

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITE DE BLIDA 1  
FACULTE DE TECHNOLOGIE  
DEPARTEMENT DE MECANIQUE



**Mémoire de fin d'études**

En vue de l'obtention du Diplôme de Master en Génie

Mécanique

**Spécialité :** Fabrication Mécanique et Productique

**Thème :**

# **Conception et réalisation d'une machine CNC 3 axes à architecture parallèle**

**Réalisé par :**

SMADI Younes.

LAKAS Mohamed Aymen.

**Encadré par :**

Mr. CHETTIBI Taha      promoteur.

Mr. LOUNICI Billel      Co-promoteur.

Année universitaire 2021/2022

## ***Remerciement***

*Nous tenons à exprimer notre sincère reconnaissance à l'égard de tous ceux qui ont contribué à la réalisation de ce travail, particulièrement à notre promoteur « Dr Chettibi Taha » et Co-promoteur « Mr Lounici Billel » pour ses conseils et suggestions.*

*Nous remercions également les membres du jury qui sont honorés d'examiner notre modeste travail.*

# *Dédicaces*

*Je dédie ce travail tout  
particulièrement aux personnes qui  
me sont les plus chères au monde ;  
Mes très chers parents pour leur  
patience, soutien et confiance,  
Je dédie ce travail également à tous  
les membres de ma famille,  
A tous ceux qui m'ont encouragé...*

*Younes*

---

*Je dédie ce travail  
A mes chers parents pour tous les  
efforts consentis pour ma formation ;  
les membres de ma famille,  
A tous ceux qui m'aiment...*

*Aymen*

## Résumé

Le monde de la machine-outil à commande numérique (CNC) est en évolution permanente afin de répondre aux besoins croissants des utilisateurs en termes de coût, délais, flexibilité et qualité. C'est dans ce contexte que les CNC à structures parallèles robotisées ont été introduites grâce à leurs performances remarquables en termes de précision, rigidité et rapidité.

Notre projet de fin d'études a pour objectif la mise en œuvre d'une petite machine CNC 3 axes à architecture parallèle. La partie mécanique responsable du positionnement de l'outil dans le plan vertical est basée sur la structure d'un robot parallèle de type biglide à deux degrés de liberté qui assure aussi bien la rigidité que la précision de positionnement.

L'étude a porté dans un premier temps sur les aspects de modélisation et simulation, à l'aide de SolidWorks et Matlab, des principales parties de la machine. Ensuite, nous nous sommes intéressés aux choix des solutions technologiques appropriées pour mettre en œuvre la machine dans les limites de nos moyens financiers et de ce qui est disponible sur le marché local. La machine a été réalisée et testée avec succès pour des opérations d'usinage simples ce qui nous a permis de valider le concept proposé. Nous espérons que ce modeste travail pourrait servir de base aux futurs projets dans le domaine de l'usinage robotisé.

**Mots clés:** conception, réalisation, CNC, arduino, GRBL, MGD, SolidWorks, Matlab.

## ملخص

يتطور عالم أدوات ماكينات CNC باستمرار لتلبية الاحتياجات المتزايدة للمستخدمين من حيث التكلفة والوقت والمرونة والجودة. في هذا السياق ، تم تقديم ماكينات CNC ذات الهياكل الروبوتية المتوازية بفضل أدائها الرائع من حيث الدقة والصلابة والسرعة.

يهدف مشروع نهاية الدراسات الخاص بنا إلى تنفيذ آلة CNC صغيرة ذات 3 محاور ببنية متوازية. يعتمد الجزء الميكانيكي المسؤول عن وضع الأداة في المستوى العمودي على هيكل روبوت موازٍ من نوع biglide بدرجتين من الحرية مما يضمن صلابة ودقة تحديد الموقع.

ركزت الدراسة في البداية على جوانب النمذجة والمحاكاة ، باستخدام SolidWorks و Matlab ، للأجزاء الرئيسية من الجهاز. ثم اهتمنا باختيار الحلول التكنولوجية المناسبة لتنفيذ الماكينة في حدود إمكانياتنا المالية وما هو متوفر في السوق المحلي. تم بناء الماكينة واختبارها بنجاح لعمليات تشغيل آلية بسيطة ، مما مكّننا من التحقق من صحة المفهوم المقترح. نأمل أن يكون هذا العمل المتواضع بمثابة أساس لمشاريع مستقبلية في مجال الآلات الروبوتية.

**الكلمات المفتاحية:** تصميم ، إنتاج ، CNC ، اردوينو ، GRBL ، MGD ، SolidWorks ، Matlab .

## **Abstract**

The world of CNC tools-machine is constantly evolving to meet the growing needs of users in terms of cost, time, flexibility and quality. That's why CNC parallel robot structures are introduced because of their remarkable performance in terms of precision, rigidity and speed.

Our project consists to implement a small 3 axis CNC machine with parallel architecture. The mechanical part responsible for positioning the tool in the vertical plane is based on the structure of a parallel robot biglide type with two degrees of freedom which ensures both rigidity and positioning precision.

The study has been, first off all, focused on modeling and simulation aspects, using SolidWorks and Matlab, of the main parts of the machine. Then, we were interested in choosing the appropriate technological solutions to implement the machine within the limits of our financial means and what is available on the local market. The machine was successfully built and tested for simple machining operations, which enabled us to validate the proposed concept. We hope that this modest work could serve as a basis for future projects in the field of robotic machining.

