, elle nous aurait offert une grande satisfaction de travail accompli, mais nous en tirons une bonne leçon sur l'organisation du travail en groupe. En tant que futurs ingénieurs, nous serons amenés à travailler en groupe et ce projet de P6 en fût un très bon exercice.

Dans une perspective d'amélioration de notre projet, nous avons plusieurs idées telles que l'implémentation d'un accéléromètre dans le robot, améliorer le dimensionnement du robot dans l'optique qu'il supporte des charges plus lourdes, ou encore, incorporer d'autres fonctionnalités sur le robot comme la possibilité de se repositionner sur les roues si besoin ou franchir/contourner des obstacles.

Enfin, nous aimerions remercier M. Grisel pour son aide et sa supervision tout au long de ce projet.

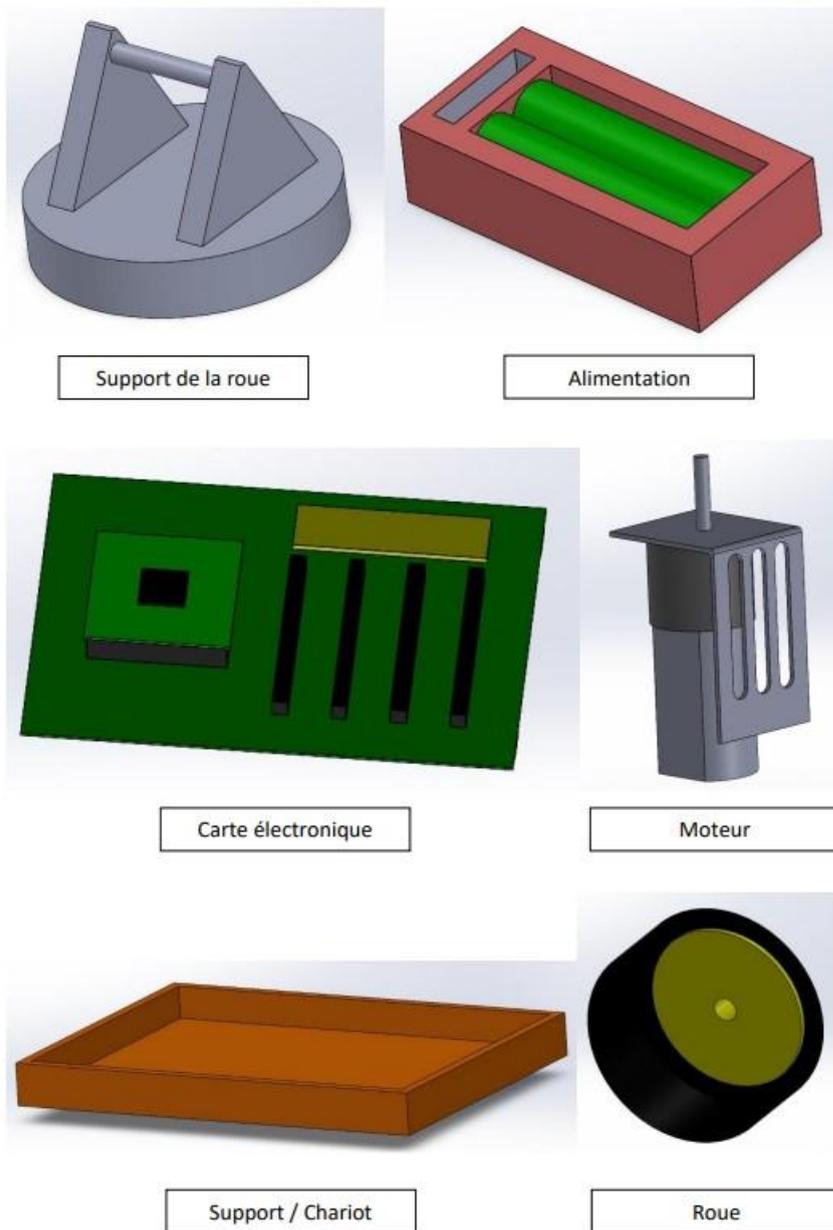
## 1. BIBLIOGRAPHIE

- [1] Wikipédia, “Système embarqué”, (valide à la date du 23/05/2022),  
[https://fr.wikipedia.org/wiki/Syst%C3%A8me\\_embarqu%C3%A9](https://fr.wikipedia.org/wiki/Syst%C3%A8me_embarqu%C3%A9)
- [2] DfRobot, [https://wiki.dfrobot.com/12V\\_DC\\_Motor\\_251rpm\\_w\\_Encoder\\_SKU\\_FIT0186](https://wiki.dfrobot.com/12V_DC_Motor_251rpm_w_Encoder_SKU_FIT0186)  
(valide à la date du 16/05/2022).
- [3] Diligent, “Pmod HB5”, <https://diligent.com/reference/pmod/pmodhb5/start> (valide à la date  
du 16/05/2022).
- [4] Diligent, “Pmod BLE”, <https://diligent.com/reference/pmod/pmodble/reference-manual>  
(valide à la date du 16/05/2022).
- [5] Microchip, “Explorer 16/32 DEVELOPMENT BOARD”,  
<https://www.microchip.com/en-us/development-tool/dm240001-2> (valide à la date du  
17/05/2022).
- [6] Conrad, “Explorer 16/32 Development Board” ,  
<https://asset.conrad.com/media10/add/160267/c1/-/en/001514342DS01/datablad-1514342-microchip-technology-dm240001-3-developmentboard-dspic-pic.pdf> (valide à la date du  
23/05/2022).
- [7] Microchip, “PIC32MX795F512L”,  
<https://www.microchip.com/en-us/product/PIC32MX795F512L> (valide à la date du  
30/05/2022).

