

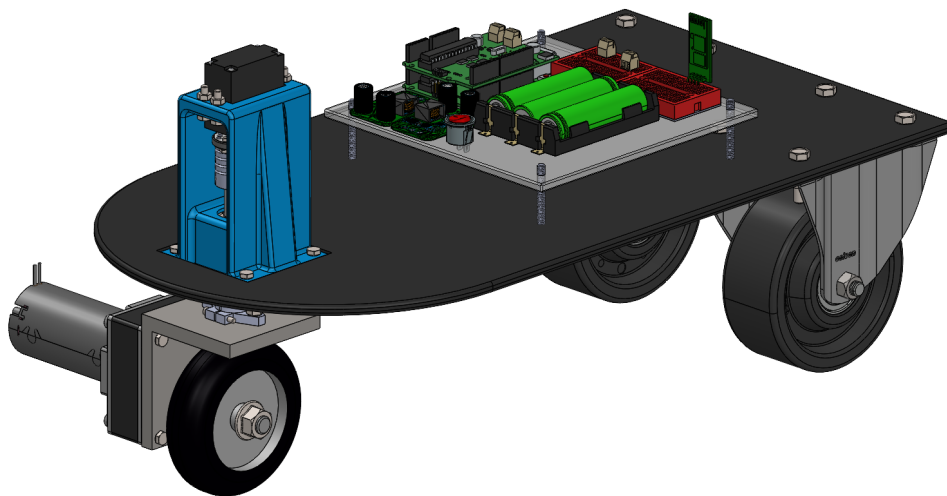


University
Mohammed VI
Polytechnic



Rapport Final

Groupe 13 — Class 2030



Samy Youssoufine
Sofia Zakani
Othmane Bouaafia

Janvier 2026

Remerciements

Nous tenons en premier lieu à remercier l'équipe pédagogique de l'UM6P – EMINES pour l'encadrement, les ressources mises à disposition et les précieux conseils prodigués tout au long de ce premier projet d'ingénierie.

Nous adressons une reconnaissance toute particulière aux responsables et techniciens du FabLab. Leur expertise technique et leur disponibilité nous ont été indispensables pour surmonter les défis liés à la fabrication.

Enfin, ce projet est le fruit d'un travail d'équipe intense. Nous nous remercions mutuellement pour l'implication, la rigueur et la bonne humeur qui ont permis de mener ce projet à son terme.



Résumé

Ce rapport synthétise tout le travail réalisé dans le cadre du premier projet d'ingénierie, en détaillant particulièrement les avancées et les choix de conception effectués depuis la dernière soutenance. Il est structuré en 6 parties principales :

- Une partie **Analyse Fonctionnelle** qui présente le cahier des charges fonctionnel, les diagrammes des cas d'utilisation et des exigences, ainsi que la chaîne fonctionnelle, le schéma synoptique et le schéma cinématique de la plateforme mobile.
- Une partie **Conception** détaillant la maquette CAO de la plateforme mobile, ses sous-ensembles majeurs, ainsi que les choix de matériaux et de composants.
- Une partie **Fabrication** décrivant les processus de fabrication utilisés pour réaliser les différents composants de la plateforme mobile.
- Une partie **Automatique** expliquant les méthodes d'asservissement en vitesse et en position mises en place pour le moteur CC.
- Une partie **Électrique et Électronique** présentant le schéma électrique de la plateforme mobile, les choix des composants électroniques, ainsi que l'intégration de ces composants dans le système.
- Une partie **Logiciel** détaillant le développement du code Arduino pour la gestion de la plateforme mobile, incluant la communication Bluetooth avec l'utilisateur, ainsi que l'interface utilisateur développée en HTML/CSS/JavaScript.
- Une partie **Tests et Validation** qui compile les résultats des tests effectués pour valider les performances de la plateforme mobile par rapport aux exigences du cahier des charges fonctionnel.

Table des matières

1	Analyse fonctionnelle	2
1.1	Présentation générale du projet	2
1.2	Rappel du cahier des charges fonctionnel	2
1.3	Diagramme des cas d'utilisation	3
1.4	Diagramme des exigences	3
1.5	Chaîne fonctionnelle	5
1.6	Schéma synoptique	5
1.7	Schéma cinématique	6
2	Conception	8
2.1	Sous-ensembles	8
2.2	Pièces principales	10
2.3	Liaisons mécaniques	20
2.4	Coupes importantes	22
3	Fabrication	25
3.1	Contraintes de fabrication	25
3.2	Récapitulatif des étapes de fabrication	25
3.3	Pièces fabriquées	27
3.4	Détail de fabrication d'une pièce : Porte-servomoteur	34
4	Automatique	37
4.1	Modélisation théorique du moteur CC	37
4.2	Identification des paramètres du moteur CC	38
4.3	Modélisation de l'asservissement	46
5	Électrique et électronique	54
5.1	Architecture électrique	54
5.2	Schéma électrique	57
6	Logiciel	59

6.1	Architecture logicielle globale	59
6.2	Logiciel embarqué (Firmware Arduino)	59
6.3	Interface Homme-Machine (IHM)	61
	Conclusion Générale	63
	Annexe	64

Table des figures

1	Diagramme des cas d'utilisation de la plateforme mobile	3
2	Diagramme des exigences de la plateforme mobile	4
3	Chaîne fonctionnelle de la plateforme mobile	5
4	Schéma synoptique de la plateforme mobile	6
5	Schéma cinématique de la plateforme mobile	6
6	Maquette CAO de la plateforme mobile	8
7	Sous-ensemble châssis de la plateforme mobile	9
8	Sous-ensemble porte-servomoteur de la plateforme mobile	9
9	Sous-ensemble groupe de propulsion avant de la plateforme mobile (vue iso- ométrique et vue de coin)	10
10	Vue isométrique du châssis	11
11	Vue de dessus du châssis	11
12	Mise en plan de la pièce châssis avec dimensions	12
13	Vue isométrique 1 (coin avant) du porte-servomoteur	13
14	Vue isométrique 2 (coin arrière) du porte-servomoteur	13
15	Vue de face du porte-servomoteur	13
16	Vue de bas du porte-servomoteur, avec les logements pour les roulements .	13
17	Vue orthogonale du palonnier	14
18	Vue de côté du palonnier	14
19	Vue de face du palonnier	14
20	Vue arrière du palonnier	14
21	Mise en plan de la pièce palonnier avec dimensions	15
22	Mise en plan de l'arbre de guidage en rotation avec dimensions	16
23	Rendu de l'arbre	16
24	Pièce cornière de la plateforme mobile	17
25	Mise en plan de la pièce cornière avec dimensions	17

26	Vue isométrique du moyeu	18
27	Vue latérale du moyeu	18
28	Mise en plan de la pièce moyeu avec dimensions	19
29	Plaque de fixation (vide et sans perçages) - Vue isométrique	19
30	Plaque de fixation (avec composants) - Vue isométrique	19
31	Plaque de fixation des composants électroniques - Vue isométrique légendée	20
32	Plan de coupe au niveau des roulements de l'arbre de guidage en rotation dans le porte-servomoteur	22
33	Coupe au niveau des roulements de l'arbre de guidage en rotation dans le porte-servomoteur	22
34	Plan de coupe au niveau du moyeu de la roue avant	23
35	Coupe au niveau du moyeu de la roue avant	23
36	Diagramme de flux des étapes de fabrication des pièces de la plateforme . .	26
37	Châssis réel de la plateforme mobile (avec roues arrières montées)	27
38	Vue de dessus du châssis	28
39	Vue latérale du châssis	28
40	Vue du châssis avec les roues arrière montées	28
41	Porte-servomoteur réel de la plateforme mobile	29
42	Vue arrière du porte-servomoteur	29
43	Vue de face du porte-servomoteur	29
44	Vue de dessus du porte-servomoteur	29
45	Cornière réelle de la plateforme mobile (avec arbre moteur inséré)	30
46	Moyeu fabriqué	30
47	Arbre de guidage en rotation fabriqué	31
48	Palonnier fabriqué et assemblé (l'intégration immédiate n'a pas permis de prise de vue isolée)	31
49	Plaque de fixation des appareils électroniques (vide)	32
50	Plaque de fixation des appareils électroniques (avec composants montés) . .	32
51	Récapitulatif des étapes de fabrication du porte-servomoteur	35
52	Modélisation électrique et mécanique du moteur CC	37
53	Schéma global du protocole expérimental pour la détermination des paramètres du moteur CC	40
54	Montage expérimental pour la détermination des paramètres du moteur CC (expériences à vide)	42

55	Montage expérimental pour la détermination des paramètres du moteur CC (expérience avec charge)	43
56	Graphique de la vitesse de rotation en régime permanent ω_{prm} sans charge pour plusieurs valeurs de tension.	44
57	Graphique des vitesses de rotation sans charge avec tension fixe (mesure de τ).	45
58	Graphique de la vitesse de rotation en régime permanent avec charge.	46
59	Schéma simplifié du correcteur PI utilisé pour l'asservissement en vitesse	47
60	Schéma bloc de l'asservissement en vitesse du moteur CC	48
61	Résultats de la simulation de l'asservissement en vitesse du moteur CC (vitesse consigne de 0,10 m/s)	50
62	Schéma bloc de l'asservissement en position du moteur CC	51
63	Résultats de la simulation de l'asservissement en position du moteur CC (position consigne de 1 m)	52
64	Câbles multibrin pour l'alimentation de puissance	55
65	Borniers à vis pour connexions électriques	55
66	Schéma électrique des différents composants de la plateforme.	57
67	Schéma du circuit électrique réalisé avec Fritzing.	57
68	Architecture logicielle globale de la plateforme mobile	59
69	Capture d'écran de l'interface utilisateur conçue	61

Liste des tableaux

1	Tableau des exigences fonctionnelles et contraintes	2
2	Récapitulatif des pièces principales de la plateforme mobile	10
3	Récapitulatif des liaisons mécaniques principales de la plateforme mobile . .	21
4	Caractéristiques des roulements utilisés pour l'arbre de guidage en rotation .	22
5	Récapitulatif des pièces fabriquées, matériaux et procédés de fabrication. . .	27
6	Problèmes rencontrés lors de la fabrication et solutions apportées.	33
7	Récapitulatif des vis et éléments de fixation utilisés.	33
8	Récapitulatif des expériences pour l'identification du moteur CC.	40
9	Paramètres déterminés expérimentalement lors de l'identification du moteur CC.	46
10	Constantes du contrôleur PI et constantes utilisées dans la simulation de l'as- servissement en vitesse	49
11	Récapitulatif des composants électriques et électroniques utilisés dans la pla- teforme mobile.	54
12	Problèmes rencontrés et solutions apportées lors du développement du sys- tème électrique	56
13	Jeu de commandes du protocole série	61

1

ANALYSE FONCTIONNELLE

```

+++++[>+++++>+++++>+++++>+++++>+++++
++++>++++>++++>+>++++>++++>+
++++>++++>++++>++++>++++>+
++++>++++>++++>++++<<<<<<<<<<<-
]>----. .>.-.-.>-.>.>----.>.>+.>+.>+.>+.>+.-.-.>----.
>+.>..>+.>--..>+.
```

1.1 Présentation générale du projet

Le projet consiste à concevoir et réaliser une plateforme mobile robotisée à trois roues, dont la roue avant est à la fois motrice et directrice. Cette plateforme doit être capable de se déplacer de manière autonome tout en respectant un ensemble de contraintes fonctionnelles et techniques définies dans le cahier des charges. Le projet s'inscrit dans le cadre du premier projet d'ingénierie de la première année du cycle préparatoire intégré à l'EMINES, visant à acquérir des connaissances théoriques et pratiques en mécanique, électronique et informatique à travers un projet concret.

1.2 Rappel du cahier des charges fonctionnel

Le cahier des charges fonctionnel (CdCF) de la plateforme mobile a été établi lors de la première phase du projet. Il définit les exigences et les contraintes auxquelles la plateforme doit répondre pour satisfaire les besoins des utilisateurs finaux. Le CdCF est structuré autour des fonctions principales que la plateforme doit accomplir, ainsi que des critères de performance, de sécurité et d'ergonomie.

TABLEAU 1 – Tableau des exigences fonctionnelles et contraintes

Id	Critère	Description	Niveau/valeur	Unité	Flexibilité
<i>FS1</i> Se déplacer	<i>Déplacement</i>	Portée minimale	5	m	0
	<i>Vitesse</i>	Plage vitesses	0-150	mm/s	0
	<i>Précision</i>	Erreur linéaire maximale	1	cm/m	0
<i>FS2</i> Soulever	<i>Charge</i>	Masse à soulever	15	kg	0
	<i>Sustentation</i>	Superficie délimitée par les points de contact	300	cm ²	1 (±15%)
<i>FS3</i> Être autonome en énergie	<i>Énergie</i>	Autonomie minimale	15	min	0
<i>FS4</i> S'orienter	<i>Positionnement</i>	Rayon de giration (braquage)	0	m	0
<i>FC1</i> S'intégrer	<i>Encombrement</i>	Gabarit maximal L×l×H	50 × 30 × 30	cm ³	0
<i>FC2</i> Ne pas être lourd	<i>Ergonomie</i>	PTAC	25	kg	0

1.3 Diagramme des cas d'utilisation

Le diagramme des cas d'utilisation illustre les interactions entre l'utilisateur (acteur) et la plateforme mobile. Il identifie l'acteur principal (l'utilisateur) et définit le périmètre du système. Il met en évidence la fonction centrale de déplacement d'une charge.

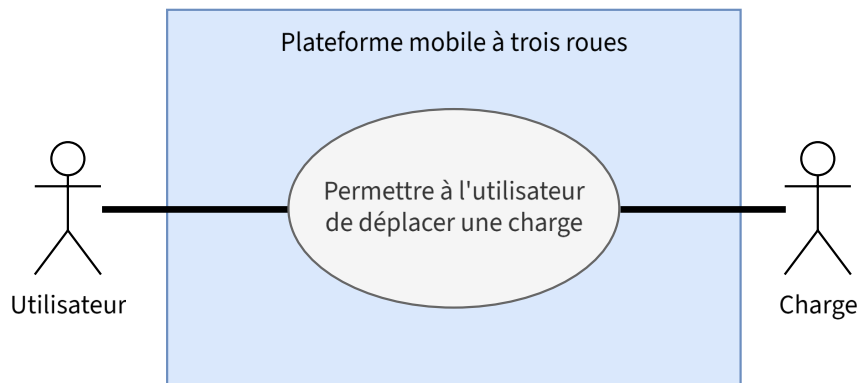


FIGURE 1 – Diagramme des cas d'utilisation de la plateforme mobile

1.4 Diagramme des exigences

Le diagramme des exigences présente une vue d'ensemble des exigences de la plateforme mobile, en les classant par catégories et en montrant leurs relations.

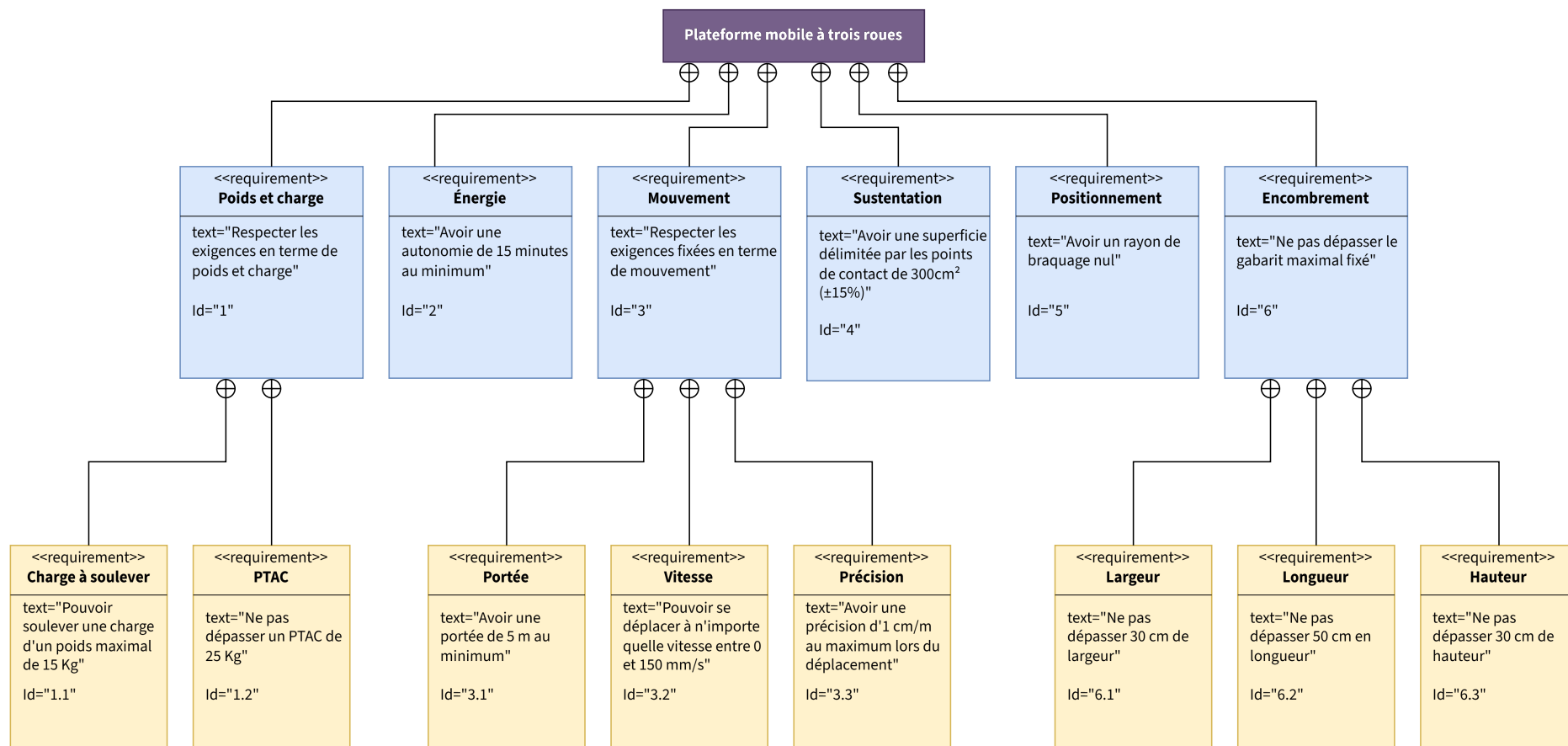


FIGURE 2 – Diagramme des exigences de la plateforme mobile

1.5 Chaîne fonctionnelle

La chaîne fonctionnelle de la plateforme mobile est séparée en deux chaînes principales : la chaîne d'énergie et la chaîne d'information. Elle décrit comment l'énergie et l'information circulent à travers le système pour accomplir les fonctions requises (Acquérir, Traiter, Communiquer - Alimenter, Distribuer, Convertir, Transmettre), en mettant en évidence les composants clés, les données et composants externes, et la fonction principale de déplacement d'une charge.

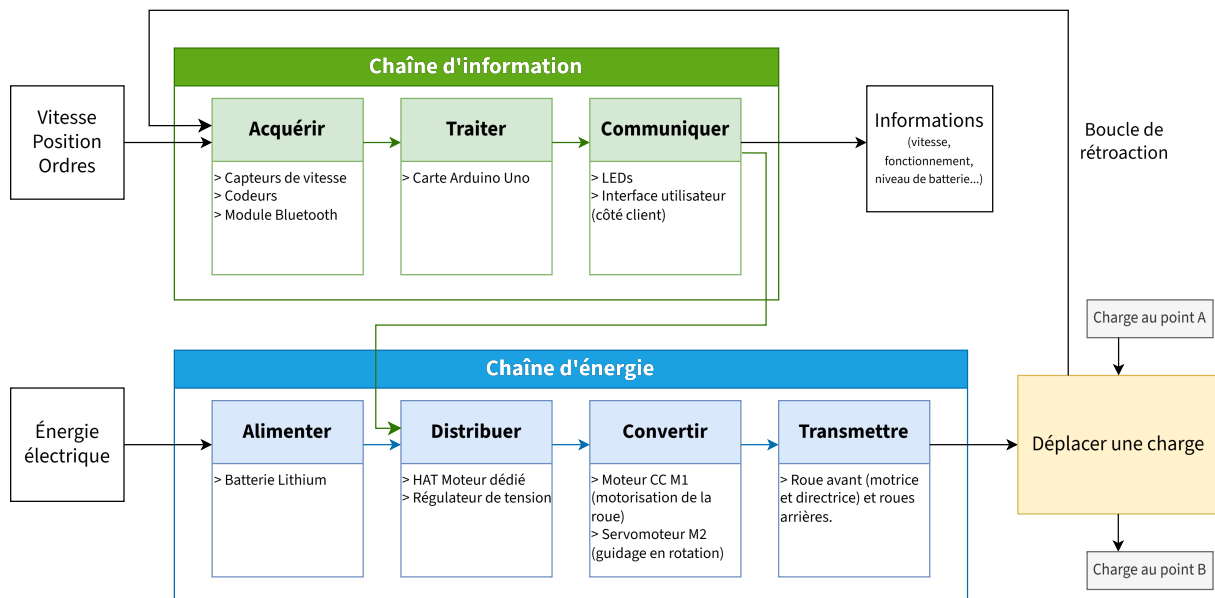


FIGURE 3 – Chaîne fonctionnelle de la plateforme mobile

1.6 Schéma synoptique

Le schéma synoptique de la plateforme mobile illustre les principaux composants et leurs interconnexions, en mettant en évidence les flux d'énergie et d'information. L'architecture matérielle s'articule autour du microcontrôleur qui centralise les entrées capteurs (Bluetooth, Codeurs) et pilote les actionneurs. Le schéma met en évidence la séparation entre le flux d'énergie (alimentation, motoréducteur) et le flux d'information (capteurs, microcontrôleur, communication Bluetooth).