**Keywords:** design, production, CNC, arduino, GRBL, MGD, SolidWorks, Matlab.

# *Sommaire*

Introduction générale .....	1
-----------------------------	---

## **Chapitre 1 : Généralités sur les robots d'usinage**

1.1) Introduction.....	3
1.2) Evolution des robots dans l'industrie .....	3
1.3) Architectures des robots industriels utilisés pour l'usinage .....	5
1.3.1) Les manipulateurs à architecture sérielle .....	5
1.3.2) Les robots manipulateurs à architecture parallèle .....	8
1.3.3) Les robots manipulateurs à architecture hybride .....	11
1.4) Conclusion .....	12

## **Chapitre 2 : Modélisation d'un robot parallèle plan à deux degrés de liberté**

2.1) Introduction .....	13
2.2) Description du manipulateur parallèle 2-DoF .....	14
2.3) Modélisation géométrique du robot .....	15
2.4) Modélisation cinématique du robot .....	18
2.4.1) Modélisation cinématique indirecte (MCI) .....	18
2.4.2) Modélisation cinématique directe (MCD) .....	18
2.5) Espace de travail du manipulateur .....	21
2.6) Conclusion .....	22

## **Chapitre 3 : Conception de la machine**

3.1) Introduction .....	23
3.2) Analyse du besoin .....	23
3.3) Etude de la faisabilité .....	25

3.3.1) Analyse fonctionnelle externe .....	26
3.3.2) Analyse fonctionnelle interne .....	27
3.4) Cahier des charges fonctionnel .....	28
3.5) Solutions technologiques .....	29
3.5.1) Choix N°1 .....	29
3.5.2) Choix N°2 .....	30
3.5.3) Choix N°3 .....	30
3.5.4) Sous systèmes de déplacement (X, Y et Z).....	31
3.5.4.1) Transformer le mouvement de rotation en mouvement de translation.....	31
3.6) Analyse de l'espace de travail du robot .....	33
3.6.1) Limites de l'espace de travail du robot dans le plan (xy) .....	33
3.6.2) Limites de l'espace de travail du robot selon l'axe z .....	36
3.7) Etude et calcul des principaux organes de la machine .....	36
3.7.1) Estimation des masses des éléments de la machine .....	36
3.7.2) vérification des vis trapézoïdales .....	37
3.7.2.1) Calcul de la charge F applique sur la vis suivant les axes Y et Z...37	
3.7.2.2) vérification des vis trapézoïdales à l'usure .....	39
3.7.2.3) vérification des vis trapézoïdales à la flexion .....	43
3.8) Choix des moteurs .....	44
3.9) Dimensionnement des roulements .....	45
3.10) Conclusion .....	46

## **Chapitre 4 : Simulation du comportement de la machine**

4.1) Introduction aux études de mouvement sous SolidWorks .....	47
4.2) Vue d'ensemble de la fenêtre étude de mouvement .....	48
4.3) Eléments mécaniques pour la mise en œuvre d'une étude mouvement .....	49
4.3.1) Les Moteurs .....	51
4.4) Chronogramme .....	53
4.5) Arbre de création Motion Manager .....	54
4.6) Contraintes dans les études de mouvement .....	55



4.6.1) Contrainte pivot .....	56
4.6.2) Contraintes de trajectoire dans les études de mouvement .....	57
4.7) Application au cas de notre machine CNC .....	58
4.7.1) Simulation du sous-ensemble Moteur / glissière .....	58
4.7.2) Simulation du sous-ensemble Table-porte-pièce .....	61
4.7.3) Simulation du sous-ensemble robot parallèle .....	62
4.8) Comparaison des résultats de simulation Matlab avec celle SolidWorks Motion.....	68
4.9) Conclusion .....	71

## **Chapitre 5 : Présentation de la Partie électrique et informatique**

5.1) Introduction .....	72
5.2) Configurer GRBL et contrôler la machine CNC avec Arduino .....	72
5.3) Matériel électronique .....	73
5.3.1) Description des principaux éléments utilisés .....	73
5.3.2) Assemblage des composants .....	75
5.4) Comment installer le GRBL .....	78
5.5) Comment communiquer avec GRBL .....	80
5.6) Configuration du GRBL .....	81
5.7) Exemples d'application .....	83
5.8) Conclusion .....	87
<b>Conclusion générale :</b> .....	<b>88</b>

## *Liste des figures*

- Figure 1.1 :** Approvisionnement mondial de robots.
- Figure 1.2 :** Installations annuelles mondiales des RI par secteurs industriels.
- Figure 1.3 :** Robot Manipulateur à architecture sérielle.
- Figure 1.4 :** Robot pour l'usinage ABB IRB 6660.
- Figure 1.5 :** Robot ABB pour le fraisage.
- Figure 1.6 :** Robot Kuka pour l'usinage d'une pièce de grande dimension.
- Figure 1.7 :** Robot RX170 hsm de Staübli.
- Figure 1.8 :** Robot Manipulateur à architecture parallèle.
- Figure 1.9 :** Exemples d'architectures de robots planaires.
- Figure 1.10 :** Robot Delta à trois degrés de liberté.
- Figure 1.11 :** Plate-forme de Gough.
- Figure 1.12 :** Robots parallèles de type SSM, TSSM et MSSM.
- Figure 1.13 :** Machine-outil Variax (Gidding & Lewis).
- Figure 1.14 :** Vue d'ensemble de la MOCN VERNE.
- Figure 1.15 :** L'Exechon concept.
- Figure 1.16 :** Tricept 845 (Neos Robotics).
- Figure 2.1 :** Architecture du manipulateur parallèle planaire 2-ddl étudié.
- Figure 2.2 :** Schéma cinématique du manipulateur et paramétrage.
- Figure 2.3 :** Fonction Matlab implémentant le modèle géométrique inverse.
- Figure 2.4 :** Fonction Matlab implémentant le modèle géométrique directe.
- Figure 2.5 :** Fonction Matlab implémentant la Jacobienne du modèle MCD.
- Figure 2.6 :** Fonction Matlab implémentant la Jacobienne du modèle MCI.
- Figure 2.7 :** Programme Matlab utilisant les différentes fonctions Modélisant notre robot.
- Figure 2.8 :** Limites du volume de travail du robot.
- Figure 3.1 :** Les besoins client identifiés.