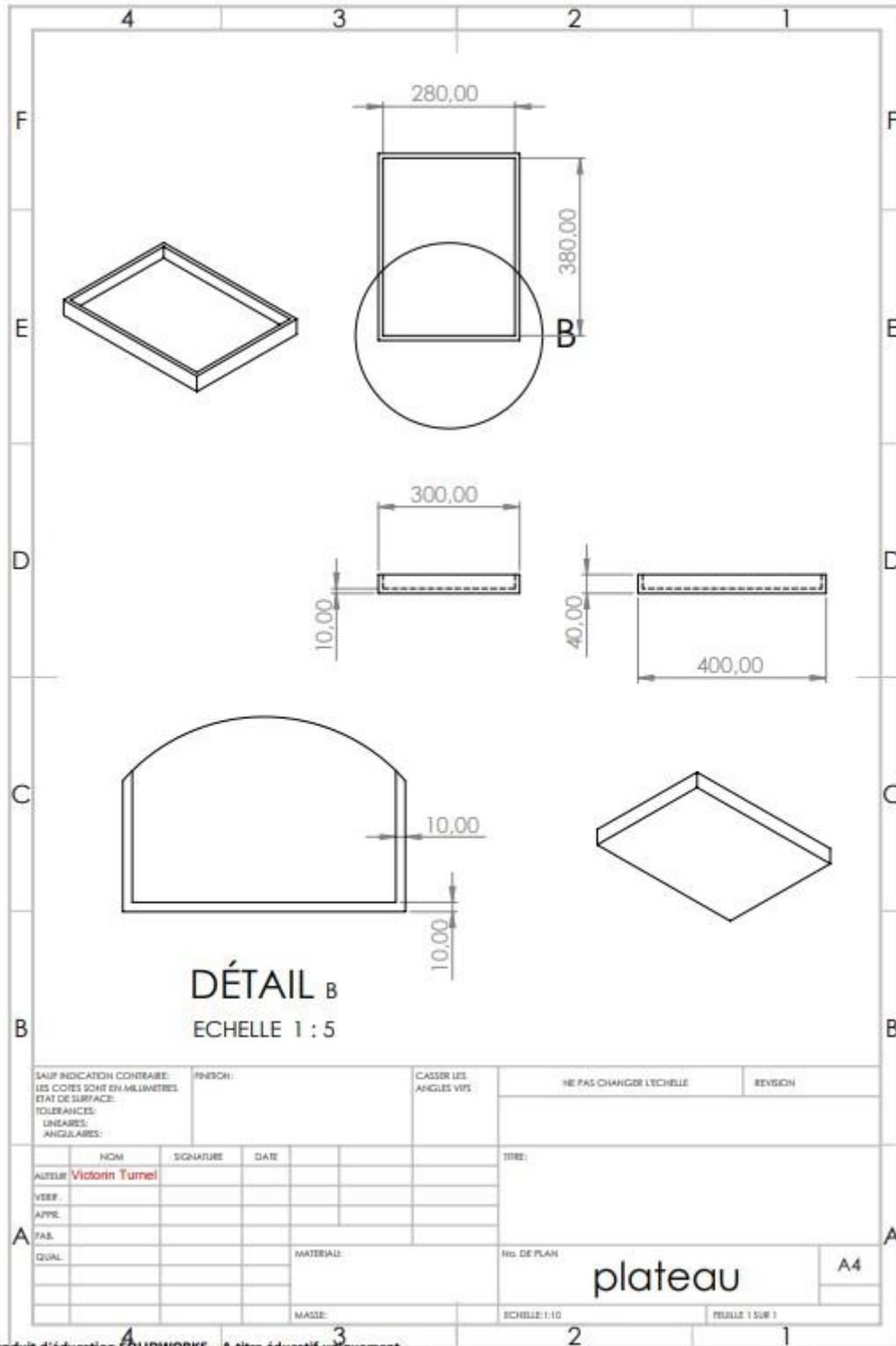
## 2. ANNEXES

### 2.1. Modélisation des éléments du projet

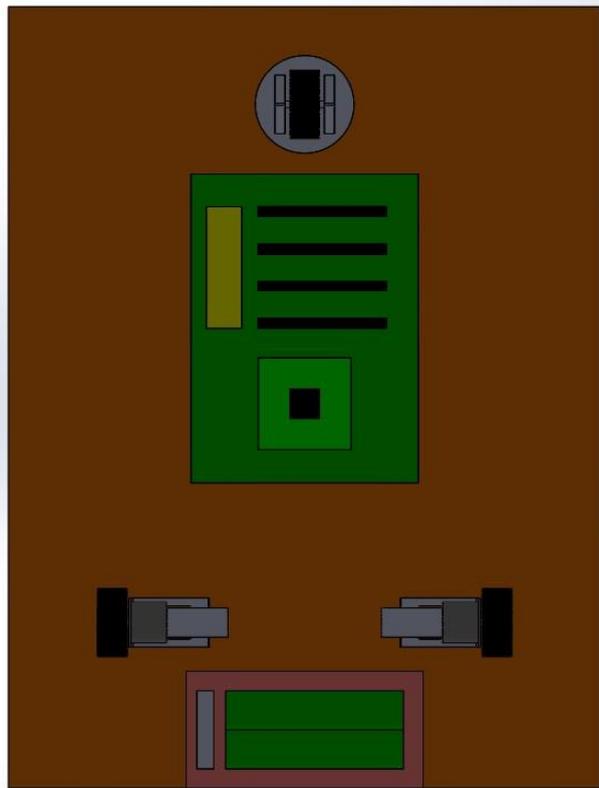
#### Modélisation des éléments du projet



2.2. Dessin technique du support pour la maquette



### 2.3. Modélisation vue inférieure du véhicule et nomenclature



No. ARTICLE	NUMERO DE PIECE	DESCRIPTION	QTE
1	carte électronique		1
2	moteur		2
3	support		1
4	roue		2
5	roue folle		1
6	batterie		1

2.4. Courbe des différentes variables du moteur.