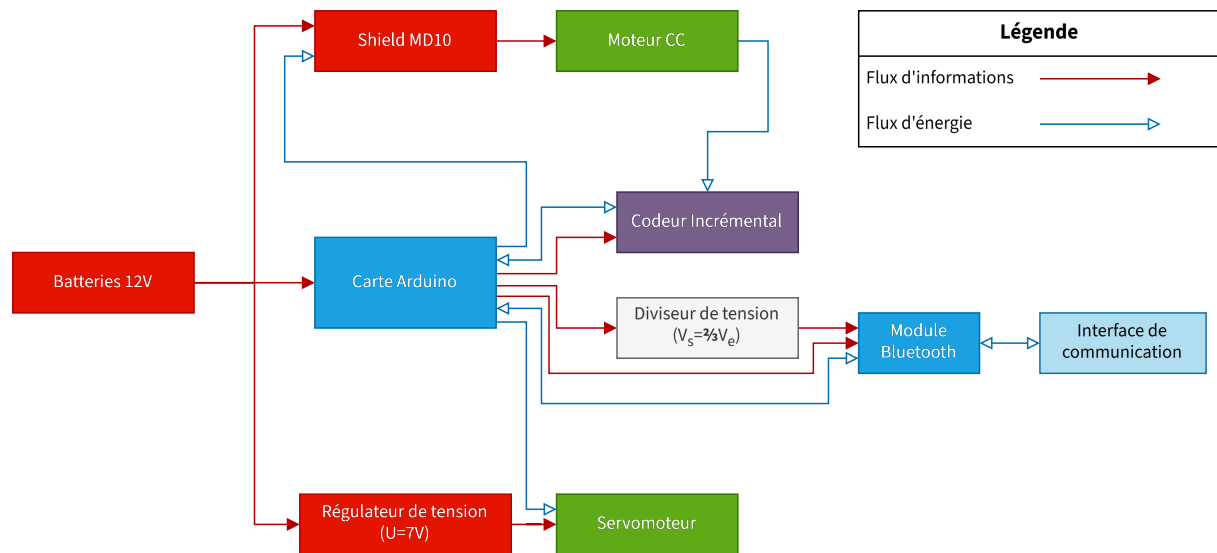


FIGURE 4 – Schéma synoptique de la plateforme mobile

1.7 Schéma cinématique

Le schéma cinématique de la plateforme mobile détaille les liaisons mécaniques entre les différents composants, en mettant en évidence les degrés de liberté et les mouvements possibles.

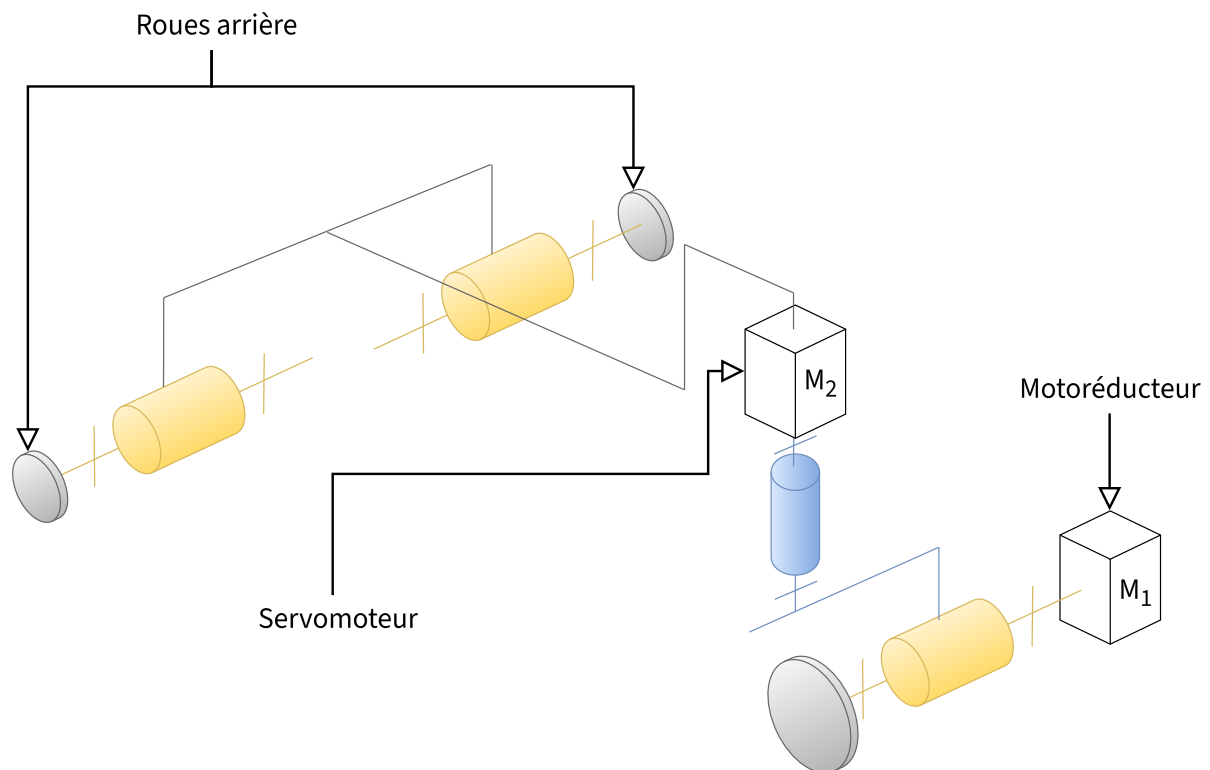


FIGURE 5 – Schéma cinématique de la plateforme mobile

2

CONCEPTION

+++++[>+++++>++++++>++++++>++++++>++++
+++>++++++>++++++>++++++>++++++>++++++
+++<<<<<<<-]>---.+.+.+.+.+.+.+.+.+.+.+.+



Afin d'optimiser l'opération de fabrication, les groupes de la classe ont été fusionnés en cinq équipes différentes, en fonction des similarités dans les maquettes CAO. La maquette présentée est le résultat du travail de l'équipe C, composée des groupes 3, 12 et 13.

La conception de la plateforme mobile a été réalisée à l'aide du logiciel de CAO SolidWorks. La maquette 3D est composée de trois sous-ensembles majeurs : le châssis, le porte-servomoteur, et le groupe de propulsion avant. La Figure 6 présente une vue isométrique de l'ensemble de la plateforme mobile.

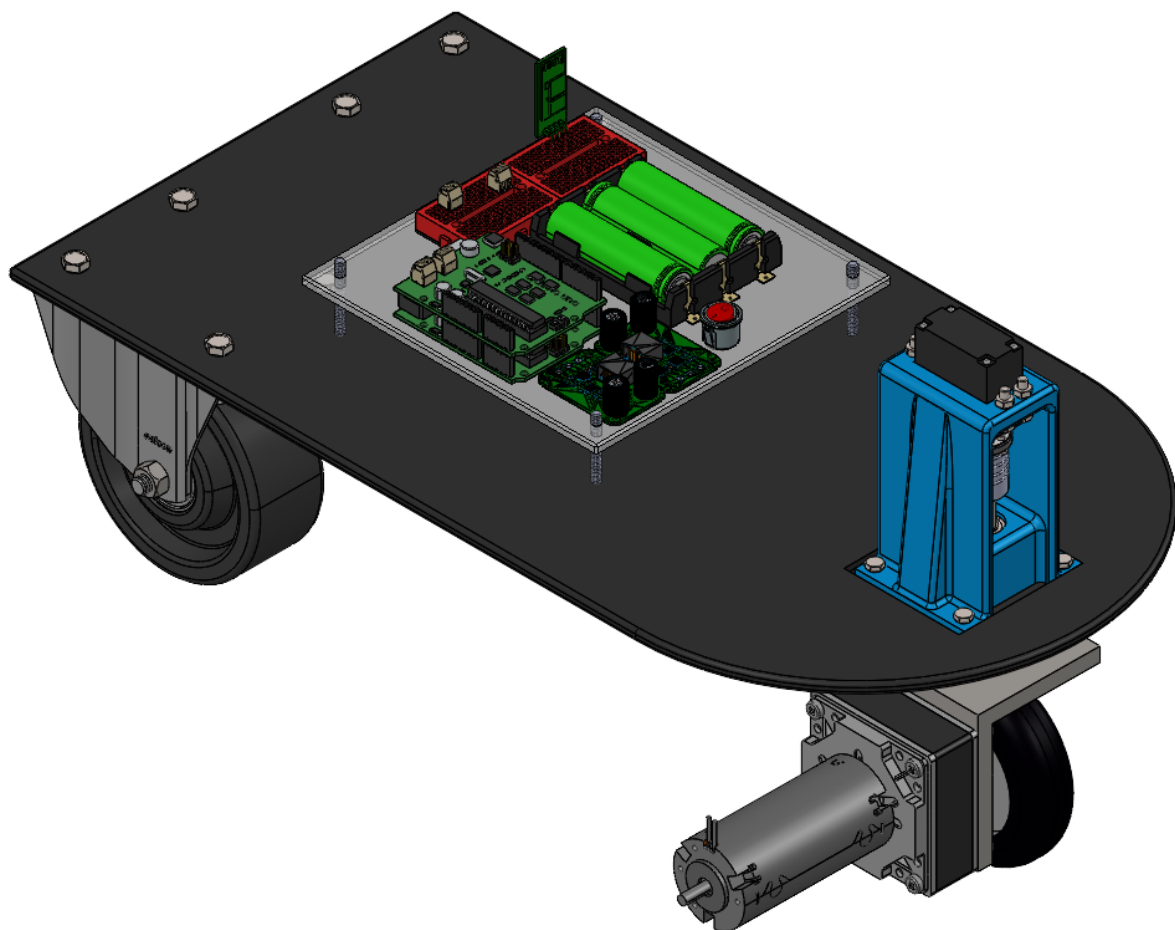


FIGURE 6 – Maquette CAO de la plateforme mobile

2.1 Sous-ensembles

Le sous-ensemble châssis (Figure 7) comprend le châssis principal en Compact HPL et les deux roues arrière.

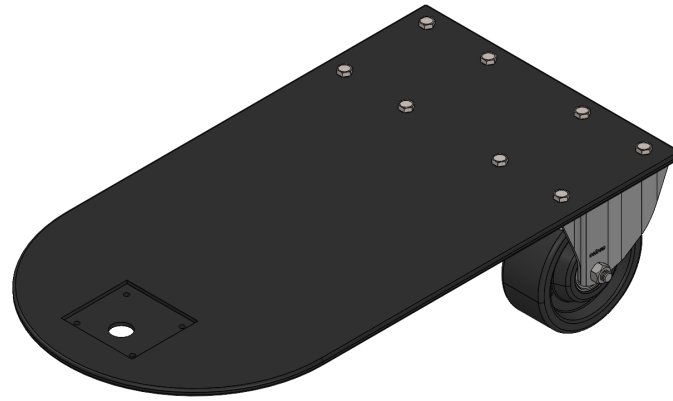


FIGURE 7 – Sous-ensemble châssis de la plateforme mobile

Le sous-ensemble porte-servomoteur (Figure 8) est constitué du porte-servomoteur (conçu pour être imprimé en 3D), du servomoteur, du palonnier et de l'arbre de guidage en rotation. Le porte-servomoteur permet de maintenir le servomoteur en position sur le châssis, tandis que l'accouplement flexible assure la liaison entre l'arbre du servomoteur (par l'intermédiaire du palonnier) et l'arbre de guidage en rotation, permettant ainsi la transmission du guidage en rotation.

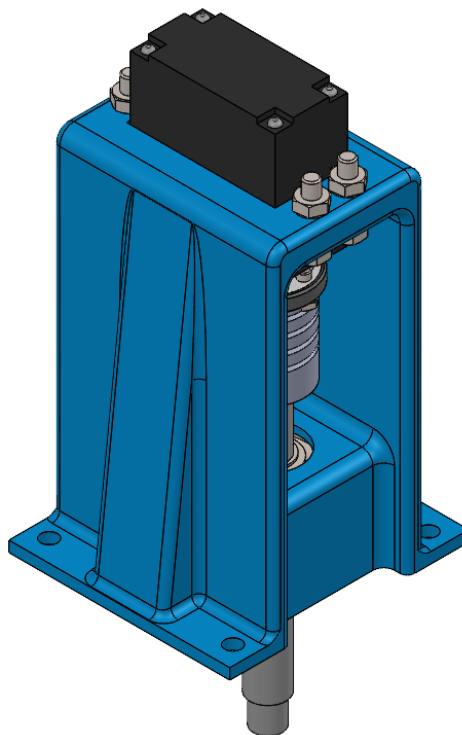


FIGURE 8 – Sous-ensemble porte-servomoteur de la plateforme mobile

Le sous-ensemble groupe de propulsion avant (Figure 9) est constitué du motoréducteur, de la cornière, du moyeu, de la roue avant et du système de pincement SHF. La cornière permet de fixer l'arbre de guidage en rotation au reste du groupe de propulsion avant, par l'intermédiaire du pincement SHF10, tandis que le moyeu assure la liaison entre la roue avant et l'arbre du motoréducteur.

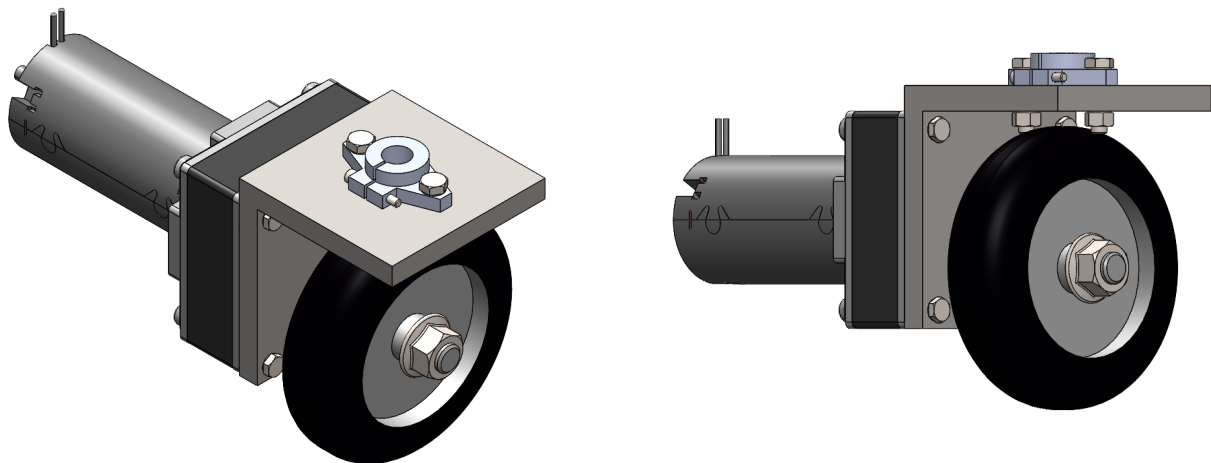


FIGURE 9 – Sous-ensemble groupe de propulsion avant de la plateforme mobile (vue isométrique et vue de coin)

2.2 Pièces principales

TABEAU 2 – Récapitulatif des pièces principales de la plateforme mobile

Pièce	Description
Châssis	Pièce principale en Compact HPL qui constitue la structure de la plateforme mobile. Elle dispose d'un chambrage dédié au porte-servomoteur et de perçages pour les roues arrière et le porte-servomoteur.
Porte-servomoteur	Pièce imprimée en 3D en PLA qui maintient le servomoteur en position sur le châssis et accueille les roulements pour l'arbre de guidage en rotation.
Palonnier	Pièce intermédiaire entre l'arbre du servomoteur et l'accouplement flexible.
Arbre de guidage en rotation	Pièce fabriquée à partir d'un brut d'aluminium, qui transmet le guidage en rotation depuis le servomoteur jusqu'au groupe de propulsion avant, avec des roulements pour assurer une rotation fluide.
Cornière	Pièce en alliage d'aluminium qui fixe l'arbre de guidage en rotation au reste du groupe de propulsion avant, en utilisant le système de pincement SHF10.
Moyeu	Pièce en aluminium qui assure la liaison entre la roue avant et l'arbre du motoréducteur.
Plaque de fixation des composants électroniques	Pièce en plexiglas qui permet de fixer les composants électroniques (carte microcontrôleur, driver de moteur, batterie) sur le châssis de la plateforme mobile.

2.2.1 Châssis

Le châssis de la plateforme mobile a été conçu en Compact HPL, un matériau robuste et léger. Il dispose d'un chambrage dédié au porte-servomoteur pour assurer la mise en position de ce dernier. Des perçages ont aussi été prévus pour maintenir les roues arrières et le porte-servomoteur en position.

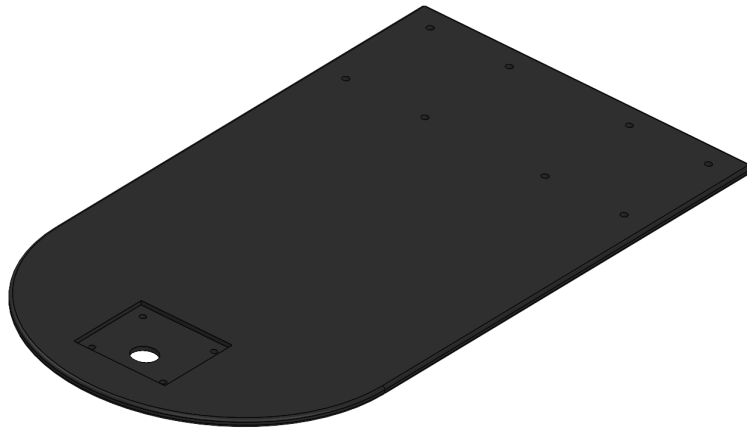


FIGURE 10 – Vue isométrique du châssis

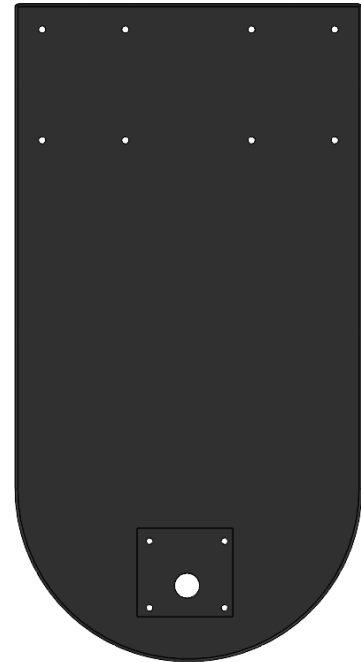


FIGURE 11 – Vue de dessus du châssis

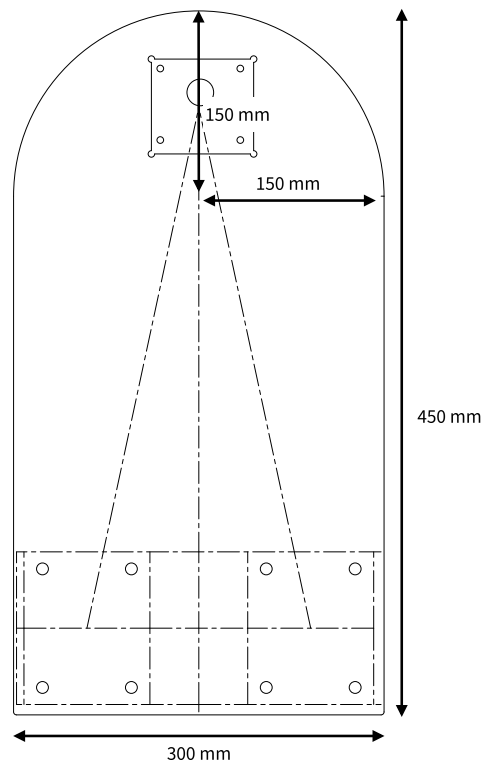


FIGURE 12 – Mise en plan de la pièce châssis avec dimensions

2.2.2 Porte-servomoteur

Le porte-servomoteur a été conçu pour être imprimé en 3D avec du plastique PLA. Il permet de maintenir le servomoteur en position sur le châssis et d'accueillir les roulements nécessaires pour l'arbre de guidage en rotation.

Les Figures 13, 14, 15 et 16 présentent le porte-servomoteur sous différents angles, mettant en évidence les caractéristiques de conception, notamment les logements pour les roulements visibles dans la vue de bas.