**Figure 3.2 :** Diagramme bête à corne.

**Figure 3.3 :** étapes de l'analyse fonctionnelle.

**Figure 3.4 :** Diagramme Pieuvre.

**Figure 3.5 :** Diagramme FAST de la fonction principale du produit ciblé.

**Figure 3.6 :** Diagramme FAST de la fonction FT31.

**Figure 3.7 :** Choix technologique N°1.

**Figure 3.8 :** Choix technologique N°2.

**Figure 3.9 :** Choix technologique N°3.

**Figure 3.10 :** Système vis-écrou (vis à billes et vis trapézoïdale).

**Figure 3.11 :** Système pignon-crémaillère.

**Figure 3.12 :** Système poulies-courroie.

**Figure 3.13 :** Etat 0 du robot.

**Figure 3.14 :** Valeurs limites des variables  $y_i$  correspondant au déplacement extrême selon l'axe x.

**Figure 3.15 :** Cas limites correspondants au contact de l'outil avec la Table.

**Figure 3.16 :** Les limites du volume de travail du robot pour un point de la nacelle.

**Figure 3.17 :** Les limites du volume de travail du robot avec un outil de 115 mm.

**Figure 3.18 :** Volume de travail final.

**Figure 3.19 :** Limites du volume de travail selon l'axe z.

**Figure 3.20 :** Système de translation suivant Y.

**Figure 3.21 :** Système de translation suivant Z.

**Figure 3.22 :** Condition de glissement des écrous de bronze sur les vis en acier.

**Figure 3.23 :** Diagramme des forces et des réactions.

**Figure 3.24 :** Diagramme du moment fléchissant.

**Figure 4.1 :** Éléments de base de l'Interface utilisateur pour SolidWorks Motion.

**Figure 4.2 :** Les lignes d'un chronogramme.

**Figure 4.3 :** Ensemble de contraintes disponibles dans SolidWorks.

**Figure 4.4 :** Sous-Ensemble Moteur-vis-écrou.

**Figure 4.5 :** Différentes étapes suivies pour effectuer la simulation du sous-ensemble Moteur-Vis-Ecrou.

**Figure 4.6 :** L'évolution temporelle du déplacement de l'écrou de la vis.

**Figure 4.7 :** L'évolution temporelle du déplacement angulaire de la vis.

**Figure 4.8 :** Sous-Ensemble Moteur-vis-écrou-table.

**Figure 4.9 :** Caractéristique de la vitesse imposée (profil trapézoïdal).

**Figure 4.10 :** La puissance du moteur.

**Figure 4.11 :** Assemblage étudié lors du suivi d'une trajectoire.

**Figure 4.12 :** choix du type d'analyse et du moteur à utiliser.

**Figure 4.13a :** Trajectoire circulaire imposée à la nacelle.

**Figure 4.13b :** Étapes suivies pour imposer le suivi d'une trajectoire circulaire par un point de référence pris sur la nacelle.