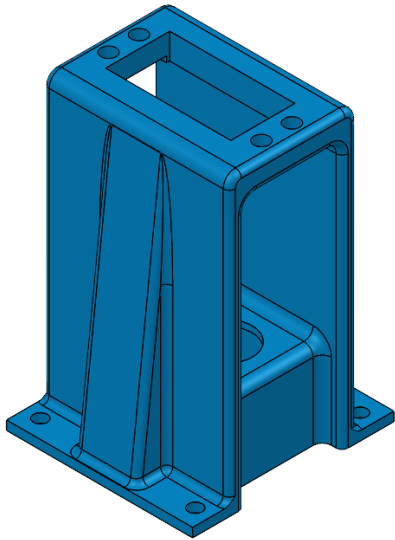


FIGURE 13 – Vue isométrique 1 (coin avant) du porte-servomoteur

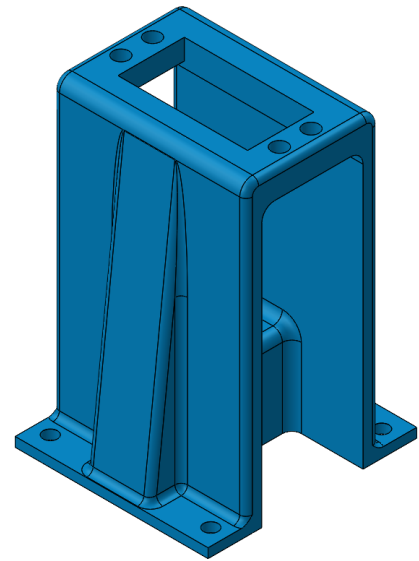


FIGURE 14 – Vue isométrique 2 (coin arrière) du porte-servomoteur

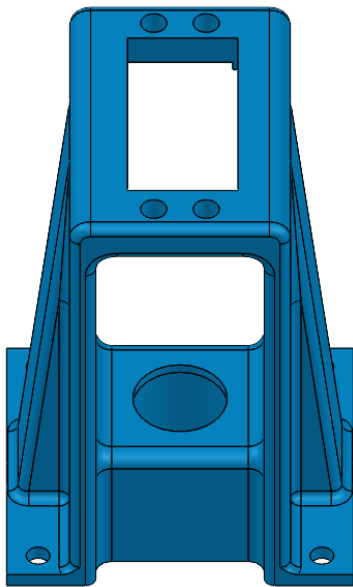


FIGURE 15 – Vue de face du porte-servomoteur

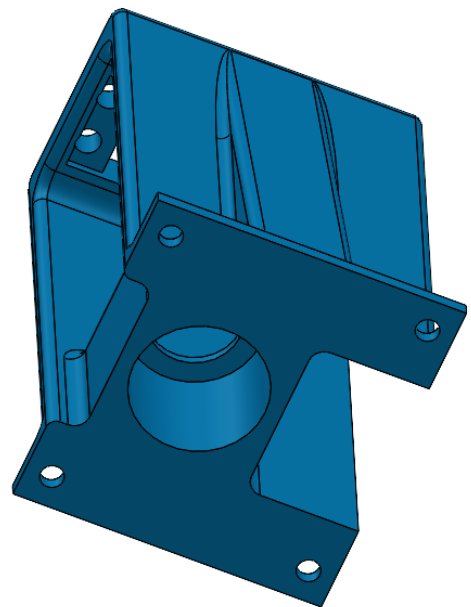


FIGURE 16 – Vue de bas du porte-servomoteur, avec les logements pour les roulements

2.2.3 Palonnier

Le palonnier est une pièce intermédiaire entre l'arbre du servomoteur et l'accouplement flexible. Il permet de réduire le diamètre de l'arbre du servomoteur pour s'adapter à l'accouplement, de $\varnothing 20$ mm à $\varnothing 5$ mm. Il a été originellement conçu pour être imprimé en 3D, mais des problèmes de fixation ont conduit à la fabrication d'une nouvelle pièce en aluminium. Il dispose de quatre perçages pour la fixation par vis-écrou.

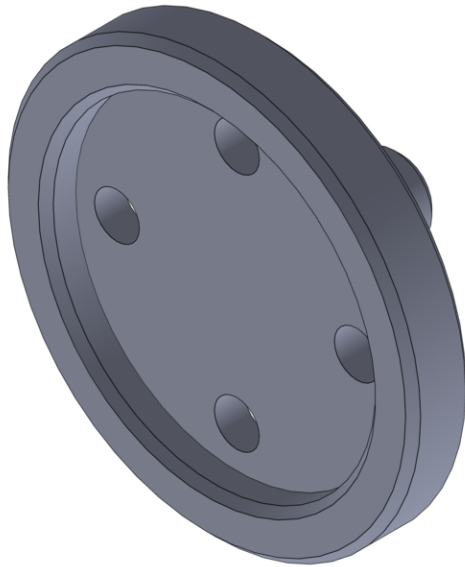


FIGURE 17 – Vue orthogonale du palonnier

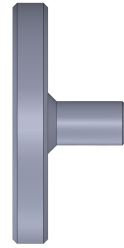


FIGURE 18 – Vue de côté du palonnier

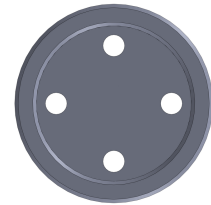


FIGURE 19 – Vue de face du palonnier

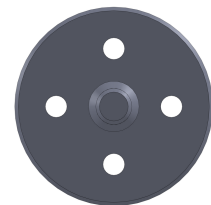


FIGURE 20 – Vue arrière du palonnier

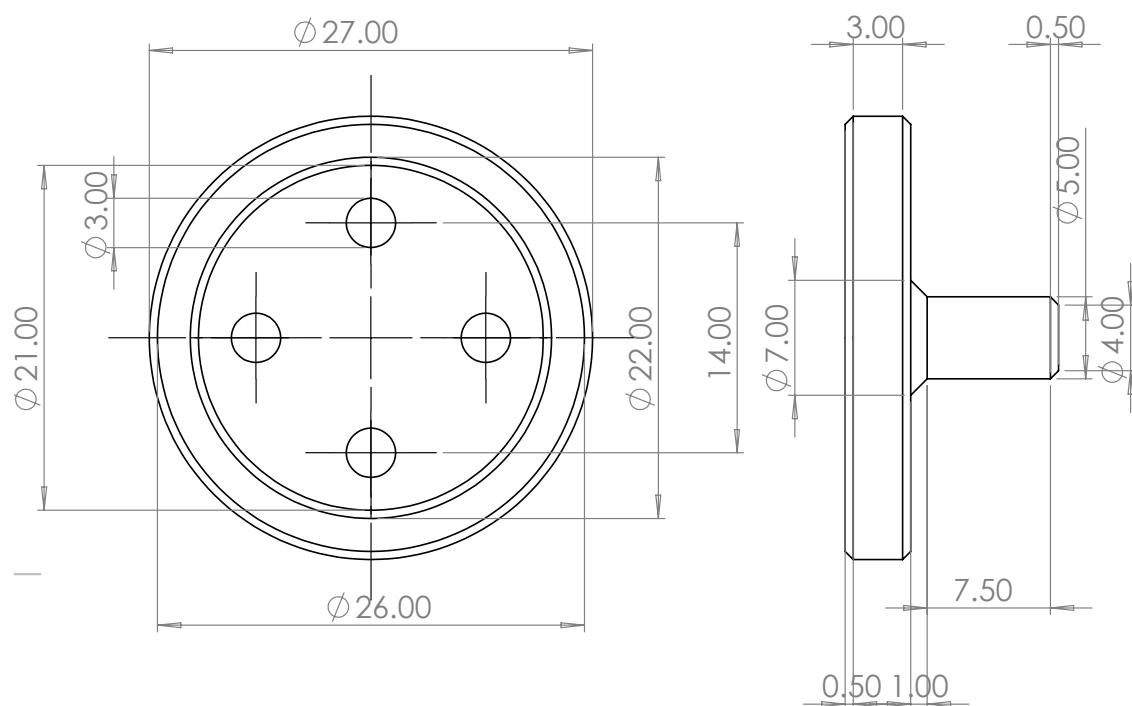


FIGURE 21 – Mise en plan de la pièce palonnier avec dimensions

2.2.4 Arbre de guidage en rotation

L'arbre de guidage en rotation a été conçu pour être fabriqué à partir d'un brut d'acier de diamètre 10,7 mm. Il permet de guider en rotation depuis le servomoteur jusqu'au groupe de propulsion avant. Des roulements situés à l'intérieur du porte-servomoteur assurent la rotation fluide de l'arbre et minimisent l'influence des défauts d'alignement. Une partie de l'arbre est usinée en diamètre 8 mm pour s'adapter à l'accouplement flexible, tandis que son extrémité opposée et une partie centrale en diamètre d'origine 10,7 mm pour agir comme un épaulement et assurer l'arrêt axial de l'arbre dans les roulements. Deux autres parties de l'arbre sont usinées pour accueillir des circlips, qui empêchent l'arbre de sortir axialement des roulements.

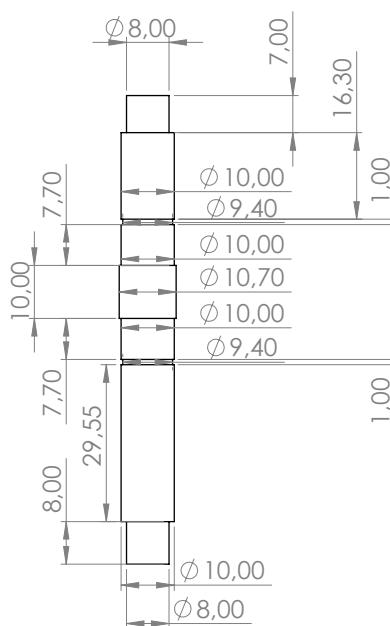


FIGURE 22 – Mise en plan de l'arbre de guidage en rotation avec dimensions

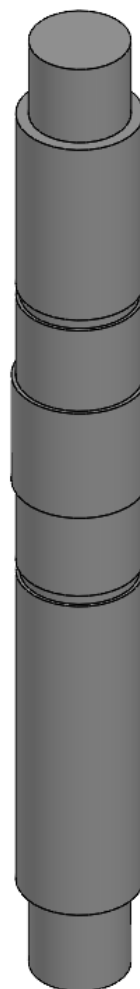


FIGURE 23 – Rendu de l'arbre

2.2.5 Cornière

La cornière est utilisée pour fixer l'arbre de guidage en rotation au groupe de propulsion avant. Elle a été conçue pour être usinée dans un brut d'alliage d'aluminium aux dimensions de 80 mm × 80 mm × 8 mm, disponible au fablab. Elle comporte des perçages pour la fixation (par vis-écrou) du pincement SHF10, ainsi qu'un trou pour la mise en position de l'arbre de guidage en rotation. Elle comporte aussi des perçages pour la fixation du motoréducteur.

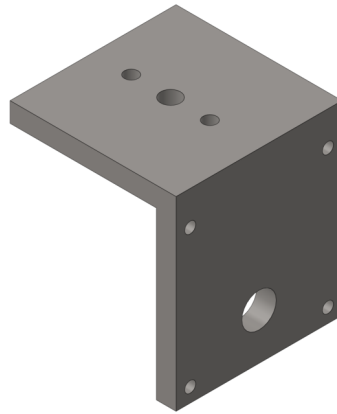


FIGURE 24 – Pièce cornière de la plateforme mobile

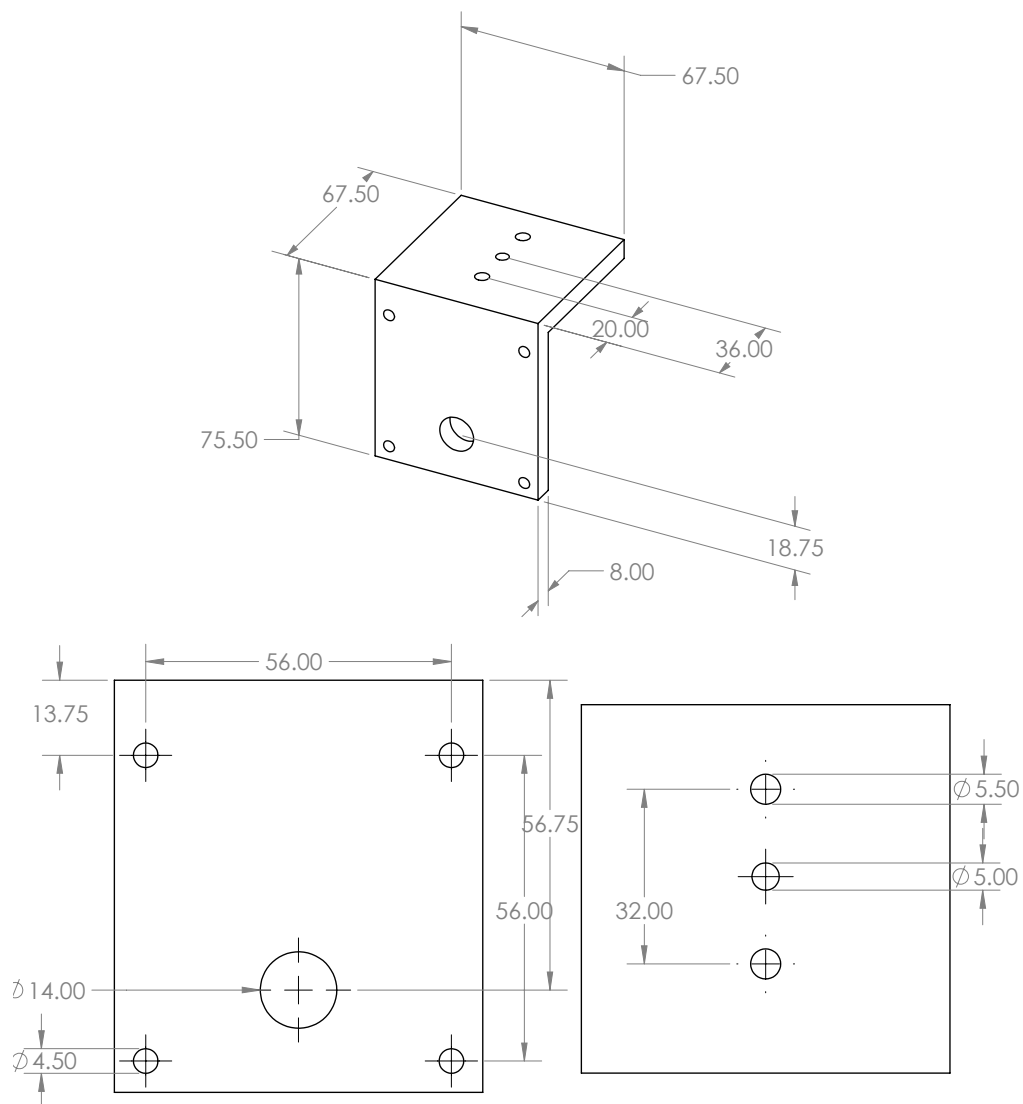


FIGURE 25 – Mise en plan de la pièce cornière avec dimensions

2.2.6 Moyeu

Le moyeu assure la liaison entre la roue avant et l'arbre du motoréducteur. Il a été conçu pour être usiné dans un brut d'aluminium, avec deux perçages pour accueillir deux vis de pression M3 au niveau de l'arbre du motoréducteur, qui possède un méplat afin de garantir un assemblage sécurisé et éviter le glissement. Il dispose aussi d'une surface filetée pour la fixation de la roue avant via écrou M12 d'un côté, et surface latérale de l'autre.



FIGURE 26 – Vue isométrique du moyeu



FIGURE 27 – Vue latérale du moyeu



La surface filetée pour la fixation de la roue avant n'est pas visible sur la figure en raison d'une simplification du modèle 3D.

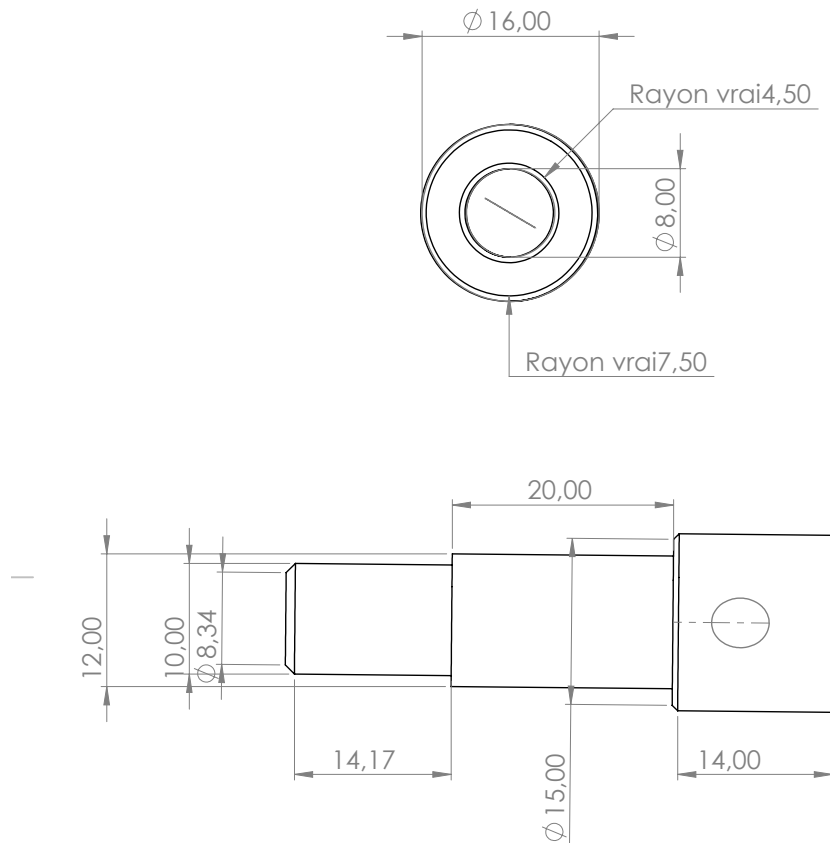


FIGURE 28 – Mise en plan de la pièce moyeu avec dimensions

2.2.7 Plaque de fixation des composants électroniques

La plaque de fixation des composants électroniques est fabriquée en plexiglas. Elle permet de fixer la carte Arduino Uno, le hacheur et les batteries sur le châssis de la plateforme mobile. Des perçages ont été prévus pour la fixation par vis-écrou, ainsi que pour la fixation des composants sur la plaque.

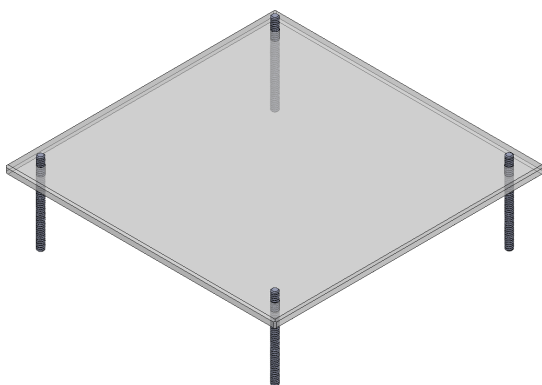


FIGURE 29 – Plaque de fixation (vide et sans perçages) - Vue isométrique

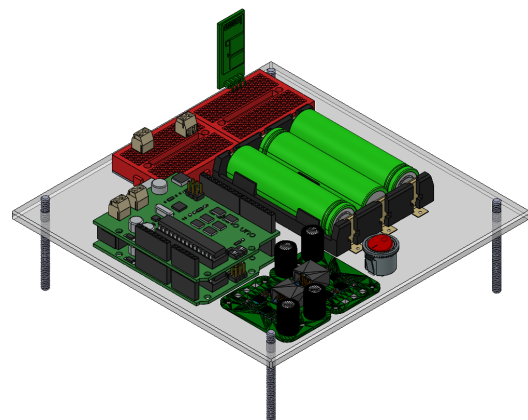


FIGURE 30 – Plaque de fixation (avec composants) - Vue isométrique

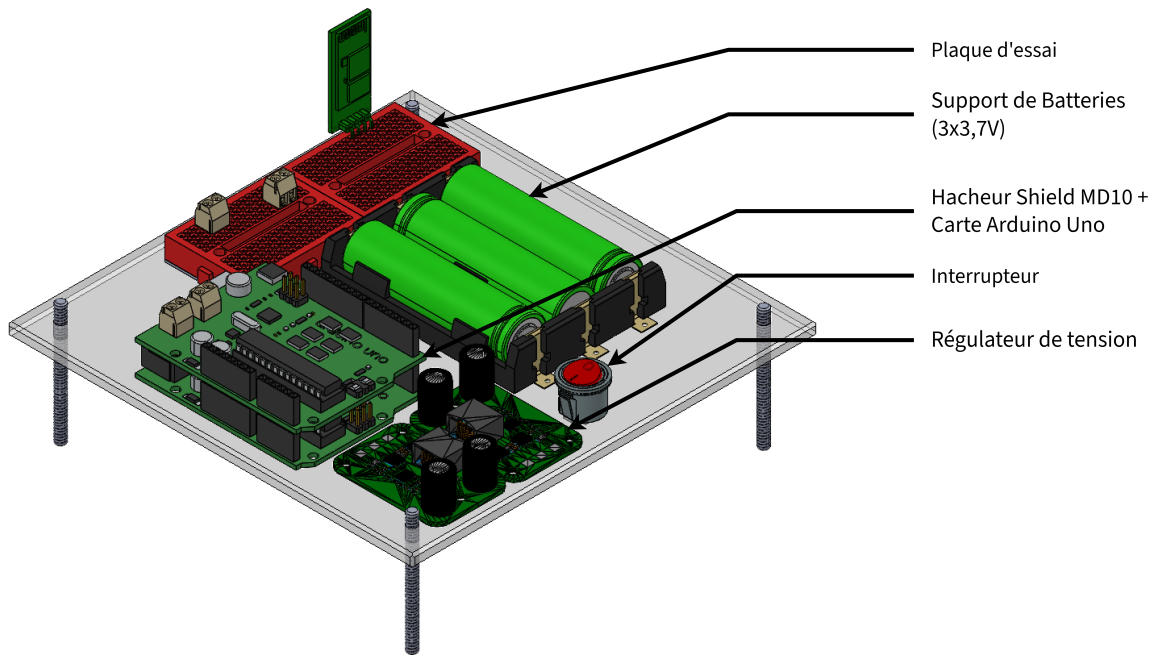


FIGURE 31 – Plaque de fixation des composants électroniques - Vue isométrique légendée

2.3 Liaisons mécaniques

Le tableau ci-dessous récapitule les liaisons mécaniques principales entre les différentes pièces de la plateforme mobile.