**Figure 4.14 :** Étapes suivies pour définir le profil de mouvement sur la trajectoire imposée.

**Figure 4.15 :** Étapes suivies pour afficher les résultats de simulation.

**Figure 4.16 :** Graphe position/vitesse/ accélération d'un point de la nacelle.

**Figure 4.17 :** Graphe position/vitesse/ accélération d'un point de l'écrou 1 ( $q_1$ ).

**Figure 4.18 :** Graphe position/vitesse/ accélération d'un point de l'écrou 2 ( $q_2$ )

**Figure 4.19 :** Graphe position/vitesse/ accélération d'un moteur.

**Figure 4.20 :** la puissance consommée pour la trajectoire spécifiée.

**Figure 4.21 :** Programme de dessin (cercle, linge H/V).

**Figure 4.22 :** les chemins générés sous Matlab.

**Figure 4.23 :** Résultat de Y1.

**Figure 4.24 :** Résultat de Y2.

**Figure 4.25 :** Dessin sous SolidWorks des chemins importés.

**Figure 5.1 :** Communication du G-code vers la machine CNC.

**Figure 5.2 :** Arduino et le shield utilisés.

**Figure 5.3 :** Driver DRV8825.

**Figure 5.4 :** Le moteur pas a pas Nema 17.

**Figure 5.5 :** Capteur de fin de course.

**Figure 5.6 :** Alimentation 12V 5A.

**Figure 5.7 :** Schéma de câblage des différents éléments électriques.

**Figure 5.8 :** Position de montage du Shield.

**Figure 5.9 :** Placement les cavaliers de micro-pas

**Figure 5.10 :** Placements possibles des cavaliers pour configurer les drivers.

**Figure 5.11 :** comment placer correctement un driver sur un shield.

**Figure 5.12 :** Placement potentiel des câbles moteurs.

**Figure 5.13 :** Placement de l'alimentation et des câbles moteurs.

**Figure 5.14 :** Montage des capteurs fin de course sur la machine.

**Figure 5.15 :** Montage de capteur fin-course en Arduino.

**Figure 5.16 :** Téléchargement du GRBL.

**Figure 5.17 :** IDE Arduino.

**Figure 5.18 :** Sélection de GRBL.

**Figure 5.19 :** Interface CNC JS.

**Figure 5.20 :** Sélection l'arduino.

**Figure 5.21 :** La section "Console".

**Figure 5.22 :** Réglage du paramètre pas/mm.

**Figure 5.23 :** Fenêtre de la Concole de CNCjs.

**Figure 5.24 :** Programme Matlab pour perçage et usinage.

**Figure 5.25 :** Introduction du programme à exécuter (macro) pour suivre un chemin précis.

## *Liste des tableaux*

**Tableau 3.1 :** tableur questions/réponses relatif au besoin.

**Tableau 3.2 :** Description des fonctions de service.

**Tableau 3.3 :** Caractérisation de fonctions de services.

**Tableau 3.4 :** Les masses des différents composants du dispositif.

**Tableau 3.5 :** Coefficients de sécurité pour différents types de charges.

**Tableau 4.1 :** Principaux outils de Motion Manager.

## *Liste des symboles*

**m** : La masse [Kg].

**g** : La gravité [N/Kg].

**p** : poids [N].

**F<sub>c</sub>** : est l'effort de coupe [N].

**F<sub>p</sub>** : L'effort de pénétration.

**P** : Pression superficielle de contact.

**V<sub>st</sub>** : Vitesse de frottement sur le diamètre moyen du filetage.

**f<sub>i</sub>** : Coefficients de sécurité.

**A<sub>t</sub>** : Surface d'appui totale entre les dents de la vis et les dents de l'écrou [mm<sup>2</sup>].

**C<sub>z</sub>** : Couples de moteurs [N.mm].

**P<sub>uz</sub>** : Puissance de moteurs [W].

**R<sub>A</sub>** : Réaction au point A [N].

**R<sub>B</sub>** : Réaction au point B [N].

**M<sub>f</sub>** : Moment fléchissant [N.mm].

**L** : Longueur [mm].

**D** : Le diamètre [mm].

**σ<sub>f</sub>** : Contrainte de flexion [MPa].

**σ<sub>fadm</sub>** : Contrainte de flexion admissible.

**C** : Capacité de charge dynamique [KN].

**L<sub>h</sub>** : Durée de vie d'un roulement [h].

# *Introduction générale*



L'ère moderne de l'industrie est marquée par une large utilisation des robots. On en trouve partout : des robots pour les applications d'assemblage, de tri, de manutention et même de contrôle. Mais la révolution dans ce domaine vient lors de leur utilisation dans les opérations d'usinage. En fait, l'apport de l'utilisation des robots n'est plus à démontrer. En effet, grâce à leur implantation le long d'une chaîne de production, on arrive à augmenter considérablement le rendement et la qualité et à diminuer les coûts et les temps d'exécution. En outre, le monde de la machine-outil est en évolution permanente afin de répondre aux besoins croissants des utilisateurs, lesquels ont des contraintes de flexibilité de plus en plus fortes : on veut atteindre des temps de cycle de plus en plus court tout en respectant la qualité des produits. C'est là où les structures parallèles robotisées viennent pour apporter des performances remarquables au monde de la machine-outil, grâce à leur précision ainsi qu'à leurs qualités dynamiques.

Un robot parallèle est un système mécanique qui utilise plusieurs membres contrôlés par ordinateur pour prendre en charge une plate-forme commune ou un effecteur final. Comparé à un robot sériel, un robot parallèle a généralement une précision et des performances dynamiques plus élevées et, par conséquent, peut être utilisé à de nombreuses applications. La recherche sur les robots parallèles a attiré beaucoup d'attention au cours des quatre dernières décennies, en particulier leur utilisation dans le domaine de la fabrication mécanique comme structure de base de machines-outils à commande numérique. C'est dans ce cadre que s'inscrit la présente étude qui cible la mise en œuvre d'une petite machine CNC à architecture parallèle, elle se trouve par conséquent à la frontière de la robotique parallèle et du monde de la machine-outil.

Pour mettre au point ce travail, nous avons adopté une approche originale qui se base essentiellement sur l'exploitation des outils informatiques et matériels développés pour la communauté des **Makers**. La culture maker (de l'anglais make, i.e. faiseur) est une culture contemporaine constituant une branche de la culture **Do It Yourself (DIY)** tournée vers la technologie et la création en groupe. Généralement, les makers prennent part à des projets orientés ingénierie. Les domaines typiques de ces projets sont ainsi l'électronique, la robotique, l'impression 3D et l'usage des machines-outils à commande numérique (CNC), mais également des activités plus traditionnelles telles que la métallurgie, la menuiserie, les arts traditionnels et l'artisanat.

Afin d'expliquer convenablement notre travail, le présent mémoire s'ordonne de la manière suivante.

La lecture du premier chapitre intitulé Généralités sur les robots d'usinage permettra de connaître l'état de l'art de la robotique industrielle et de situer notre travail parmi quelques travaux qui se sont attaqués au problème d'intégration des structures robotisées dans le domaine de fabrication.

Le deuxième chapitre abordera la modélisation d'un robot parallèle plan à deux degrés de liberté, cette architecture sera la base de notre Machine-outil robotisée. Nous détaillerons en particuliers les modèles géométriques et cinématiques direct et inverse ainsi que leur implémentation sous Matlab. Ces modèles sont indispensables pour avoir des

informations sur l'espace de travail de la machine ainsi que l'élaboration de ses commandes dans les limites admissibles en termes de performances cinématiques.

Le troisième chapitre aborde la conception propre de la machine. La procédure commence par une analyse fonctionnelle extérieure puis intérieure, ce qui nous a permis de faire des choix rationnels en termes des solutions technologiques en fonction d'un côté le cahier de charges et de l'autre côté des moyens disponibles sur le marché local. La procédure de conception est mise en œuvre essentiellement dans SolidWorks Le dimensionnement des différents organes est aussi discuté dans ce chapitre.

Dans le but de valider les modèles analytiques développés dans le chapitre 2, le quatrième chapitre regroupe l'ensemble des simulations faites à l'aide de l'outil Motion de SolidWorks sur la partie robot parallèle porteur de l'outil. Une présentation détaillée de l'outil utilisée est donnée ainsi que les différentes étapes suivies pour réaliser les différentes simulations.

Le cinquième chapitre discute de la partie électrique et informatique de notre machine. Notre choix s'est porté sur des solutions Open-sources qu'on a adaptés pour notre machine. Le chapitre décrit l'essentiel des choix effectués et comment on les a adaptés pour notre application.

Enfin, le manuscrit se termine par une conclusion générale dans laquelle on résume les travaux réalisés ainsi que les perspectives de notre travail.

*Chapitre I :*

*Généralité sur les robots  
d'usinage*

## 1.1) Introduction

La fabrication moderne se caractérise par une forte personnalisation des outils de production pour une fabrication optimale des pièces manufacturées. En général, le processus de fabrication des pièces manufacturées repose souvent sur un procédé d'usinage permettant l'obtention de formes par enlèvement de matière. Ce procédé requiert généralement l'utilisation d'une machine-outil à commande numérique MOCN. Actuellement, les MOCN effectuent majoritairement des opérations d'usinage, car elles sont capables de garantir une précision d'usinage élevée avec une bonne stabilité, cependant les coûts de ces machines sont relativement élevés et leurs fonctions sont uniques. A cet effet, une machine multifonctionnelle et à faible coût est une alternative intéressante pour remplacer les machines-outils classiques. A cet effet, un **Robot Industriel (RI)** pourrait être une solution potentielle.