TABLEAU 3 – Récapitulatif des liaisons mécaniques principales de la plateforme mobile

Pièce 1	Pièce 2	Type de liaison	Description
Roues arrière	Châssis	ϕ Pivot	Rotation assurée par roulements intégrés. Axe encastré sur le châssis, mise en position par appui plan (surface du châssis + axe) et maintien en position par vis-écrou CHC M8 avec rondelles
Porte-servomoteur	Châssis	$\perp$ Encastrement	Mise en position par surfaces latérales (chambrage du châssis + surfaces du porte-servomoteur) + maintien en position par vis-écrou M4
Palonnier	Arbre servomoteur	$\perp$ Encastrement	Mise en position par surfaces latérales + maintien en position par vis-écrou M3
Arbre de guidage en rotation	Palonnier	$\perp$ Encastrement	Mise en position par surfaces latérales + maintien en position par accouplement flexible
Arbre de guidage en rotation	Porte-servomoteur	ϕ Pivot	Mise en position par roulements (surfaces latérales) + maintien en position par surface latérale du porte-servomoteur
Cornière	Arbre de guidage en rotation	ϕ Pivot	Mise en position par trou dans la cornière + maintien en position par pincement SHF10
Pincement SHF10	Cornière	$\perp$ Encastrement	Mise en position par appui plan + maintien en position par vis-écrou M4
Moyeu	Roue avant	$\perp$ Encastrement	Mise en position par surface latérale + maintien en position par écrou M12
Moyeu	Arbre motoréducteur	$\perp$ Encastrement	Mise en position par surface latérale + maintien en position par vis de pression
Cornière	Motoréducteur	$\perp$ Encastrement	Mise en position par surface latérale + maintien en position par vis

2.4 Coupes importantes

2.4.1 Coupe au niveau des roulements de l'arbre de guidage en rotation dans le porte-servomoteur

La Figure 33 présente une coupe au niveau des roulements de l'arbre de guidage en rotation dans le porte-servomoteur. Les roulements sont montés avec un ajustement serré dans le porte-servomoteur et sont maintenus en position axiale par un épaulement de l'arbre de guidage, ainsi que par le châssis en bas et le porte-servomoteur en haut. Des circlips sont également utilisés sur l'arbre de guidage pour assurer un maintien supplémentaire en position axiale. L'utilisation d'un double roulement permet de mieux répartir les charges exercées sur l'arbre de guidage.

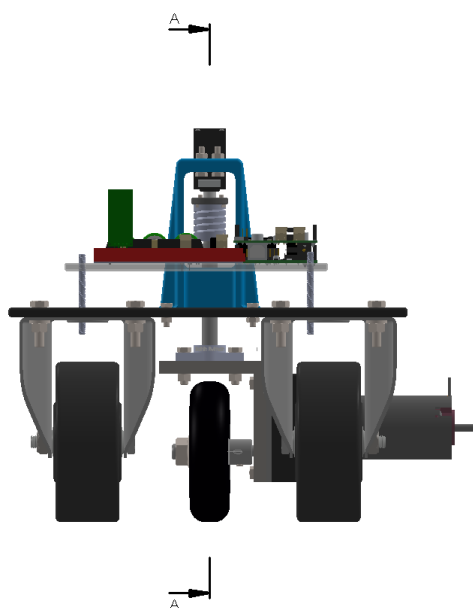


FIGURE 32 – Plan de coupe au niveau des roulements de l'arbre de guidage en rotation dans le porte-servomoteur

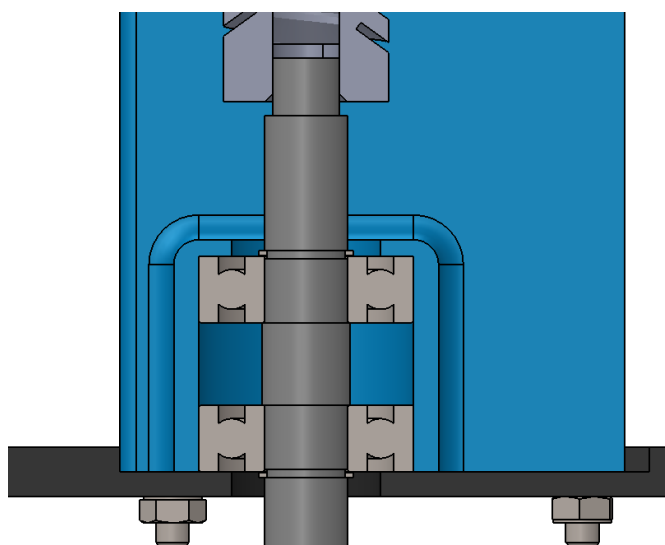


FIGURE 33 – Coupe au niveau des roulements de l'arbre de guidage en rotation dans le porte-servomoteur

TABEAU 4 – Caractéristiques des roulements utilisés pour l'arbre de guidage en rotation

Type	Diamètre intérieur (mm)	Diamètre extérieur (mm)	Largeur (mm)
6000	10	26	8

2.4.2 Coupe au niveau du moyeu de la roue avant

La Figure 35 présente une coupe au niveau du moyeu de la roue avant. Le moyeu est maintenu en position axiale sur l'arbre du motoréducteur par des vis de pression, tandis que la roue avant est fixée en position axiale sur le moyeu à l'aide d'un écrou M12 vissé sur une surface filetée.

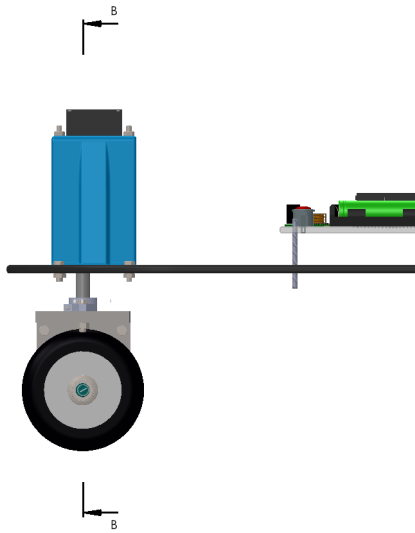


FIGURE 34 – Plan de coupe au niveau du moyeu de la roue avant

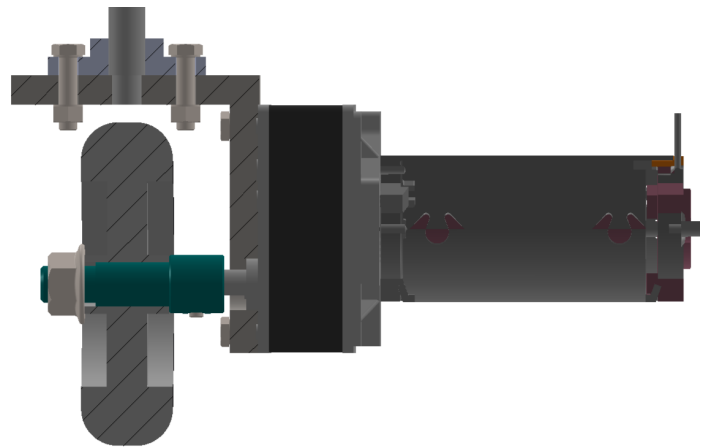


FIGURE 35 – Coupe au niveau du moyeu de la roue avant

3

FABRICATION

+++++++[>++++++>+++++++>+++++++>+++++++>+++++++>+++++++
+++>+++++++>+++++++>+++++++>+++++++>+++++++>+++++++
++>+++++++>+<<<<<<<<<<-]>.>---.>--.>++++.>-----.>-.>
---.>----.>-----.>+.>.>++++.>.

3.1 Contraintes de fabrication

Lors de la conception de la plateforme mobile, plusieurs contraintes de fabrication ont été prises en compte afin d'assurer la faisabilité et la robustesse du produit final.

- **Adaptation au stock disponible (Arbre de guidage) :** Initialement conçue pour un arbre standard de $\varnothing$ 10 mm, la conception a dû être révisée pour s'adapter au brut d'acier spécifique fourni par le laboratoire, mesuré à $\varnothing$ 10,7 mm. Cette contrainte d'approvisionnement a nécessité une mise à jour paramétrique des pièces liées (porte-servomoteur, cornière) pour garantir les ajustements.
- **Précision de l'imprimante 3D :** Le porte-servomoteur a été conçu en tenant compte des limites de précision de l'imprimante 3D utilisée. Des tolérances appropriées ont été intégrées dans la conception pour garantir un ajustement correct des composants imprimés.
- **Dimensions de la cornière brute :** La cornière utilisée pour la fixation de l'arbre de guidage en rotation au groupe de propulsion avant a été définie par les dimensions de celle disponible au fablab, qui étaient de 80 mm $\times$ 80 mm $\times$ 8 mm.

3.2 Récapitulatif des étapes de fabrication

Le diagramme de flux ci-dessous illustre les différentes étapes de fabrication des pièces de la plateforme mobile. Il est décomposé en trois phases clés : la conception et préparation des fichiers de fabrication, la fabrication des pièces, et l'assemblage et la validation finale. Les différentes sous-étapes liées aux deux premiers processus sont détaillées.

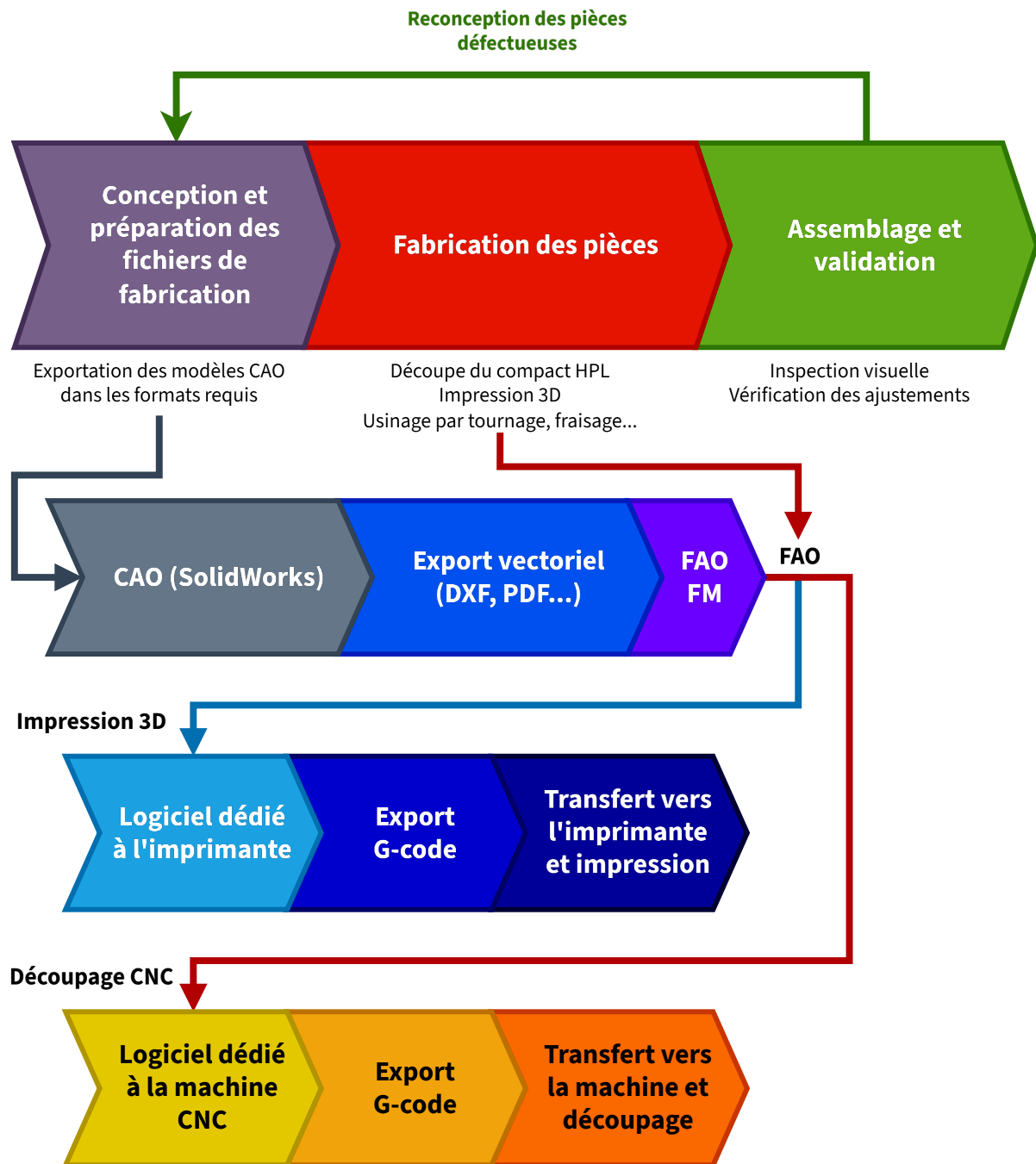


FIGURE 36 – Diagramme de flux des étapes de fabrication des pièces de la plateforme

3.3 Pièces fabriquées

3.3.1 Récapitulatif des pièces fabriquées

Pièce	Matériau	Procédé de fabrication
Châssis	Compact HPL	Découpe CNC
Porte-servomoteur	Plastique PLA	Impression 3D
Palonnier	Aluminium	Tournage/Fraisage
Arbre de guidage en rotation	Acier	Tournage/Fraisage
Cornière	Alliage d'aluminium	Perçage
Moyeu	Aluminium	Tournage/Fraisage
Plaque de fixation des appareils électroniques	Plexiglas	Découpe/Perçage

TABLEAU 5 – Récapitulatif des pièces fabriquées, matériaux et procédés de fabrication.

3.3.2 Châssis

Le châssis a été fabriqué en Compact HPL au fablab, en utilisant une machine CNC pour la découpe. Les congés prévus dans la conception initiale ont été omis lors de la fabrication, et des trous ont été ajoutés aux coins du chambrage dédié au porte-servomoteur pour faciliter la découpe, compte tenu de la difficulté de réaliser des angles droits avec la CNC.



FIGURE 37 – Châssis réel de la plateforme mobile (avec roues arrières montées)



FIGURE 38 – Vue de dessus du châssis



FIGURE 39 – Vue latérale du châssis



FIGURE 40 – Vue du châssis avec les roues arrière montées

Choix du matériau pour le châssis

Bien que le prototypage initial envisageait l'usage de bois, le choix du matériau s'est porté sur le compact HPL en raison de contraintes d'approvisionnement. Ce changement s'est avéré bénéfique, car le compact HPL offre une robustesse et une durabilité supérieures tout en conservant une excellente usinabilité pour la découpe CNC.

3.3.3 Porte-servomoteur

Le porte-servomoteur a été imprimé en 3D au fablab en utilisant du plastique PLA. Lors de l'assemblage, il a été constaté que les roulements étaient très difficiles à insérer dans leur logement. Pour faciliter l'insertion, un léger élargissement du trou dédié aux roulements a été effectué, ainsi que l'utilisation d'un lubrifiant.



FIGURE 41 – Porte-servomoteur réel de la plateforme mobile



FIGURE 42 – Vue arrière du porte-servomoteur



FIGURE 43 – Vue de face du porte-servomoteur



FIGURE 44 – Vue de dessus du porte-servomoteur

3.3.4 Cornière

La cornière a été fabriquée au fablab en utilisant des techniques de tournage et de fraisage. La cornière a d'abord été sciée aux bonnes dimensions avant d'être usinée. Un problème a été rencontré lors de l'assemblage, où la roue avant et le moyeu risquaient de toucher l'une des surfaces de la cornière. Pour résoudre ce problème, un léger chambrage a été ajouté à ces zones afin d'éviter tout contact.

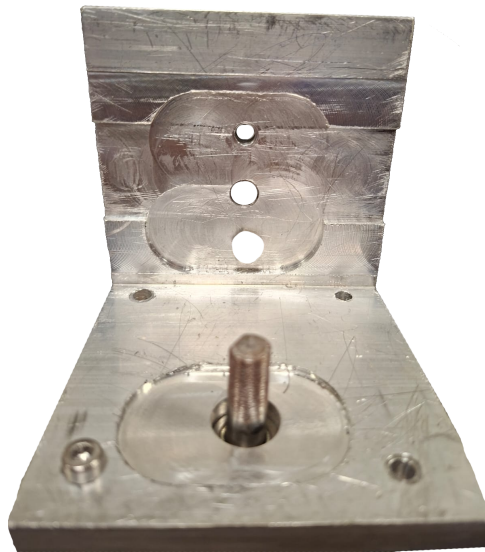


FIGURE 45 – Cornière réelle de la plateforme mobile (avec arbre moteur inséré)

3.3.5 Moyeu

Le moyeu a été fabriqué au fablab en utilisant des techniques de tournage et de fraisage. Aucun problème particulier n'a été rencontré lors de sa fabrication.



FIGURE 46 – Moyeu fabriqué

3.3.6 Arbre de guidage en rotation

L'arbre de guidage en rotation a été fabriqué au fablab en utilisant des techniques de tournage et de fraisage. Aucun problème particulier n'a été rencontré lors de sa fabrication.



FIGURE 47 – Arbre de guidage en rotation fabriqué

3.3.7 Palonnier

Le palonnier a été initialement conçu pour être imprimé en 3D en plastique PLA. Cependant, des problèmes de fixation ont été rencontrés, car la pièce s'est avérée trop fragile et s'est cassée facilement. Après deux tentatives infructueuses, il a été décidé de refabriquer le palonnier en aluminium. Des jeux supplémentaires ont été ajoutés pour faciliter son insertion sur l'arbre du servomoteur.

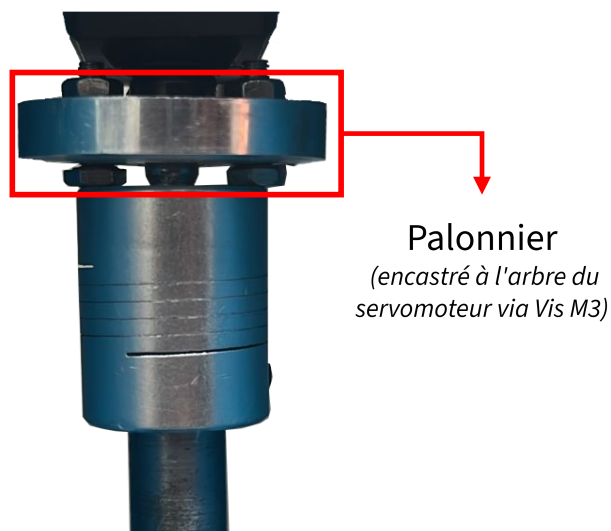


FIGURE 48 – Palonnier fabriqué et assemblé (l'intégration immédiate n'a pas permis de prise de vue isolée)

3.3.8 Plaque de fixation des appareils électroniques

La plaque de fixation des appareils électroniques est une plaque en plexiglas découpée et percée au fablab. Elle mesure 167 mm de long, 161 mm de large et a une épaisseur de 4 mm. Elle est fixée légèrement au-dessus du châssis à l'aide de vis-écrous, permettant ainsi de fixer les appareils électroniques tels que la carte mère et la batterie. Son intégration a été validée dans la maquette numérique, mais elle a été fournie par le fablab.



FIGURE 49 – Plaque de fixation des appareils électroniques (vide)

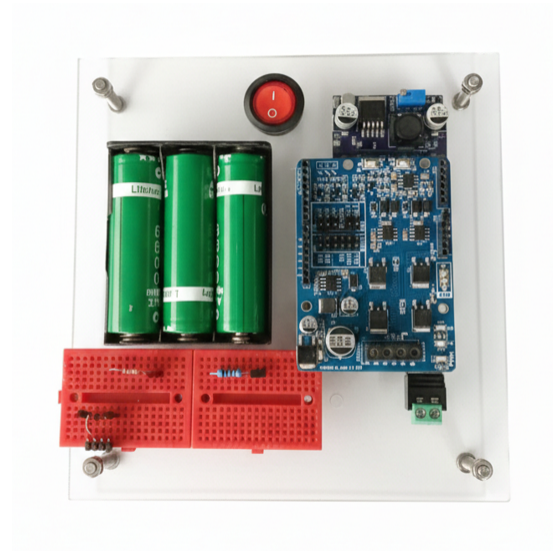


FIGURE 50 – Plaque de fixation des appareils électroniques (avec composants montés)

3.3.9 Récapitulatif des problèmes rencontrés lors de la fabrication

Pièce	Problème rencontré	Solution apportée
Palonnier	Plastique PLA fragile et facilement fracturable, jeu très serré causant des difficultés de fixation	Refonte complète de la pièce en aluminium avec des jeux adaptés
Porte-servomoteur	Difficulté d'insertion des roulements, manque de flexibilité pour le montage	Élargissement très léger du trou dédié aux roulements et utilisation d'un lubrifiant pour faciliter l'insertion
Châssis	Erreurs d'alignement lors du perçage des trous pour le porte-servomoteur	Ajustement manuel lors de l'assemblage final
Cornière	Risque de contact entre la roue avant et le bas de la partie supérieure de la cornière	Ajout d'un léger chambrage pour éviter tout contact

TABLEAU 6 – Problèmes rencontrés lors de la fabrication et solutions apportées.

3.3.10 Récapitulatif des vis et éléments de fixation utilisés

Élément de fixation	Longueur	Quantité
Vis M4	4 mm	6
Vis M4	8 mm	4
Vis M4	12 mm	4
Vis M4	16 mm	4
Vis M4	20 mm	4
Vis M5	20 mm	2
Vis M5	60 mm	4
Vis CHC M8	25 mm	8
Vis de pression	19 mm	5
Vis M2.5	8 mm	2
Écrous M4	N/A	16
Écrous M12	N/A	1
Rondelles M8	N/A	8

TABLEAU 7 – Récapitulatif des vis et éléments de fixation utilisés.

3.4 Détail de fabrication d'une pièce : Porte-servomoteur

3.4.1 Préparation du fichier de fabrication

Le porte-servomoteur a été conçu à l'aide du logiciel SolidWorks, où toutes les dimensions et caractéristiques nécessaires ont été définies. Une fois la conception terminée, le modèle 3D a été exporté au format STL, qui est couramment utilisé pour l'impression 3D.

3.4.2 Paramétrage de l'impression 3D

Ce fichier STL a ensuite été importé dans le logiciel Bambu Studio, qui est le logiciel de préparation d'impression pour l'imprimante 3D Bambu Lab X1 Carbon. La préparation de l'impression (slicing) a été réalisée via le logiciel *Bambu Studio*. Afin de garantir un résultat fiable et reproductible, nous avons opté pour le **profil d'impression standard recommandé par le constructeur** pour le filament PLA sur la *Bambu Lab X1 Carbon*.

Ce profil par défaut privilégie un compromis optimal entre vitesse et qualité dimensionnelle, caractérisé typiquement par :

- **Hauteur de couche** : Standard (0.20 mm), offrant une résolution suffisante pour les ajustements mécaniques.
- **Remplissage (Infill)** : Densité standard (type Grid ou Gyroid) assurant la rigidité structurelle nécessaire au maintien du servomoteur.
- **Matériau** : PLA+ Générique (température d'extrusion gérée automatiquement par le profil machine).

Le fichier a été ensuite exporté au format G-code, qui contient les instructions spécifiques pour l'imprimante 3D. Le fichier G-code a ensuite été transféré sur une carte SD, qui a été insérée dans l'imprimante 3D Bambu Lab X1 Carbon pour lancer le processus d'impression.

3.4.3 Assemblage et validation

L'assemblage du porte-servomoteur a débuté par l'insertion du servomoteur dans son logement. Cette étape a présenté des difficultés en raison d'un ajustement très serré, mais le servomoteur a finalement pu être inséré avec précaution.

L'insertion des roulements dans le porte-servomoteur a été particulièrement problématique. Les roulements étaient extrêmement difficiles à insérer dans leurs logements prévus. Pour surmonter cette difficulté, un léger perçage a été effectué pour élargir très légèrement les trous dédiés aux roulements. De plus, l'utilisation d'un lubrifiant a grandement facilité l'insertion des roulements, permettant ainsi de les positionner correctement sans endommager la pièce.

Enfin, l'assemblage avec l'arbre de guidage en rotation a également été délicat en raison de l'espace limité disponible dans le porte-servomoteur, conçu pour être aussi compact que possible. **Malgré ces problèmes, l'assemblage final a été réalisé avec succès, et le porte-servomoteur a été validé pour son bon fonctionnement au sein de la plateforme mobile.**

La Figure 51 présente un diagramme récapitulatif des différentes étapes de fabrication du porte-servomoteur, depuis la conception initiale jusqu'à l'assemblage final.

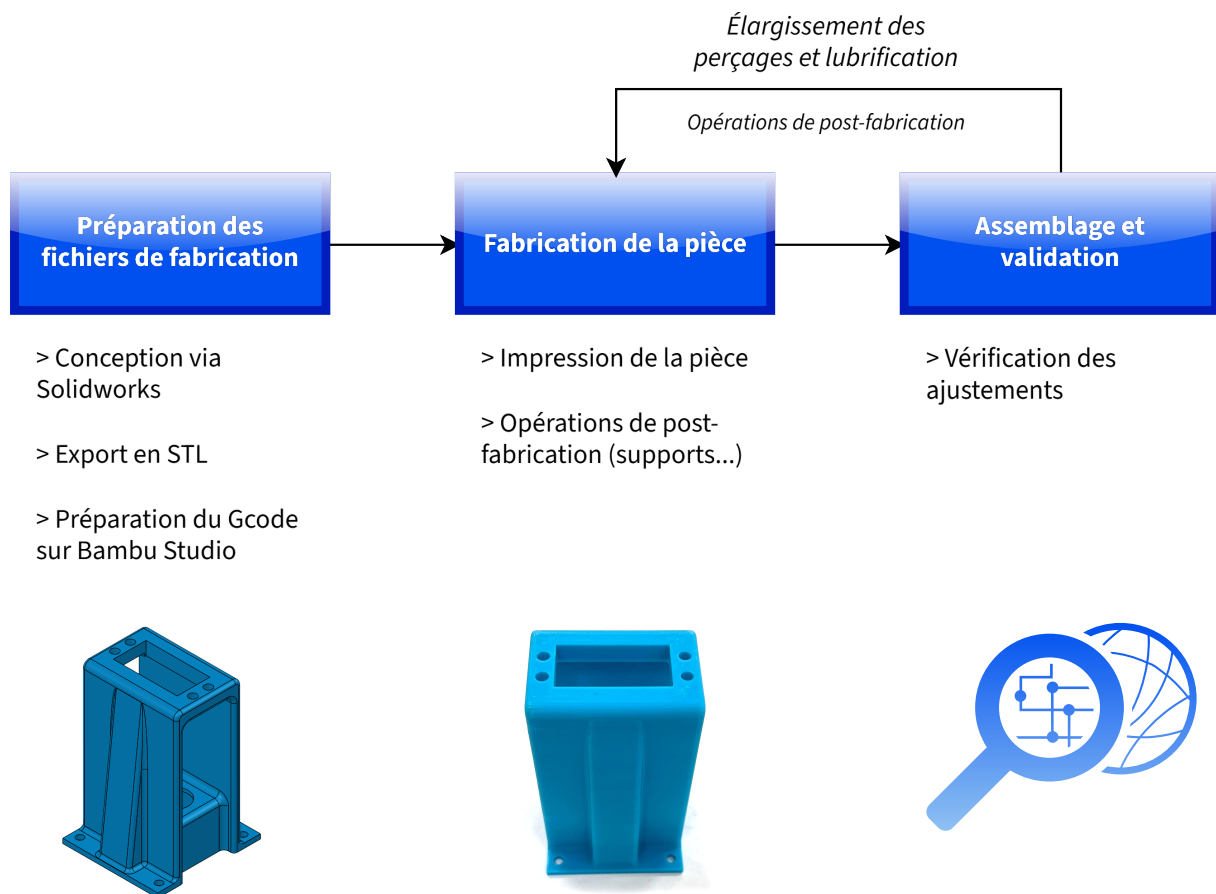


FIGURE 51 – Récapitulatif des étapes de fabrication du porte-servomoteur

4

AUTOMATIQUE

+++++++[>++++++>+++++++>+++++++>+++++++>++++
++++++>+++++++>+++++++>+++++++>+++++++>++++
++++++>+++++++<<<<<<<<-]>-----,>---.>---.>+.>-,>---.
>---,>---.>+,>---,>+.

4.1 Modélisation théorique du moteur CC

La modélisation du comportement du moteur à courant continu (CC) à partir des équations différentielles régissant son fonctionnement est nécessaire afin d'**obtenir une relation entre la vitesse de rotation du moteur $\omega(t)$ et la tension U appliquée à ses bornes, ainsi que l'influence d'un couple résistant C_r appliqué à l'axe du moteur.**

Nous avons modélisé le moteur (Fig. 52) électriquement par une inductance L en série avec une résistance R , alimentées par une tension continue U . La force contre-électromotrice (CEM) $e(t)$ est proportionnelle à la vitesse de rotation $\omega(t)$. Mécaniquement, le moteur est modélisé par un couple moteur C_m , qui subira l'action d'un couple résistant C_r et d'un couple de frottement visqueux $C_f(t) = f\omega(t)$.

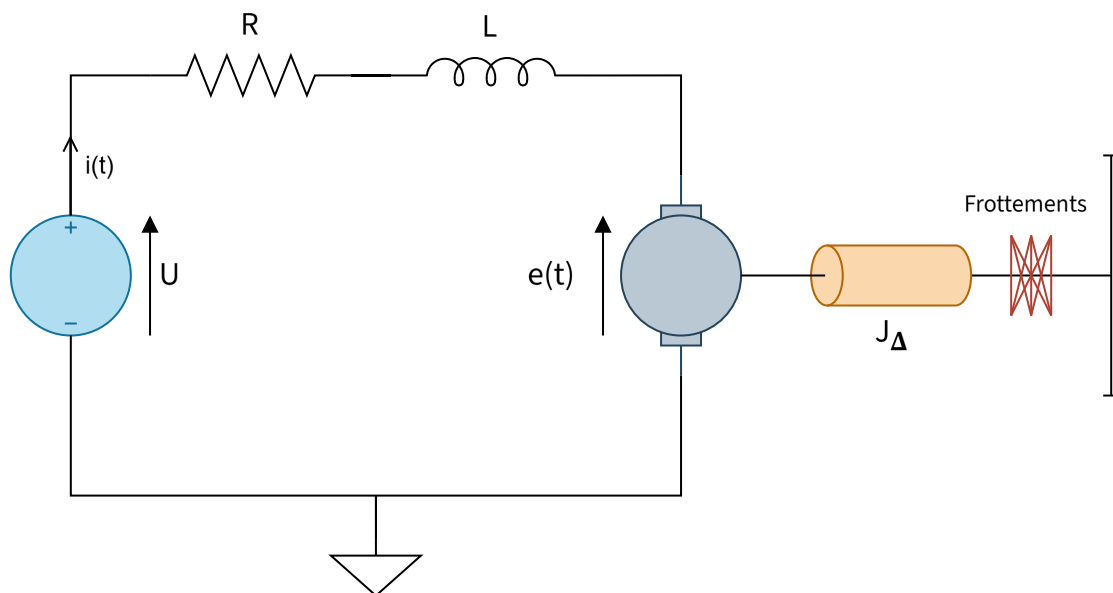


FIGURE 52 – Modélisation électrique et mécanique du moteur CC

Nous sommes donc partis des équations suivantes :

$$\begin{cases} U = e(t) + L \cdot \frac{di(t)}{dt} + R \cdot i(t) \\ J_{\Delta} \cdot \frac{d\omega(t)}{dt} = C_m(t) - C_r - C_f(t) \\ C_m(t) = K \cdot i(t) \\ e(t) = K \cdot \omega(t) \end{cases}$$

Puis, après avoir appliqué la transformée de Laplace et résolu le système d'équations, nous avons obtenu la fonction de transfert du moteur CC entre la vitesse de rotation $\Omega(p)$ et la tension d'alimentation $U(p)$ en présence d'un couple résistant $C_r(p)$:

$$\Omega(p) = \frac{R + L \cdot p}{(R + L \cdot p)(J_{\Delta} \cdot p + f) + K^2} \cdot \left(\frac{K \cdot U(p)}{R + L \cdot p} - C_r \right)$$

Nous avons ensuite comparé les constantes de temps mécanique et électrique du moteur, respectivement $\tau_{me} = \frac{J_{\Delta}}{f}$ et $\tau_{el} = \frac{L}{R}$. Dans la majorité écrasante des cas, la constante de

temps électrique est bien plus petite que la constante de temps mécanique ($\tau_{el} \ll \tau_{me}$). Nous avons donc fait l'approximation $\tau_{el} \approx 0$, ce qui revient à négliger l'inductance L du moteur dans notre modélisation. La fonction de transfert se simplifie donc au premier ordre.

Après simplification, nous avons établi l'expression de la vitesse de rotation du moteur en régime permanent avec couple résistant C_r appliqué à l'axe du moteur :

$$\omega(t) = (K_u \cdot U - K_r \cdot C_r) \cdot (1 - e^{-\frac{t}{\tau}})$$

avec $K_u = \frac{K}{R \cdot f + K^2}$, $K_r = \frac{K}{f + \frac{K^2}{R}}$, et $\tau = \frac{J_{\Delta} \cdot R}{f \cdot R + K^2}$ où K est la constante de couple et de CEM du moteur, et R sa résistance interne.

Pour adapter cette équation à notre utilisation dans Scilab/Xcos, nous devons aussi inclure un retard t_r à partir duquel le couple résistant est introduit. On obtient donc deux expressions de la vitesse, la première quand $t \in [0, t_r]$ où le moteur ne subit pas l'action de ce couple, et la deuxième quand $t \in [t_r, +\infty]$.

$$\begin{cases} \omega(t) = (K_u \cdot U) \left(1 - e^{-\frac{t}{\tau}}\right) & \text{avec } t \in [0, t_r] \\ \omega(t) = (K_u \cdot U) \left(1 - e^{-\frac{t}{\tau}}\right) - (K_r \cdot C_r) \left(1 - e^{-\frac{(t-t_r)}{\tau}}\right) & \text{avec } t \in [t_r, +\infty] \end{cases}$$

4.2 Identification des paramètres du moteur CC

Nous souhaitons maintenant déterminer expérimentalement les paramètres K_u , K_r et τ afin de les utiliser dans l'implémentation de l'asservissement en vitesse et en position du moteur.



Lors de cette identification du moteur, le couple résistant C_r est appliqué instantanément, nous utilisons donc une expression simplifiée de la vitesse, $\omega(t) = (K_u \cdot U - K_r \cdot C_r) \left(1 - e^{-\frac{t}{\tau}}\right)$.

4.2.1 Protocole expérimental

Nous avons séparé la détermination des paramètres en deux expériences distinctes :

Mesure de K_u (sans charge) L'expression de la vitesse de rotation sans couple résistant est donnée par $\omega(t) = K_u \cdot U \cdot (1 - e^{-\frac{t}{\tau}})$. En régime permanent, celle-ci devient $\omega_{prm} = K_u \cdot U$. Cette relation suppose une linéarité parfaite entre ω_{prm} et U . En réalité, le moteur présentera des non-linéarités, il est donc nécessaire de faire varier la tension U lors des mesures de ω_{prm} puis d'effectuer une **régression linéaire** pour obtenir K_u . Nous pouvons en déduire les paramètres K_u et τ en réalisant l'expérience suivante :

1. Fixer le moteur sans charge (couple résistant $C_r = 0$).
2. Appliquer une tension d'alimentation U connue au moteur.

3. Mesurer la vitesse de rotation $\omega(t)$ du moteur grâce au codeur incrémental à intervalles de temps réguliers.
4. Répéter les étapes 2 et 3 pour différentes valeurs de la tension U (dans notre cas, 12 valeurs différentes).

En utilisant les valeurs de régime permanent obtenues pour chaque tension, nous pouvons tracer le graphe de ω_{prm} en fonction de U et effectuer une régression linéaire pour déterminer la pente, qui correspond à K_u .

Mesure de la constante temporelle τ La constante de temps τ peut également être déterminée à partir des données obtenues lors de l'expérience sans charge. En revanche, comme la détermination de celle-ci est sensible au bruit, nous devons l'isoler en utilisant une seule valeur de tension U et en analysant le démarrage du moteur N fois afin de réduire l'influence des erreurs aléatoires. Nous procédons donc comme suit :

1. Fixer le moteur sans charge (couple résistant $C_r = 0$).
2. Appliquer une tension d'alimentation U connue au moteur.
3. Mesurer la vitesse de rotation $\omega(t)$ du moteur grâce au codeur incrémental.
4. Répéter les étapes 2 et 3 un grand nombre de fois (dans notre cas, $N = 10$ fois).

La détermination finale de la valeur de τ est effectuée en ajustant la courbe des vitesses obtenues à la fonction exponentielle théorique $\omega(t) = K_u \cdot U \cdot (1 - e^{-\frac{t}{\tau}})$, en utilisant une méthode de régression non linéaire. En utilisant le coefficient de corrélation R^2 , nous pouvons obtenir l'incertitude associée à la valeur de τ .

Mesure de K_r (avec charge) L'expression de la vitesse en régime permanent s'écrit : $\omega_{\text{perm}} = \omega_0 - K_r \cdot C_r$, où $\omega_0 = K_u \cdot U$ est la vitesse à vide pour une tension donnée. Pour déterminer K_r avec précision, nous fixons la tension $U = 12 \text{ V}$ et nous faisons varier le couple résistant C_r . Nous procédons donc comme suit :

1. Fixer une tension d'alimentation constante.
2. Suspendre une unique masse m à l'axe du moteur (via une poulie de rayon r) pour générer un couple résistant unique $C_r = m \cdot g \cdot r$.
3. Comparer les vitesses de rotation en régime permanent ω_{prm} obtenues à vide et avec la masse suspendue.

Nous utilisons ensuite l'expression théorique de K_r , qui est $K_r = \frac{2500 \times (\omega_0 - \omega_{\text{charge}})}{C}$, où C est le couple résistant appliqué (en N.m) et ω_{charge} la vitesse de rotation mesurée avec la charge.

Paramètre	Grandeur à mesurer	Paramètres fixes	Paramètres variables	Méthode de calcul
K_u	Vitesse de rotation en régime permanent ω_{prm}	Sans charge ($C_r = 0$)	Tension d'alimentation U	Régression linéaire de ω_{prm} en fonction de U
τ	Vitesse de rotation $\omega(t)$ lors du démarrage	Sans charge ($C_r = 0$), tension fixe U	Aucun.	Régression non linéaire de $\omega(t)$ sur la fonction exponentielle théorique
K_r	Vitesse de rotation en régime permanent ω_{prm}	Tension fixe U et couple résistant C_r	Aucun	Calcul direct

TABLEAU 8 – Récapitulatif des expériences pour l'identification du moteur CC.