L'utilisation des robots pour des opérations d'usinage est devenue un enjeu de développement des robots ces dernières années. L'objectif est de proposer une alternative aux machines-outils classique. Toutefois, les premiers robots développés présentaient des précisions de suivi de trajectoire uniquement de l'ordre du dixième de millimètre, même si leur rigidité a été améliorée par rapport aux robots utilisés pour des opérations moins exigeantes. En effet, les comportements géométrique, statique et dynamique des architectures classiques des robots, particulièrement sérielles, ne permettent pas de réaliser des usinages avec une grande précision. En conséquence, les regards des chercheurs se sont tournés vers les robots d'usinage à architecture parallèle. L'utilisation de machines-outils à structure parallèle devrait permettre d'accroître la productivité de l'usinage par rapport aux machines-outils à structure sérielle pour des performances cinématiques des axes similaires.

Les robots parallèles ont déjà fait leurs preuves pour des applications médicales ou de simulateurs de vol. Dans le domaine industriel, ils réalisent efficacement des tâches de manutention ou d'assemblage. On exploite pour cela leurs fortes vitesses et accélérations, grâce à la faible masse des éléments en mouvement. L'idée d'employer ces architectures pour des opérations d'usinage et en particulier pour l'UGV (usinage à grande vitesse) est relativement récente. Il a fallu en effet attendre le développement de certaines technologies et l'émergence d'une nouvelle génération de calculateurs numériques pour voir apparaître les premières machines à structure parallèle capables de réaliser des usinages à grande vitesse de type industriel.

## 1.2) Evolution des robots dans l'industrie

Au cours des cinq dernières décennies, les applications des robots se sont considérablement développées afin de répondre à des problématiques de maîtrise de la qualité du produit, de la productivité et des coûts. Selon les rapports de la Fédération Internationale de Robotique (IFR) [1], le nombre entre 2011 et 2016 a été porté à 212 000 unités, par rapport au nombre annuel moyen de robots vendus entre 2005 et 2008, soit environ 115 000 unités, soit une augmentation de 84 %, et le nombre estimé en 2020 est de 520 900 unités. A titre indicatif, la Figure 1.1 illustre l'évolution du nombre de robots installés à travers le monde pour la période 2015-2024. Par ailleurs, Le rapport de l'année 2021 de l'IFR montre un record de 3 millions de robots industriels opérant dans des usines

du monde entier, soit une augmentation de 10 %. Les ventes de nouveaux robots ont légèrement augmenté de 0,5 % malgré la pandémie mondiale, avec 384 000 unités expédiées dans le monde en 2020. Cette tendance a été dominée par l'évolution positive du marché en Chine, compensant les contractions des autres marchés. Il s'agit de la troisième année la plus réussie de l'histoire de l'industrie de la robotique, après 2018 et 2017.

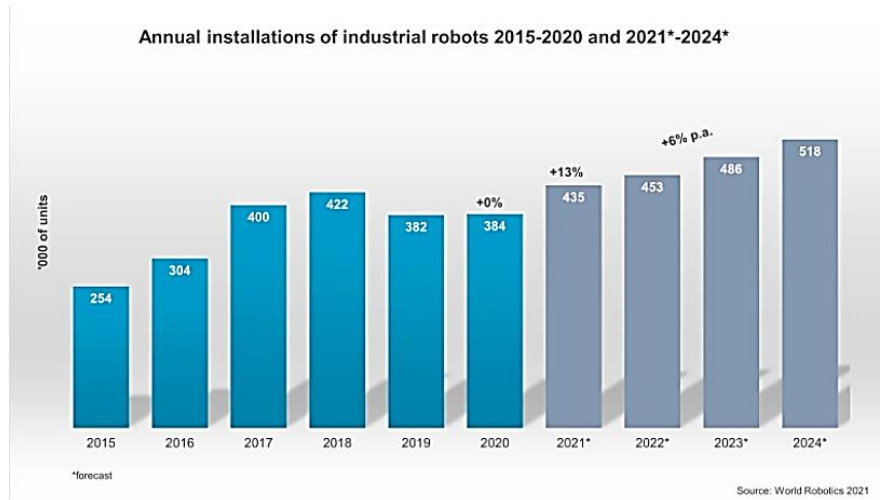


Figure 1.1. Approvisionnement mondial de robots [1].

Les robots industriels **RI** sont généralement utilisés pour l'exécution de tâches telles que le prélèvement et le placement, le soudage, la peinture, l'emballage et l'étiquetage, dans différentes applications comme les industries de l'électricité/électronique, l'automobile, le métal/machine, et l'agroalimentaire (Figure 1.2).

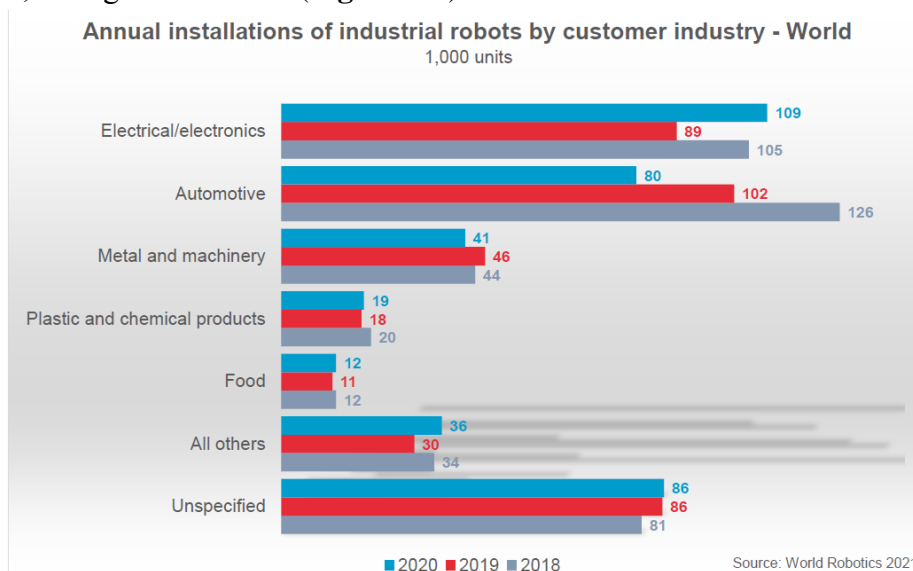


Figure 1.2. Installations annuelles mondiales des RI par secteurs industriels [1]