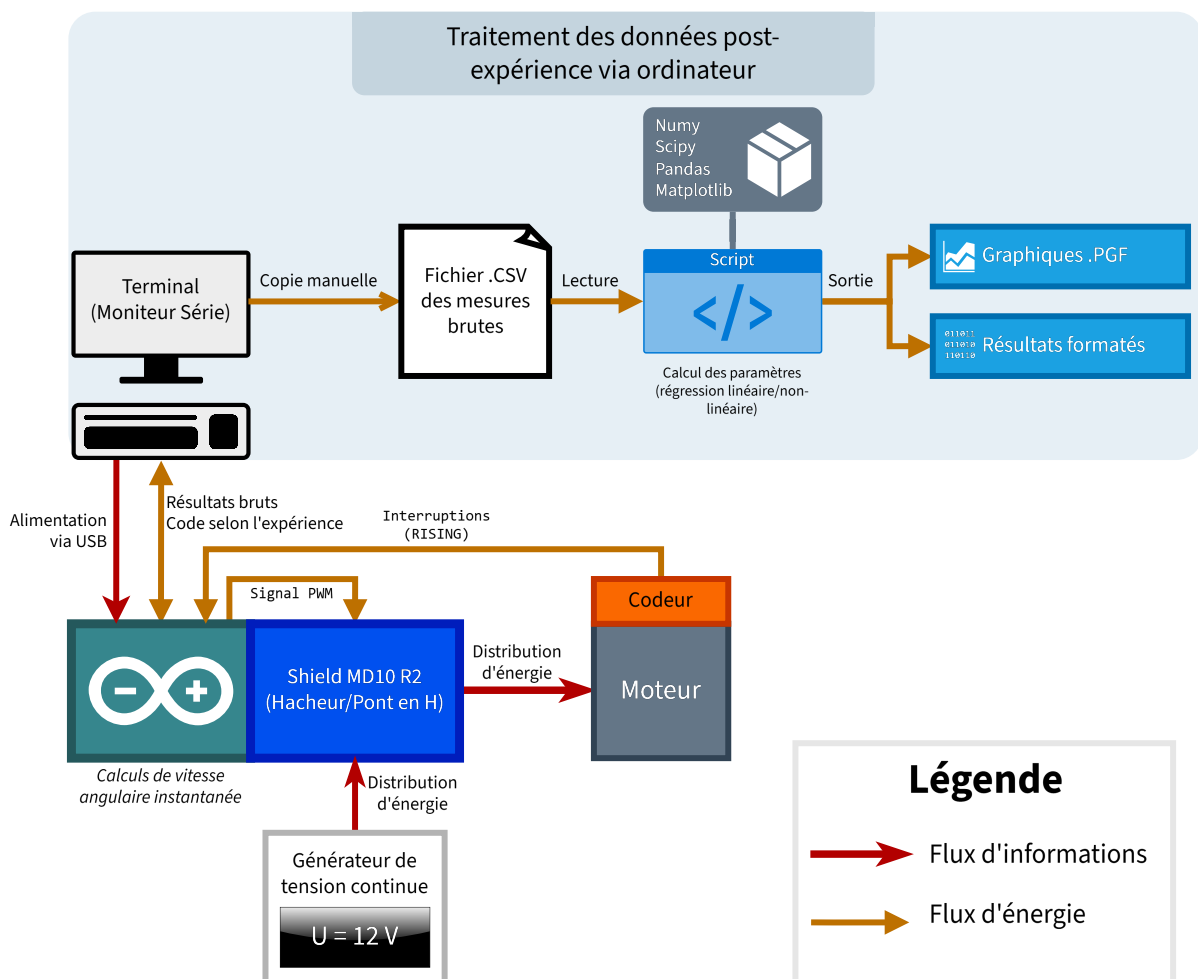


FIGURE 53 – Schéma global du protocole expérimental pour la détermination des paramètres du moteur CC

Régression linéaire

Pour déterminer K_u , nous n'avons pas simplement calculé la moyenne des rapports $\frac{\omega}{U}$, car cette méthode est sensible aux erreurs de mesure à basse tension et ne prend pas en compte le seuil de démarrage du moteur (causé par des frottements internes...). Nous avons plutôt procédé à une régression linéaire au sens des moindres carrés sur l'ensemble des points de mesure (ω_{prm}, U) . L'équation modèle utilisée est de la forme affine $\omega = a \cdot U + b$, où la pente a correspond à notre coefficient K_u .

Fréquence d'échantillonnage

Pour garantir une mesure précise de la constante de temps τ , il est impératif que la dynamique d'acquisition soit nettement plus rapide que la dynamique du système. Nous devons donc choisir une période d'échantillonnage T_e suffisamment petite par rapport à τ . Nous avons donc procédé à un pré-dimensionnement de cette dernière.

Ne connaissant pas τ a priori, nous avons estimé sa valeur théorique, qui se trouve généralement dans l'ordre de quelques dizaines de millisecondes pour un petit moteur CC. Nous avons donc fixé la période d'échantillonnage à $T_e = 5 \text{ ms}$, ce qui nous permet d'obtenir une résolution temporelle adéquate pour capturer la dynamique du moteur.

Variation de la charge

Idéalement, pour déterminer K_r , il serait préférable de faire varier la charge appliquée au moteur à l'aide de plusieurs masses suspendues différentes. Cependant, en raison de contraintes matérielles, nous n'avons pas pu accéder à un ensemble suffisant de masses pour effectuer cette variation. Par conséquent, nous avons choisi de suspendre une seule masse connue à l'axe du moteur pour générer un couple résistant unique. Bien que cette approche limite la fiabilité de la détermination de K_r , elle nous permet néanmoins d'obtenir une estimation précise de ce paramètre.

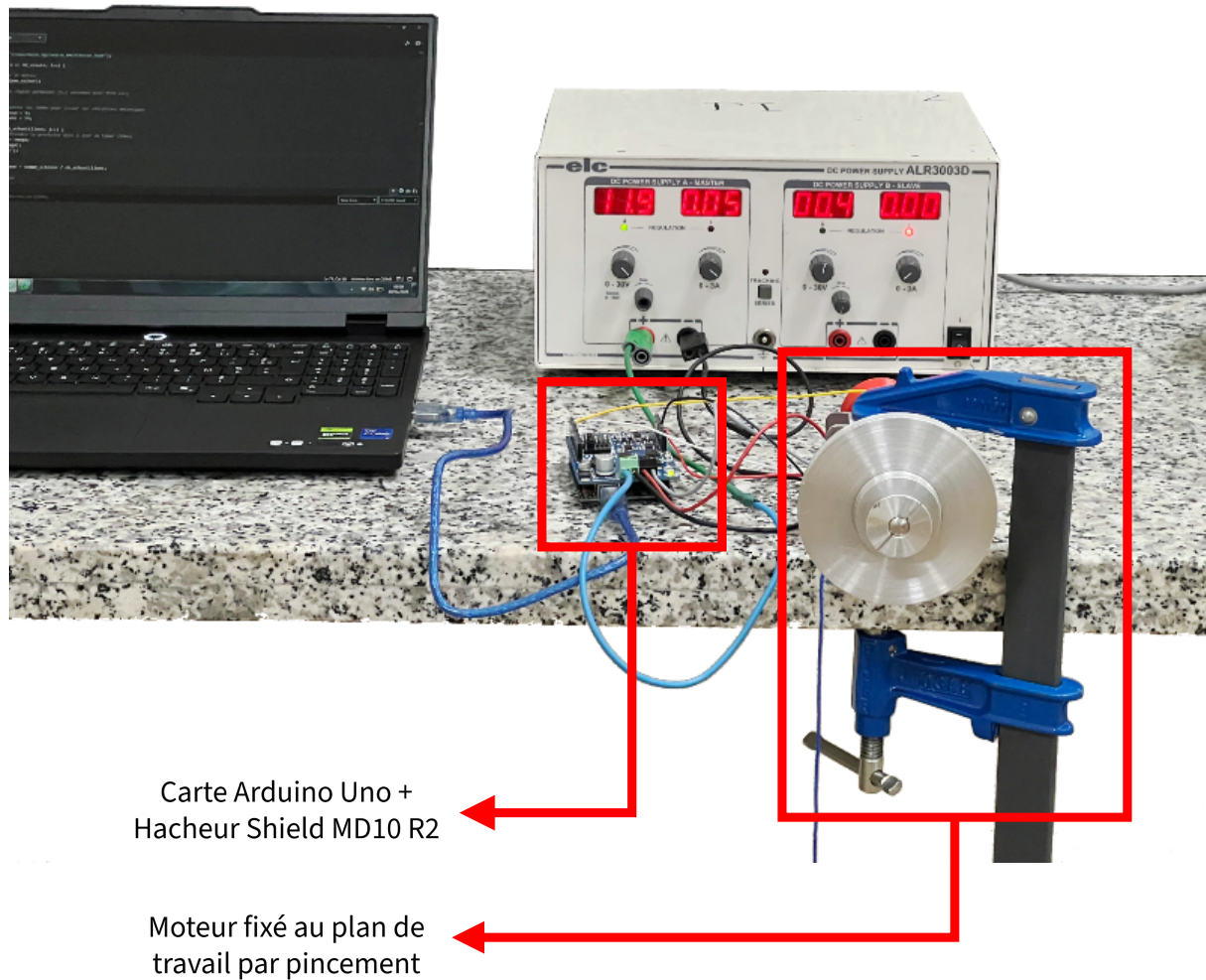


FIGURE 54 – Montage expérimental pour la détermination des paramètres du moteur CC (expériences à vide)

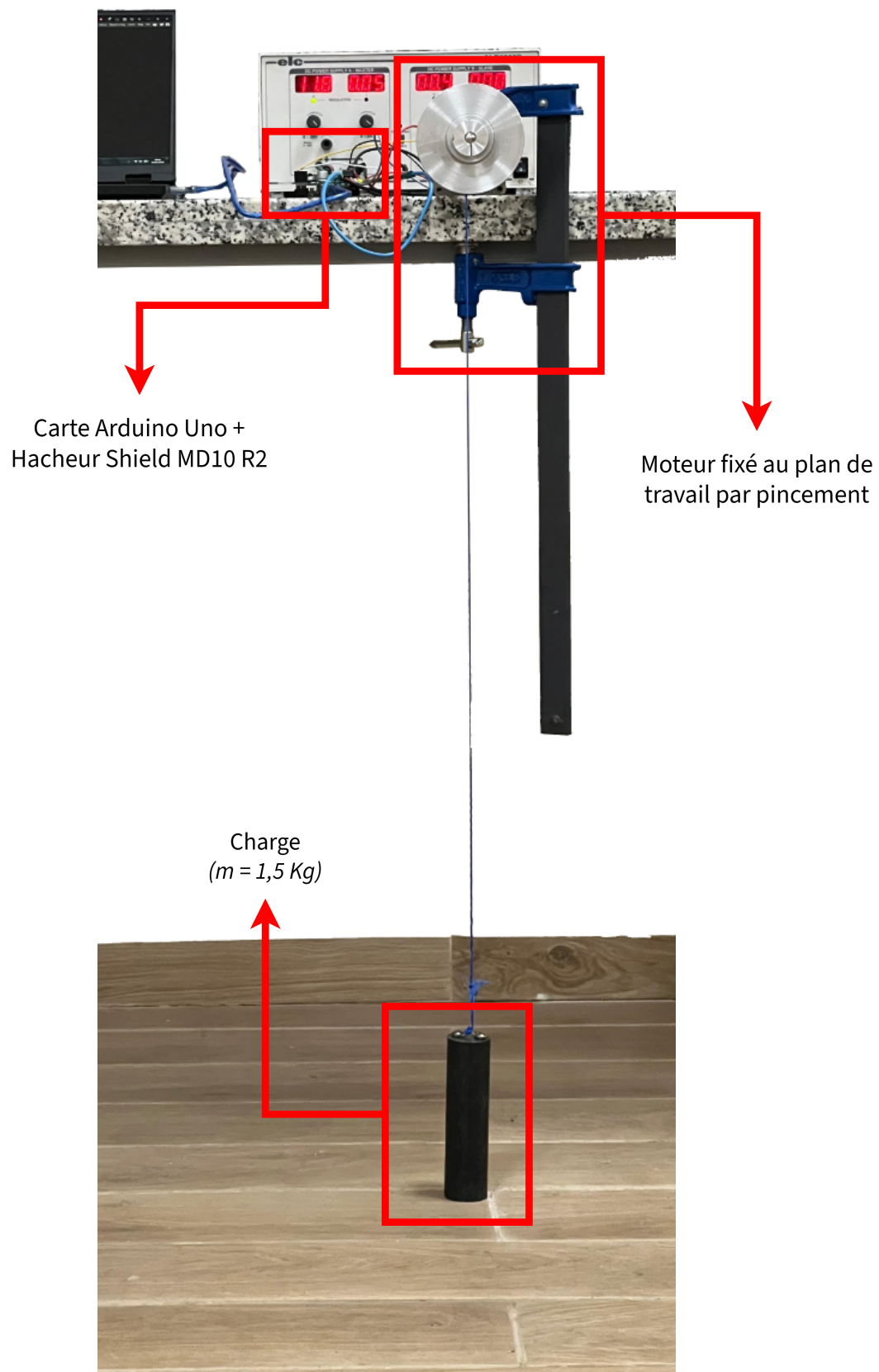


FIGURE 55 – Montage expérimental pour la détermination des paramètres du moteur CC (expérience avec charge)

4.2.2 Traitement des résultats

Les résultats bruts des expériences sont enregistrés dans des fichiers CSV. Nous utilisons un script Python (disponible dans la section Annexe) pour analyser ces données, générer les courbes expérimentales afin d'extraire les paramètres K_u , K_r et τ .

Nous avons représenté les données expérimentales et les ajustements effectués dans les figures suivantes :

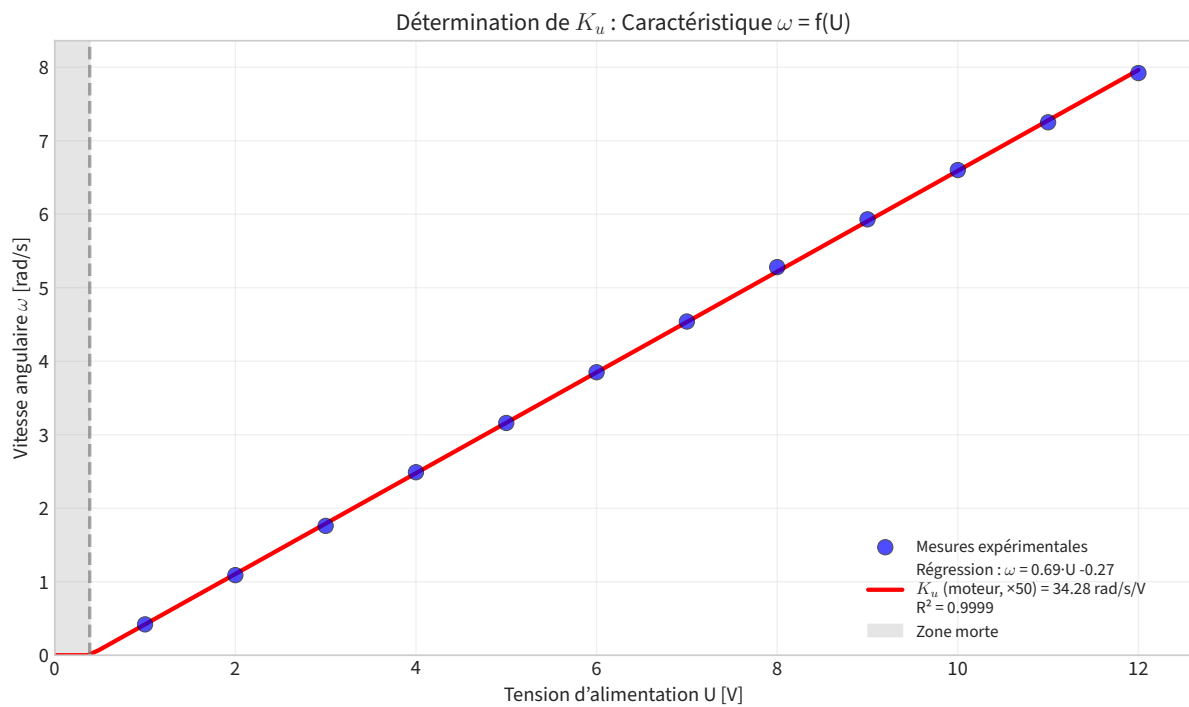


FIGURE 56 – Graphique de la vitesse de rotation en régime permanent $\omega_{pr\dot{m}}$ sans charge pour plusieurs valeurs de tension.

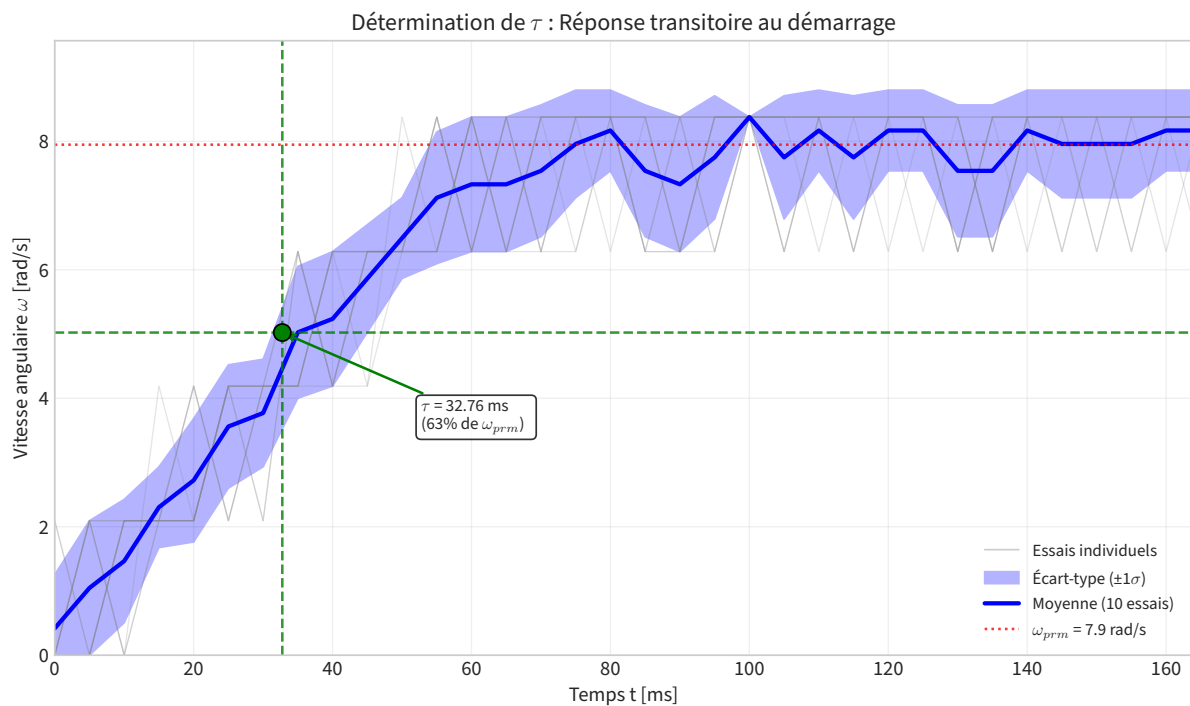


FIGURE 57 – Graphique des vitesses de rotation sans charge avec tension fixe (mesure de τ).

Courbes bruitées

Les courbes de vitesse obtenues (indiquées en gris sur le graphique) sont bruitées ; elles présentent des fluctuations sous forme de zigzags, qui sont dues au bruit de quantification du codeur incrémental utilisé.

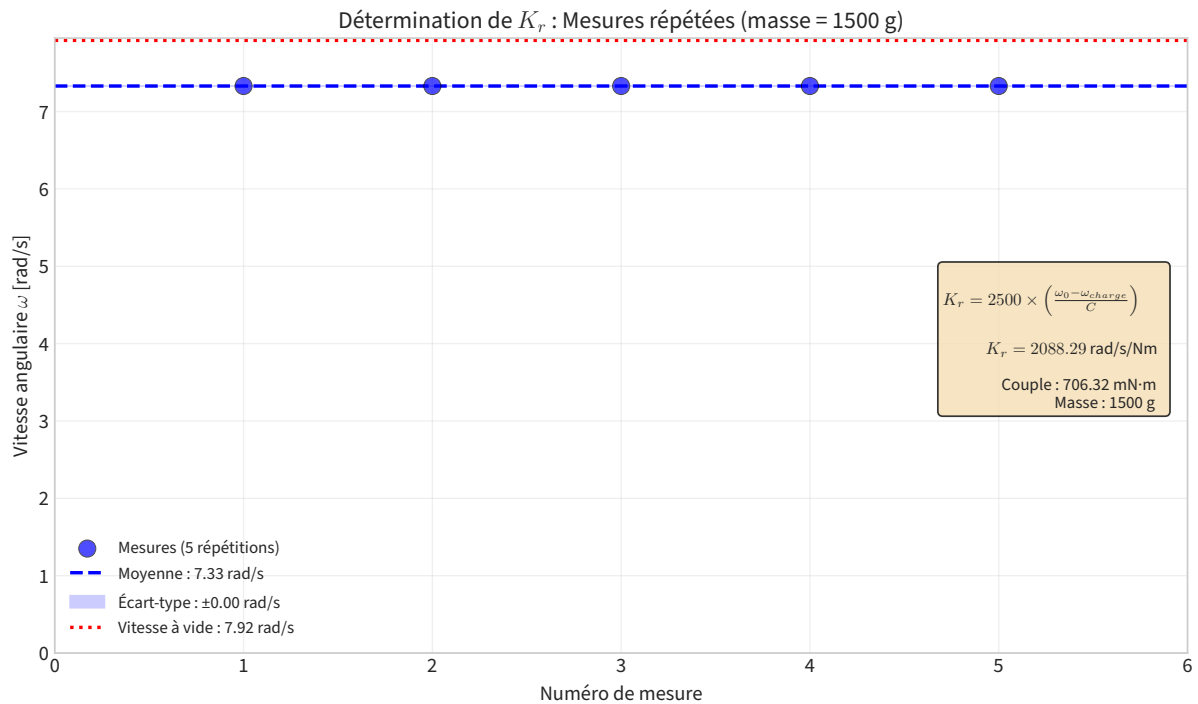


FIGURE 58 – Graphique de la vitesse de rotation en régime permanent avec charge.

Après analyse des données, nous avons obtenu les valeurs suivantes pour les paramètres du moteur CC :

Paramètre	Valeur	R ²	Incertitude
Constante K_u	34,28 rad/(V.s)	0,9999	Négligeable
Constante K_r	2088,29 rad/(N.m.s)	0,9986	Négligeable
Constante de temps τ	32,76 ms	0,9020	$\pm 3,82$ ms

TABLEAU 9 – Paramètres déterminés expérimentalement lors de l'identification du moteur CC.

Ces valeurs seront utilisées dans la modélisation de l'asservissement en vitesse et en position du moteur.

4.3 Modélisation de l'asservissement

4.3.1 Asservissement en vitesse

La modélisation de l'asservissement en vitesse du moteur CC est réalisée à l'aide d'un correcteur proportionnel-intégral (PI). Le correcteur PI ajuste la tension d'alimentation du moteur en fonction de l'erreur entre la vitesse de consigne et la vitesse mesurée.

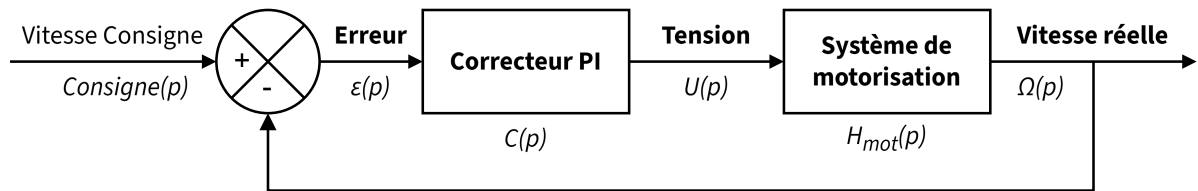


FIGURE 59 – Schéma simplifié du correcteur PI utilisé pour l'asservissement en vitesse

Justification mathématique du correcteur PI Nous cherchons à annuler l'erreur statique ε_s (écart entre la consigne et la mesure en régime permanent) pour une consigne de vitesse constante (entrée échelon d'amplitude V_0). Modélisons tout d'abord le système présenté dans la figure 59. La fonction de transfert du moteur (entrée Tension U , sortie Vitesse Ω) est modélisée par un premier ordre :

$$H_{mot}(p) = \frac{\Omega(p)}{U(p)} = \frac{K_u}{1 + \tau p}$$

On sait que l'erreur est la consigne $Consigne(p)$ moins la sortie du système $\Omega(p)$, soit :

$$\varepsilon(p) = Consigne(p) - \Omega(p)$$

Dans le cas d'un asservissement en boucle fermée avec un correcteur $C(p)$, on a :

$$\Omega(p) = C(p) \cdot H_{mot}(p) \cdot \varepsilon(p)$$

D'où l'expression de l'erreur en boucle fermée :

$$\varepsilon(p) = \frac{Consigne(p)}{1 + C(p) \cdot H_{mot}(p)} = \frac{Consigne(p)}{1 + C(p) \cdot \frac{K_u}{1 + \tau p}}$$

où $C(p)$ est la fonction de transfert du correcteur et $Consigne(p) = \frac{V_0}{p}$ (transformée de Laplace d'un échelon).

Testons maintenant l'effet d'un correcteur Proportionnel pur ($C(p) = K_p$). Calculons l'erreur finale à l'aide du *Théorème de la Valeur Finale* :

$$\begin{aligned} \varepsilon_s &= \lim_{p \rightarrow 0} p \cdot \varepsilon(p) = \lim_{p \rightarrow 0} p \cdot \frac{V_0/p}{1 + K_p \cdot \frac{K_u}{1 + \tau p}} \\ \varepsilon_s &= \lim_{p \rightarrow 0} \frac{V_0}{1 + \frac{K_p K_u}{1 + 0}} = \frac{V_0}{1 + K_p K_u} \neq 0 \end{aligned}$$

On constate donc qu'avec un correcteur P, l'erreur statique n'est jamais nulle. La vitesse réelle sera toujours inférieure à la consigne.

Essayons maintenant avec un correcteur Proportionnel-Intégral ($C(p) = K_p + \frac{K_i}{p} = \frac{K_p p + K_i}{p}$). L'expression de la limite devient :

$$\varepsilon_s = \lim_{p \rightarrow 0} p \cdot \frac{V_0/p}{1 + \frac{K_p p + K_i}{p} \cdot \frac{K_u}{1 + \tau p}} = \lim_{p \rightarrow 0} \frac{V_0}{1 + \frac{K_i K_u}{p(1 + \tau p)}}$$

On multiplie numérateur et dénominateur par p pour lever l'indétermination, ou plus simplement, on observe que le terme $\frac{K_i K_u}{p}$ au dénominateur tend vers l'infini quand $p \rightarrow 0$.

$$\varepsilon_s = \frac{V_0}{1 + \infty} = 0$$

L'ajout de l'action intégrale ($\frac{1}{p}$) dans le correcteur permet mathématiquement d'annuler l'erreur statique. Le correcteur PI est donc indispensable pour cet asservissement.

Le schéma bloc de l'asservissement en vitesse réalisé via Scilab/Xcos est présenté dans la Figure 60.

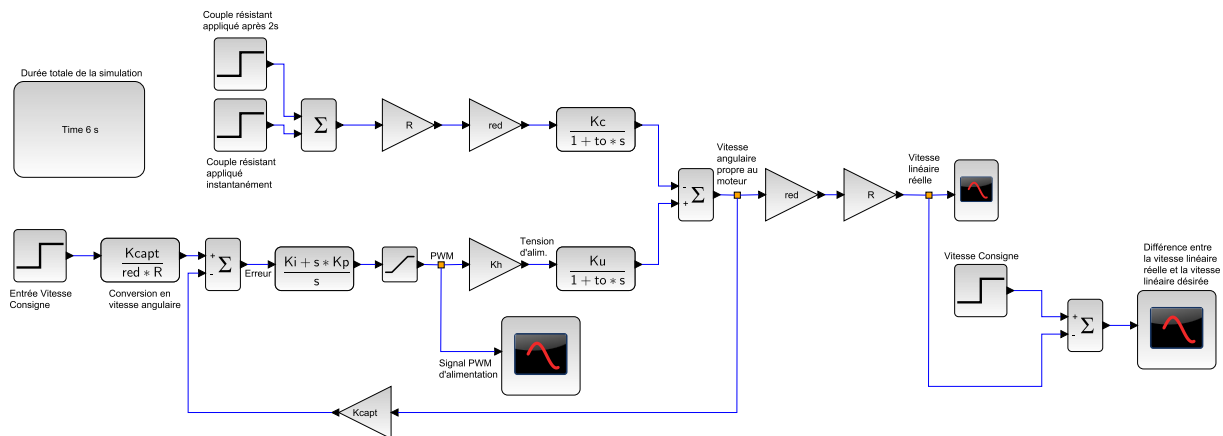


FIGURE 60 – Schéma bloc de l'asservissement en vitesse du moteur CC

Dans le schéma bloc, les variables notées s correspondent à la variable de la transformée de Laplace, notée p dans la littérature francophone. Cette notation est utilisée dans le logiciel de simulation Scilab/Xcos.

Description du schéma bloc L'asservissement en vitesse commence par la définition d'une vitesse de consigne, qui est ensuite convertie en vitesse angulaire (rad/s) en utilisant la constante de conversion du codeur incrémental ($K_{capt} = 7,64 \text{ rad/s/impulsion}$), qui est égale à $\frac{4 \times \text{rés. du codeur}}{2\pi}$, tout en sachant que la résolution est $X = 12 \text{ impulsions/tour}$, et que la capteur possède deux cercles dont on capte les changements CHANGE (d'où la multiplication par 4). Le facteur de réduction du moteur et le rayon de la roue sont également pris en compte dans cette conversion.

On compare ensuite la vitesse consigne avec la vitesse mesurée pour calculer l'erreur. Cette erreur est ensuite traitée par le contrôleur PI, qui calcule la tension de commande à appliquer au moteur. Un saturateur est utilisé pour limiter cette tension entre 0 et 12 V (correspondant à un signal PWM entre 0 et 255). Dans la simulation, nous avons utilisé des signaux PWM situés entre -255 et 255 pour permettre un contrôle bidirectionnel du moteur, ce qui sera en réalité assuré par le pin Dir du hacheur.

Dans une autre partie du schéma bloc, deux couples résistants sont appliqués au moteur : un instantanément et un autre après un délai de 2 secondes. Afin d'analyser mathématiquement ce comportement, nous devons légèrement modifier les équations d'état utilisées dans

la modélisation théorique du moteur CC précédemment. Les nouvelles expressions de la vitesse en fonction du temps deviennent :

$$\begin{cases} \omega(t) = (K_u \cdot U - K_r C_{r1}) \left(1 - e^{-\frac{t}{\tau}}\right) & \text{avec } t \in [0, t_{r,\text{add}}] \\ \omega(t) = (K_u \cdot U - K_r \cdot C_{r1}) \left(1 - e^{-\frac{t}{\tau}}\right) - (K_r \cdot C_{r2}) \left(1 - e^{-\frac{(t-t_{r,\text{add}})}{\tau}}\right) & \text{avec } t \in [t_{r,\text{add}}, +\infty] \end{cases}$$

où C_{r1} est le couple résistant appliqué instantanément, C_{r2} est le couple résistant appliqué après le délai $t_{r,\text{add}}$ (2 secondes dans notre cas).

Après calcul de la vitesse angulaire du moteur, nous appliquons le facteur de réduction et le rayon de la roue pour obtenir la vitesse linéaire réelle de la plateforme mobile.

Choix des constantes du contrôleur PI Le choix des constantes de gain proportionnel K_p et intégral K_i du contrôleur PI a été effectué par tâtonnement (méthode empirique). Nous avons commencé avec des valeurs initiales faibles pour K_p et K_i , puis nous les avons ajustées progressivement en observant la réponse du système aux variations de la consigne de vitesse. Nous avons cherché à minimiser le temps de réponse tout en évitant les dépassements excessifs et les oscillations. Après plusieurs itérations, nous avons déterminé les valeurs optimales de K_p et K_i qui offrent une performance satisfaisante de l'asservissement en vitesse.

Le tableau 10 récapitule les constantes du contrôleur PI ainsi que les constantes utilisées dans la simulation de l'asservissement en vitesse.

Constante	Valeur
Gain proportionnel K_p	1,0
Gain intégral K_i	0,5
Constante de conversion du codeur incrémental K_{capt}	7,64 imp/tour
Rayon de la roue R	0,05 m
Constante K_u	34,28 rad/(V.s)
Constante de temps τ	32,76 ms
Constante K_r	2088,29 rad/(N.m.s)
Constante de conversion du hacheur K_h	0,0471 V

TABLEAU 10 – Constantes du contrôleur PI et constantes utilisées dans la simulation de l'asservissement en vitesse

La courbe de la figure 61 présente les résultats de la simulation de l'asservissement en vitesse du moteur CC. La première courbe montre l'évolution de la vitesse réelle de la plateforme mobile en m/s, la deuxième courbe affiche le signal PWM généré par le contrôleur PI, et la troisième courbe illustre l'erreur entre la vitesse consigne et la vitesse réelle. On peut

observer l'influence du couple résistant appliqué après 2 secondes sur la vitesse de la plateforme, et comment le contrôleur PI ajuste le signal PWM pour compenser cette perturbation et maintenir la vitesse consigne.

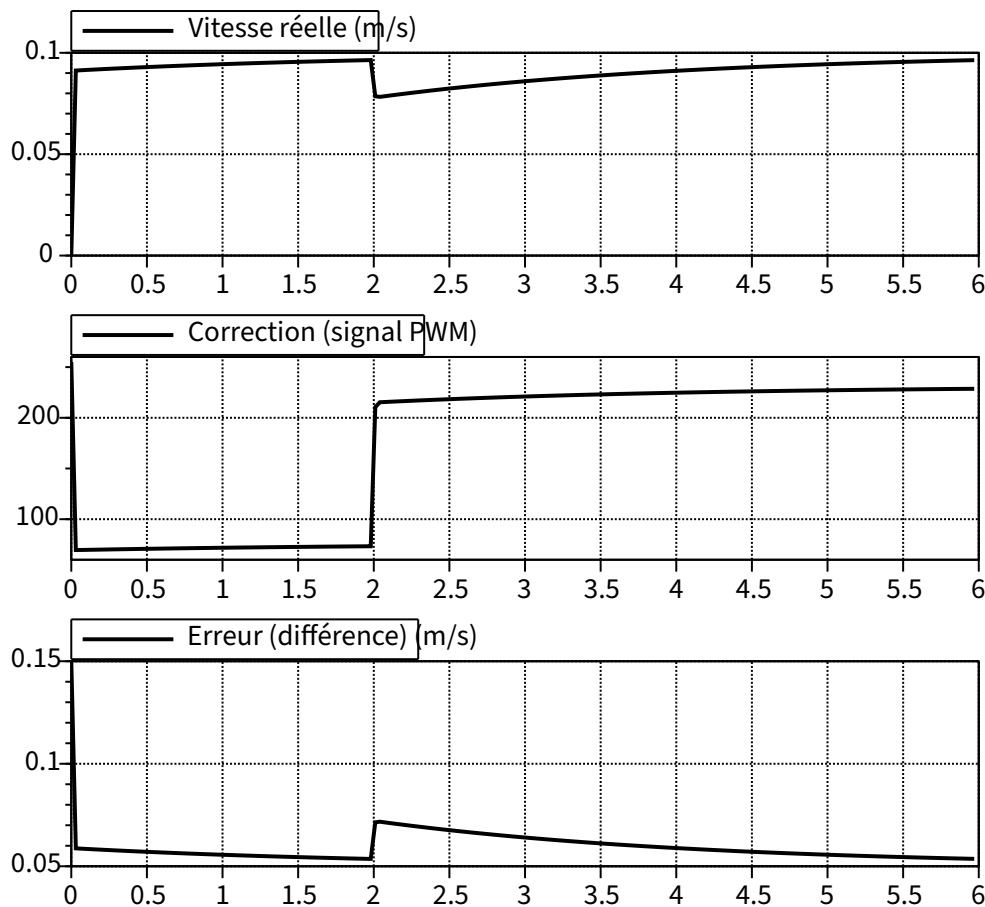


FIGURE 61 – Résultats de la simulation de l'asservissement en vitesse du moteur CC (vitesse consigne de 0,10 m/s)

4.3.2 Asservissement en position

L'asservissement en position du moteur CC est implémenté à l'aide d'un correcteur proportionnel uniquement (P).

Justification mathématique du correcteur P Pour l'asservissement en position, nous cherchons à atteindre une position cible θ_0 sans erreur (Consigne Pos(p) = $\frac{\theta_0}{p}$). Le système physique est composé du moteur suivi de la relation cinématique entre vitesse et position ($\theta = \int \omega dt$, soit $\frac{1}{p}$ dans le domaine de Laplace). La fonction de transfert globale à piloter est donc :

$$H_{syst}(p) = H_{mot}(p) \times \frac{1}{p} = \frac{K_u}{p(1 + \tau p)}$$

On remarque la présence naturelle d'un terme p au dénominateur (intégrateur pur).

Testons d'abord l'influence d'un simple correcteur Proportionnel ($C(p) = K_p$). L'erreur en boucle fermée s'écrit :

$$\varepsilon(p) = \frac{\text{Pos}(p)}{1 + C(p) \cdot H_{\text{syst}}(p)} = \frac{\theta_0/p}{1 + K_p \cdot \frac{K_u}{p(1+\tau p)}}$$

Appliquons le *Théorème de la Valeur Finale* :

$$\varepsilon_s = \lim_{p \rightarrow 0} p \cdot \varepsilon(p) = \lim_{p \rightarrow 0} p \cdot \frac{\theta_0/p}{1 + \frac{K_p K_u}{p(1+\tau p)}}$$

$$\varepsilon_s = \lim_{p \rightarrow 0} \frac{\theta_0}{1 + \frac{K_p K_u}{p(1+0)}}$$

Lorsque $p \rightarrow 0$, le terme $\frac{K_p K_u}{p}$ tend vers l'infini.

$$\varepsilon_s = \frac{\theta_0}{1 + \infty} = 0$$

On peut donc en déduire que grâce à l'intégration naturelle de la vitesse vers la position, un simple correcteur Proportionnel (P) suffit théoriquement pour garantir une erreur nulle en position.

Frottements secs et terme intégral additionnel très faible

Un terme intégral faible peut être ajouté en pratique uniquement pour vaincre les frottements secs (non-linéarités non prises en compte dans ce calcul linéaire). Près de la position consigne, la commande P du correcteur devient si faible qu'elle ne peut plus vaincre les frottements secs, et la plateforme s'arrête.

Un terme Intégral additionnel servira à détecter ce blocage en accumulant cette petite erreur résiduelle au fil du temps jusqu'à générer une commande suffisante pour "décoller" le moteur et finir le mouvement. On le choisira faible pour que cette accumulation soit douce et ne crée pas d'instabilité.

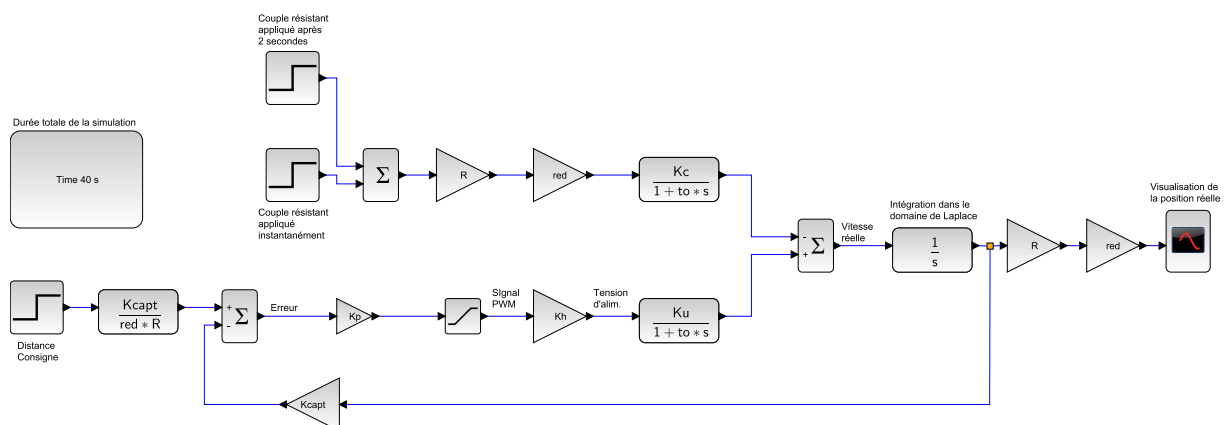


FIGURE 62 – Schéma bloc de l'asservissement en position du moteur CC

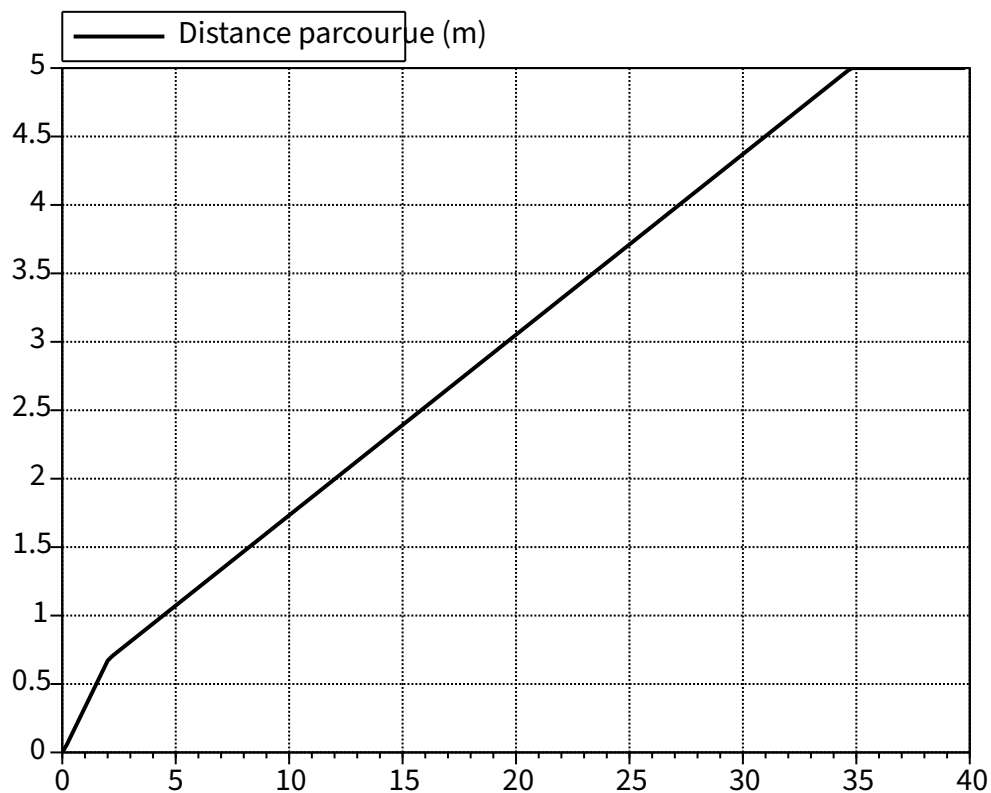


FIGURE 63 – Résultats de la simulation de l'asservissement en position du moteur CC (position consigne de 1 m)

Description du schéma bloc La courbe de la figure 63 présente les résultats de la simulation de l'asservissement en position du moteur CC. On peut observer l'évolution de la position réelle de la plateforme mobile en mètres, ainsi que l'influence du couple résistant appliqué après 2 secondes.