En 2018, l'industrie automobile était la plus demandeuse en robots industriels. Cependant, selon le rapport de IFR, l'année 2020 a créé un moment historique dans l'histoire de la robotique. L'industrie automobile a perdu sa position de premier client des robots

industriels. 79 849 unités (-22%) ont été installées. C'était près de 29 500 unités de moins que ce que l'industrie de l'électronique a reçu. L'industrie automobile était le client le plus important de robots industriels depuis l'installation de la toute première unité vendue dans le commerce à l'usine General Motors du New Jersey en 1961. En 2017, l'industrie électronique s'est rapprochée mais restait à 1 500 unités de moins que l'industrie automobile. En 2018, les installations de l'industrie électronique ont diminué, tandis que celles de l'industrie automobile ont atteint un nouveau niveau record de 125 581 unités, creusant un écart de 20 000 unités avec l'industrie électronique. Cet écart est resté important, quelque 13 000 unités en 2019, lorsque la demande de robots dans les deux industries a diminué.

La pandémie a contraint de nombreux fournisseurs et constructeurs automobiles à cesser temporairement leur production, car les chaînes d'approvisionnement mondiales ont été perturbées. Les produits en amont (inputs) n'étaient pas disponibles et les produits finaux (outputs) n'ont pas pu être livrés en raison de la fermeture des frontières et d'autres restrictions. La production mondiale de voitures et de véhicules utilitaires a chuté de 16 % en 2020.

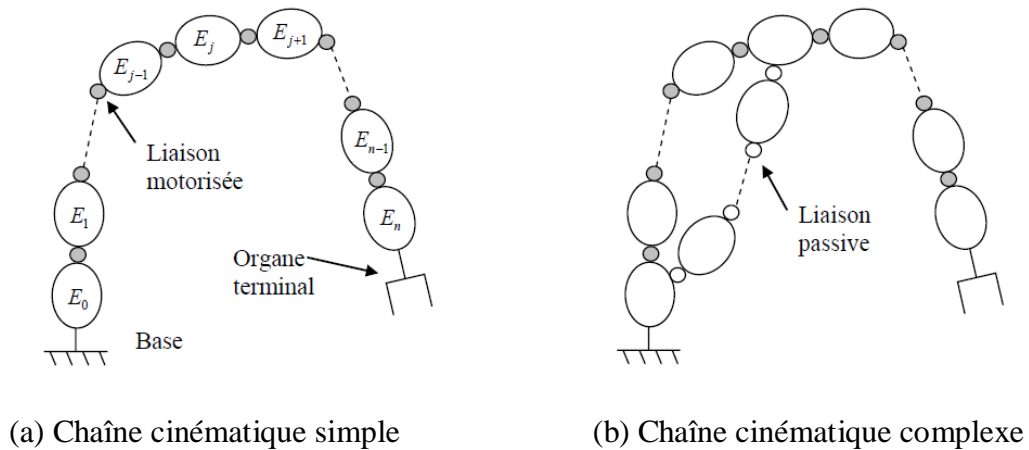
Par conséquent, de nombreux investissements majeurs ont été arrêtés ou reportés. Mais même avant la pandémie, la production mondiale de voitures et de véhicules utilitaires avait diminué deux années de suite, de 5,2 % en 2019 et de 1,1 % en 2018. Alors que l'industrie automobile doit investir dans la transition des moteurs à combustion vers les moteurs électriques, la diminution des limites de la demande la nécessité d'augmenter la capacité. De 2015 à 2020, les installations annuelles dans l'industrie automobile ont diminué de 4 % en moyenne chaque année. La part de l'industrie automobile dans le nombre total d'installations a diminué de manière continue, passant de 38 % en 2015 à seulement 21 % en 2020.

Selon le dernier rapport de La Fédération Internationale de Robotique, le «boom après crise» des installations des robots devrait s'estomper en 2022 à l'échelle mondiale. De 2021 à 2024, des taux de croissance annuels moyens dans la fourchette moyenne à un chiffre sont attendus. La barre des 500 000 unités installées par an dans le monde devrait être atteinte en 2024. Le marché nord-américain devrait croître un peu plus fort de 10 % en moyenne chaque année. Les attentes de croissance pour le marché européen sont un peu inférieures dans la fourchette basse à un chiffre. L'Europe centrale et orientale devrait mieux performer que l'Europe occidentale. Par ailleurs, le marché asiatique restera fort.

### **1.3) Architectures des robots industriels utilisés pour l'usinage**

#### **1.3.1) Les manipulateurs à architecture sérielle**

Les robots manipulateurs sériels sont les plus répandus dans le milieu industriel. Ils sont caractérisés par l'existence d'une chaîne cinématique unique entre l'effecteur et la base. Suivant le degré de connexion des différents éléments, cette chaîne cinématique peut être simple ou complexe [2]. Elle est dite *simple* (Figure 1.3.a) si les éléments intermédiaires possèdent un degré de connexion égale à deux. Elle est dite *complexe* (Figure 1.3.b) si au moins un élément intermédiaire possède un degré de connexion supérieur à deux.



**Figure 1.3.** Robot Manipulateur à architecture sérielle

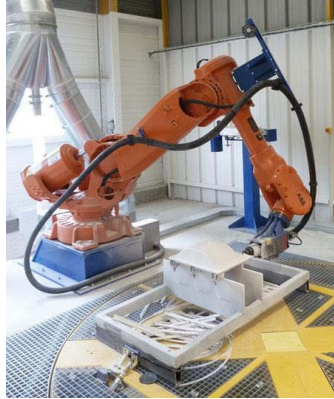
Dans un premier temps, les robots sériels étaient utilisés pour des opérations industrielles simples qui exigent une précision de remise en position et de suivi de trajectoire assez faible. Toutefois, à l'heure actuelle, les robots font de nombreuses autres tâches industrielles plus complexes. Ainsi, les robots sériels sont, de plus en plus, utilisés dans les secteurs aéronautiques et automobiles, pour des opérations de pré-usinage, perçage, polissage, découpage et parachèvement des pièces. Mais seulement 3% à 4% des RI sont dédiés à l'usinage [3] [4]. Les performances de ces robots sont généralement comparées à celle des machines-outils qui ont été spécifiquement conçues pour des opérations d'usinage avec une précision de l'ordre de 0,01 mm. Malgré la faible utilisation des RI sériels dans les opérations d'usinage, ces robots offrent un certain nombre d'avantages par rapport aux MOCN classiques notamment une adaptabilité élevée, un coût relativement bas, un grand volume de travail et un nombre plus important d'applications [4] [5].

Ainsi, les entreprises concevant et fabricant des robots continuent de développer des robots plus performants afin de proposer une alternative aux MOCN classiques. Nous pouvons citer la société ABB Robotique qui propose son robot à haute performance IRB 6660 (Figure 1.4), son premier robot dédié au pré-usinage des pièces de fonderie (opérations de fraisage, découpage ou de sciage) [5]. L'avantage de ce robot vis-à-vis d'une machine-outil repose sur son grand rayon d'action tout en respectant une certaine capacité de charge (le robot ABB 6660-205/1.9 a un rayon d'action de 1,9 m. et une capacité de charge de 205 Kg.).



**Figure 1.4.** Robot pour l'usinage ABB IRB 6660 [5]

A titre indicatif, la Société Axioma a installé un robot ABB 6640 pour le fraisage de produits en composites polyester tels que des vasques et de lavabos. Le choix de ce robot est basé sur son grand rayon d'action (2,8 m) et sur sa capacité de charge permettant de recevoir une broche de fraisage et supporter les efforts de coupe (Figure 1.5)



**Figure 1.5.** Robot ABB pour le fraisage [6]

Les travaux sur les pièces de grandes dimensions montrent un gain significatif lié à l'utilisation de robots sériels comme dans l'exemple précédent. Un autre exemple concerne la fabrication de pièces de grandes dimensions avec un robot de l'entreprise KUKA. La pièce à usiner a une longueur de 1524 cm et une hauteur allant jusqu'à 366 cm. Un robot KUKA KR100 L80 a été monté sur une unité linéaire de 13 m (Figure 1.6).



**Figure 1.6.** Robot Kuka pour l'usinage d'une pièce de grande dimension [5]

Dans les deux cas précédents, une broche a été ajoutée au dernier bras du robot comme effecteur final, cela permet d'intégrer différents outils nécessaires pour les opérations de fabrication à réaliser comme le fraisage, le perçage, la rectification ou le polissage. D'autres robots sériels comme le robot RX170 hsm (high speed machining) de Stäubli ont une conception basée sur les besoins effectifs des opérations d'usinage en termes d'accessibilité. Ainsi, le nombre de degrés de liberté a été réduit à 5 afin d'intégrer une électro-broche dans l'avant-bras et ainsi d'améliorer la rigidité et la précision de pose de l'outil en réduisant le porte-à-faux de l'outil [5] (Figure 1.7).





Figure 1.7. Robot RX170 hsm de Staübli [5]

### 1.3.2) Les robots manipulateurs à architecture parallèle

Selon la référence [7], un manipulateur parallèle est défini comme étant un mécanisme constitué d'un organe terminal à  $n$  degrés de liberté et d'une base fixe, reliés entre eux par au moins deux chaînes cinématiques indépendantes, la motorisation s'effectuant par  $n$  actionneurs simples. Si le nombre de chaînes cinématiques indépendantes est égal au nombre de degrés de liberté alors le manipulateur est dit *pleinement parallèle*, s'il est strictement inférieur il est dit *hybride*. Ainsi, les manipulateurs parallèles se différencient des manipulateurs sériels dont l'architecture consiste en un empilement de segments motorisés. Ainsi, un robot parallèle comprend au moins deux jambes, qui sont des mécanismes séries liés à la base et à l'effecteur (Figure 1.8).

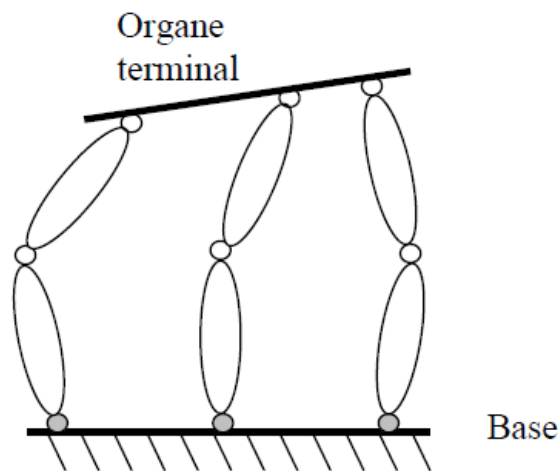


Figure 1.8. Robot Manipulateur à architecture parallèle

Depuis quelques dizaines d'années, les mécanismes parallèles ont une place de choix au sein de la robotique. Leurs propriétés permettent notamment d'effectuer des tâches nécessitant un rapport charge utile-poids élevé, ou encore des dynamiques très importantes [8]. L'exemple le plus célèbre est très certainement la plateforme de Gough-Stewart-Cappel [9]. Il s'agit d'un mécanisme à six jambes montées sur des liaisons universelles (ou joints de Cardan) et terminées par des liaisons rotules. Utilisée dans de nombreuses applications, la plateforme est considérée comme l'archétype du mécanisme parallèle. En mécanique comme en génie civil, il est bien connu que soumettre un matériau aux seules forces de traction et compression en constitue un usage optimal. Une telle conception mène à la

création de composants particulièrement légers et aptes à soulever des charges bien plus pesantes que lesdits composants.