5

ÉLECTRIQUE ET ÉLECTRONIQUE

[illegible]

5.1 Architecture électrique

5.1.1 Aperçu général

La plateforme mobile est équipée d'un système électrique et électronique conçu pour assurer son fonctionnement autonome et sa communication avec l'utilisateur. Le cœur du système est une carte Arduino Uno, qui gère l'asservissement en vitesse et en position du moteur CC, contrôle le servomoteur de direction, lit les données du codeur incrémental, et communique avec l'utilisateur via un module Bluetooth HC-06. L'alimentation de la plateforme est assurée par un ensemble de batteries Lithium-Ion fournissant une tension de 12V.

5.1.2 Composants électroniques utilisés

Le tableau 11 présente un récapitulatif des principaux éléments utilisés, ainsi que leur rôle spécifique dans le système.

Composant	Rôle
Carte Arduino Uno	Asservissement en vitesse et position, contrôle du servomoteur, lecture des données du codeur, communication avec l'utilisateur via Bluetooth
Hacheur/Pont en H Shield MD10 R2	Distribution de l'énergie des batteries vers le moteur CC
Batteries Lithium-Ion 12V (3x3.7V nominale en série)	Alimentation de l'ensemble du système
Régulateur de tension	Alimentation du servomoteur à 7V
Codeur incrémental	Mesure de la vitesse et de la position du moteur CC
Servomoteur Futaba S3010	Direction de la plateforme mobile
Module Bluetooth HC-06	Communication avec l'utilisateur
Diviseur de tension (résistances de 200 Ω et 100 Ω)	Protection du module Bluetooth contre les surtensions du port TX de la carte Arduino (réduction de 5V à 3.3V)
Adaptateurs divers (adaptateurs DC Jack, etc.)	Alimentation et connexion des différents composants
Motoréducteur SGP67S	Transmission de la puissance du moteur CC à la roue avant

TABLEAU 11 – Récapitulatif des composants électriques et électroniques utilisés dans la plateforme mobile.

5.1.3 Câblage et choix techniques

Interrupteur principal Un interrupteur principal est intégré au système pour permettre la coupure de l'alimentation générale de la plateforme mobile. Cela facilite la mise en marche et l'arrêt du système, tout en assurant une sécurité supplémentaire lors des opérations de maintenance ou de transport.

Régulateur de tension pour le servomoteur Le servomoteur Futaba S3010 nécessite une alimentation en 7V pour fonctionner de manière optimale. Pour répondre à cette exigence, un régulateur de tension est utilisé pour abaisser la tension fournie par les batteries (12V) à la valeur requise de 7V. Cela garantit que le servomoteur reçoit une alimentation stable et appropriée, évitant ainsi tout risque de surchauffe ou de dysfonctionnement.

Dimensionnement du câblage de puissance La distribution de l'énergie (Batterie → Hacheur → Moteur) implique des courants pouvant atteindre plusieurs ampères lors des pics de couple. Afin de limiter les chutes de tension et l'échauffement par effet Joule, nous avons proscrit les fils de prototypage standards (type Jumper), dont la section est insuffisante. Les liaisons de puissance ont donc été réalisées avec du **câble multibrin** (adaptée aux courants nominaux du moteur), garantissant ainsi la sécurité électrique et la performance motrice.



FIGURE 64 – Câbles multibrin pour l'alimentation de puissance

Utilisation de connecteurs électriques Pour faciliter les connexions électriques entre les différents composants de la plateforme mobile, des connecteurs électriques tels que des serres-fils/borniers à vis sont utilisés. Ces connecteurs permettent de relier rapidement et solidement plusieurs fils ensemble sans nécessiter de soudure. Cette approche offre une flexibilité accrue pour le montage et la maintenance du système, tout en assurant des connexions fiables et sécurisées. C'est un choix de prototypage pratique qui permet également de modifier facilement le câblage si nécessaire, mais une version industrielle doit privilégier des connexions soudées pour une meilleure durabilité.



FIGURE 65 – Borniers à vis pour connexions électriques

Diviseur de tension pour le module Bluetooth Le module Bluetooth HC-06 utilisé pour la communication avec l'utilisateur est conçu pour fonctionner avec une tension maximale de 3.3V sur son port RX. Cependant, le port TX de la carte Arduino Uno fournit une tension de 5V, ce qui pourrait endommager le module Bluetooth. Pour protéger ce dernier, un diviseur de tension composé de deux résistances (200Ω et 100Ω) est mis en place. Ce diviseur réduit la tension de sortie du port TX de l'Arduino à une valeur sûre de 3.3V (rapport $2/3 \approx 0.66$), assurant ainsi la sécurité et la fiabilité des communications sans fil. La résistance de 100Ω est connectée entre le port TX de l'Arduino et le port RX du module Bluetooth, tandis que

la résistance de 200Ω est connectée entre le port RX du module Bluetooth et la masse. On obtient donc le rapport souhaité entre V_{out} et V_{in} , qui est de $\frac{R_2}{R_1+R_2} = \frac{200}{100+200} = \frac{2}{3}$.

Masses communes Toutes les masses des différents composants électriques et électroniques de la plateforme mobile, y compris la batterie et la carte Arduino Uno, sont reliées ensemble pour former une masse commune. Cette configuration garantit un référentiel de tension stable pour l'ensemble du système.

5.1.4 Problèmes rencontrés et solutions apportées

Au cours du développement du système électrique de la plateforme mobile, plusieurs problèmes ont été rencontrés :

Problème	Solution apportée
Surchauffe et défaillance du régulateur de tension	Remplacement du régulateur
Câbles de prototypage fondus lors des pics de courant	Utilisation de câbles multibrins de section renforcée pour l'alimentation de puissance
Court-circuit dans le support de batteries	Remplacement du support de batteries

TABEAU 12 – Problèmes rencontrés et solutions apportées lors du développement du système électrique

5.2 Schéma électrique

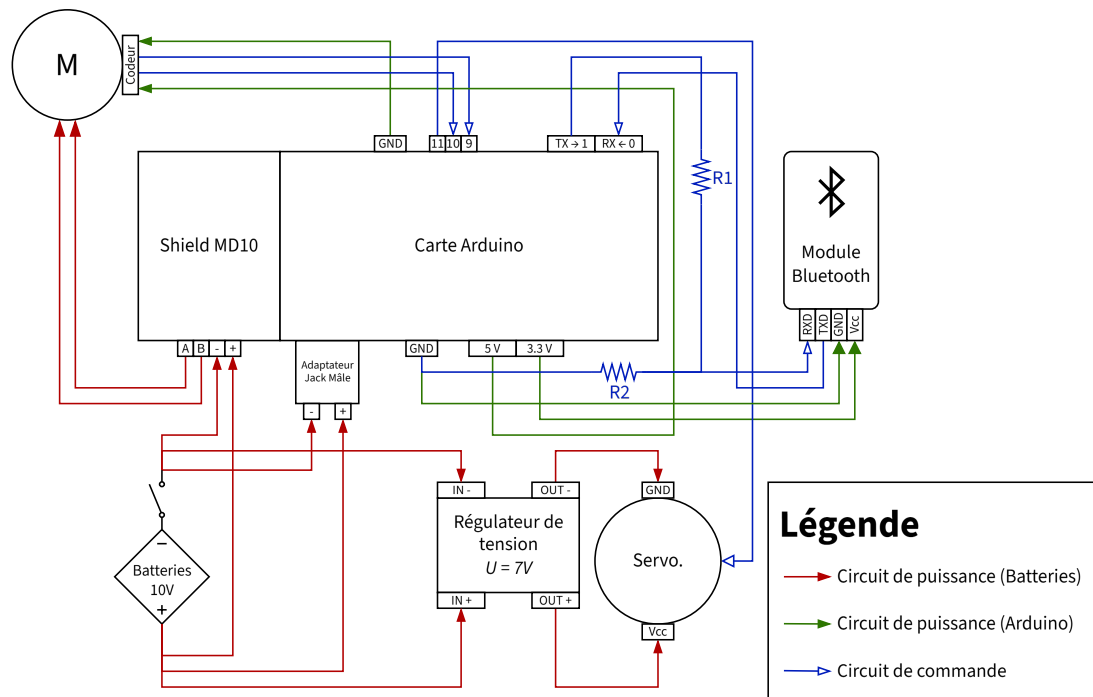


FIGURE 66 – Schéma électrique des différents composants de la plateforme.

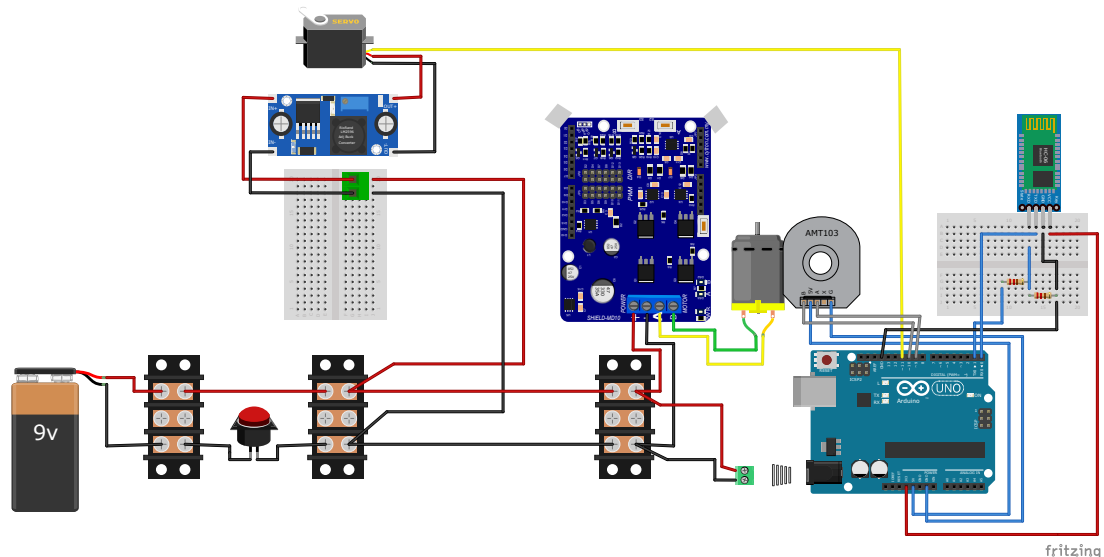


FIGURE 67 – Schéma du circuit électrique réalisé avec Fritzing.



Sur le schéma Fritzing, certains composants utilisés sont légèrement différents de ceux réellement employés dans la plateforme mobile (notamment les borniers à vis et le codeur incrémental), en raison de contraintes liées à la disponibilité des modèles.

6 LOGICIEL

+++++++[>+++++++>+++++++>+++++++>+++++++>+++++++>+++++++
++++>+++++++>+++++++>+++++++>+++++++<<<<<<-]>----.>+.>+
++.>----.>+.>----.>+.>--.

6.1 Architecture logicielle globale

Le système repose sur une architecture distribuée :

- **Niveau Bas (Temps Réel)** : Le microcontrôleur Arduino assure l'asservissement fin de la vitesse et de la position, la lecture des codeurs par interruption et le pilotage des actionneurs (Moteur CC et Servo).
- **Niveau Haut (Supervision)** : Une interface Web moderne (Web Serial API) permet le contrôle manuel, la visualisation de la télémétrie et le calcul de trajectoire pour la navigation 2D.

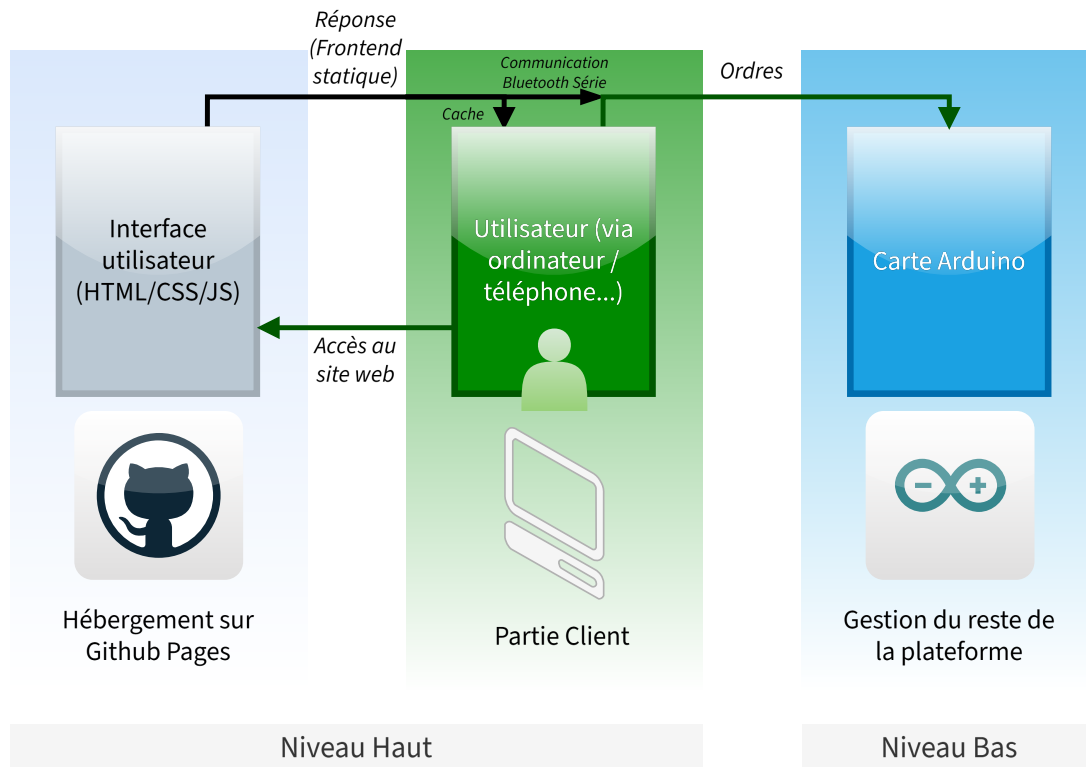


FIGURE 68 – Architecture logicielle globale de la plateforme mobile

6.2 Logiciel embarqué (Firmware Arduino)

Le micrologiciel est développé en C++ et s'appuie sur une gestion rigoureuse des ressources temporelles pour garantir la stabilité de l'asservissement.

6.2.1 Ordonnancement et Temps Réel

Pour s'affranchir des gigue (jitter) inhérentes à une boucle principale standard, l'échantillonnage de l'asservissement est piloté par interruption matérielle via le *Timer 2* du microcontrôleur (bibliothèque *FLexiTimer2*). La fréquence d'échantillonnage est fixée à 20 Hz ($T_e = 50$ ms).

6.2.2 Acquisition et Optimisation

La lecture des codeurs est critique. Elle est assurée par des interruptions externes (sur les broches 2 et 3) utilisant la lecture rapide des registres (`digitalWriteFast.h`) pour minimiser l'overhead. La méthode "X2" (lecture sur changement d'état des deux phases A et B) permet de doubler la résolution effective du capteur.

6.2.3 Stratégie d'Asservissement : Cascade Position-Vitesse

L'architecture de contrôle implémente une boucle en cascade pour gérer les modes "Vitesse" et "Position" :

1. **Boucle Interne (Vitesse) :** Elle utilise la bibliothèque `PID_v1`. Le PID calcule la commande PWM nécessaire pour maintenir la vitesse de rotation de la roue (ω) égale à la consigne.
2. **Boucle Externe (Position) :** Lorsque le mode position est activé (commande P), un correcteur proportionnel calcule une consigne de vitesse dynamique selon l'erreur de distance restante :

$$V_{consigne} = K_p \cdot (Pos_{cible} - Pos_{actuelle})$$

Cette consigne est saturée à ± 150 mm/s et injectée dans la boucle de vitesse. Une décélération progressive est logiciellement imposée à l'approche de la cible pour éviter le dépassement (overshoot).

6.3 Interface Homme-Machine (IHM)

L'interface utilisateur abandonne les serveurs backend lourds au profit d'une approche *Client-Side* pure, exploitant les capacités des navigateurs modernes basés sur la technologie Chromium.

6.3.1 Communication via Web Serial API

L'application web communique directement avec le port série (USB ou Bluetooth émulé) via l'API `navigator.serial`. Cela permet une latence minimale et une connexion "Plug & Play" sans installation de drivers spécifiques.

6.3.2 Protocole de Communication

Un protocole ASCII léger, orienté ligne, a été défini pour les échanges. Chaque trame suit le format TYPE:VALEUR :

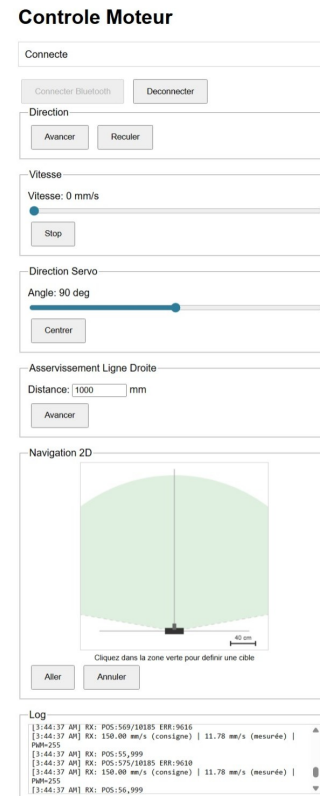


FIGURE 69 – Capture d'écran de l'interface utilisateur conçue

TABLEAU 13 – Jeu de commandes du protocole série

Commande	Syntaxe	Description
Vitesse	V:100	Définit la vitesse linéaire (0-150 mm/s).
Direction	D:1	Définit le sens de marche (1 : Avant, 0 : Arrière).
Angle	A:90	Pilotage du servomoteur (0-180°, 90° centré).
Position	P:500	Lance un asservissement en position sur 500 mm.

6.3.3 Algorithme de Navigation 2D

L'interface intègre un module de navigation visuelle sur *Canvas HTML5*. Le calcul de trajectoire est déporté sur le navigateur (JavaScript) :

1. L'utilisateur définit une cible (x, y) dans la zone verte valide (cône de braquage $\pm 80^\circ$).
2. Une boucle de contrôle JavaScript (10 Hz) calcule en temps réel l'angle de cap et la distance.

3. L'angle du servomoteur est ajusté dynamiquement ($Angle_{servo} = 90^\circ + \theta_{cible}$) tandis que la vitesse est régulée jusqu'à l'atteinte de la zone de tolérance (< 15 px).

Cette approche déporte "l'intelligence" géométrique vers l'interface, laissant le microcontrôleur se concentrer sur la régulation bas niveau.



Sur le canevas HTML5 de l'interface de navigation 2D, l'échelle est fixée à 5 mm par pixel.

Conclusion Générale

Ce projet d'ingénierie nous a permis de concevoir, fabriquer et asservir une plateforme mobile complète. Sur le plan mécanique, la structure s'avère fonctionnelle et robuste. Sur le plan électronique et automatique, l'identification du moteur a permis de dimensionner un correcteur PI efficace, assurant un suivi de vitesse précis et une réponse dynamique adaptée. Enfin, l'interface logicielle offre une ergonomie satisfaisante pour le pilotage.

Le prototype final répond à la majorité des exigences du cahier des charges, notamment en termes de maniabilité et d'autonomie. Des améliorations pourraient être apportées sur la qualité du câblage pour fiabiliser le système en vue d'une utilisation industrielle.

```
+++++++ [ >+++++++>+++++++>+++++++>+++++++>+++++++>++++>+++++++>
>++++>+++++++>+++++++>+++++++>+++++++>+++++++>++++>+++++++>
>+++++++>++++>+++++++>+++++++>++++>+++++++>+++++++>+++++++>++++
+++++++>+++++++>+++++++>++++>++++>+++++++>+++++++>+++++++>++++
+++++++>++++>+++++++>+++++++>+++++++>+++++++>+++++++>+++++++
++++>+++++++>+++++++>++++>+++++++>+++++++>+++++++>++++>+++++++
++++>+++++++>+++++++>+++++++>++++>+++++++>+++++++>+++++++
++>+++++++>+++++++>++++>+++++++>+++++++>+++++++>+++++++>++++
++++>+++++++>+++++++>++++>+++++++>+++++++>+++++++>++++>+++++++
++>+++++++>+++++++>+++++++>+++++++>+++++++>+++++++>++++>+++++++>
+++++++>+++++++>+++++++>+++++++>+++++++>+++++++>+++++++>++++<<<<
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<-
]>---, >+, >++++, >-, >-----, >+, >, >-, >---, >--, >+, >-----, >++++, >+, >--, >---, >+, >-, >+
, >+, >++++, >---, >+, >, >+, >++++, >---, >-----, >+, >--, >+, >-----, >-----, >+, >+, >-----, >+
+, >+, >++++, >+, >, >-----, >+, >++++, >---, >+, >+, >--, >+, >---, >-----, >+, >---, >---, >++++
+, >+, >+, >+, >---, >+, >, >++++, >+, >-, >-----, >+, >+, >-, >+, >-----, >+, >+, >+, >-----, >-
-----, >-----, >-----, >+, >, >+, >-----, >---, >-----, >--, >-----, >-----.
```

Annexe

1. Code source de l'interface graphique disponible sur GitHub : <https://github.com/Samy-Y/g13-plateforme-mobile>
2. Code source de la solution de traitement des expériences de caractérisation du moteur CC : <https://github.com/Samy-Y/g13-moteur-cc>