Par ailleurs, on distingue deux classes des robots parallèles, les robots planaires et les robots spatiaux. Un robot planaire (Figure 1.9) peut être constitué d'une plate-forme mobile ayant trois degrés de liberté : deux translations et une rotation autour de la normale au plan de la plate-forme. Si l'on cherche une structure de robot pleinement parallèle, alors le robot doit posséder trois chaînes cinématiques indépendantes motorisées par trois actionneurs  $A_1$ ,  $A_2$  et  $A_3$  (Figure 1.9). Chacune de ces chaînes devant être liées à la fois au sol et à la plate-forme mobile, nous aurons donc trois points d'attache au sol et sur la plate-forme mobile. On peut donc considérer sans perte de généralités une plate-forme triangulaire.

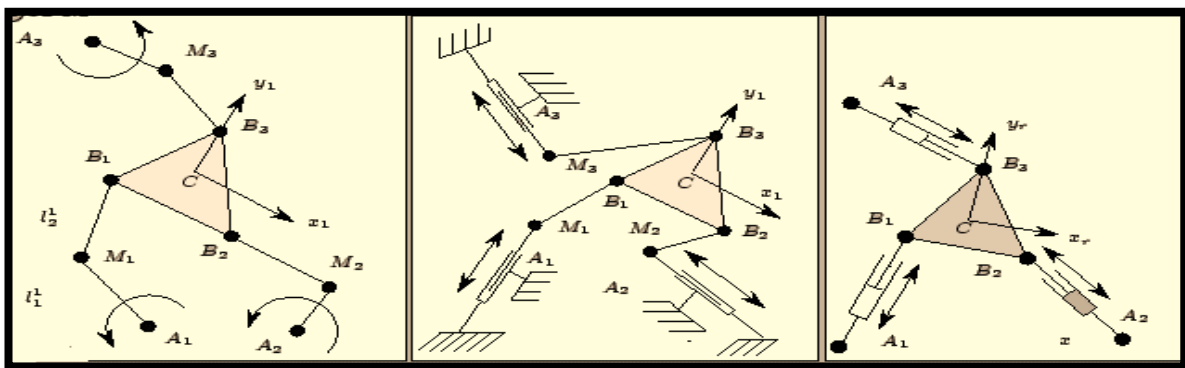


Figure 1.9. Exemples d'architectures de robots planaires [7]

Dans les robots parallèles spatiaux, la plate-forme et la base sont reliées entre elles par des chaînes cinématiques assurant trois ou six degrés de liberté à la plate-forme. Le plus connu des robots à trois degrés de liberté en translation est le robot Delta [10], (Fig. 1.10) et le plus connu pour les robots parallèles à six degrés de liberté est la plate-forme de Gough-Stewart [11], [9] (Fig. 1.11).

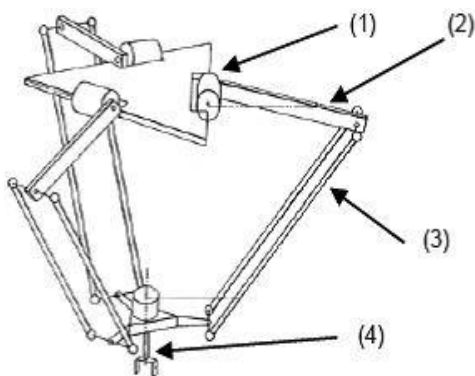
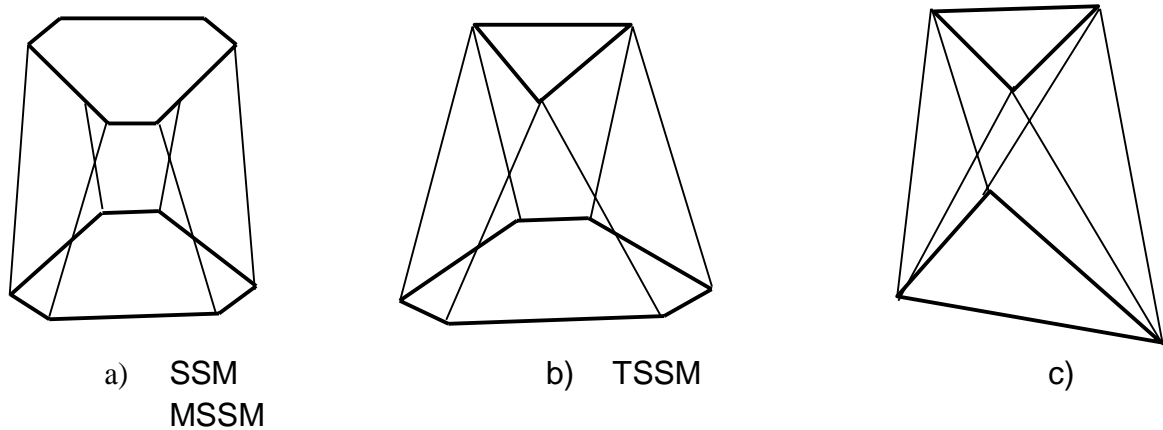


Figure 1.10. Robot Delta à trois degrés de liberté



Figure 1.11. Plate-forme de Gough

Par ailleurs, Merlet [7] décrit trois concepts de structures (SSM, TSSM, MSSM) à six degrés de liberté avec base et une plate-forme reliée entre elles par six actionneurs linéaires comme indiqué sur la (Figure 1.12)



**Figure 1.12.** Robots parallèles de type SSM, TSSM et MSSM

La machine Variax [12], construite en 1994 sur la base d'un hexapode, est la première machine-outil à structure parallèle (Figure 1.13). Avant l'abandon du projet, un unique exemplaire de cette machine a été commercialisé à l'Université de Nottingham. Depuis cette date, de nombreux prototypes ont été développés, mais très peu de machines sont actuellement utilisées dans le milieu industriel. Lorsque cela est le cas, ces machines réalisent principalement des opérations de perçage lamage-taraudage, ne nécessitant que trois degrés de liberté [13]. Parmi les principales raisons de cette désaffection, il faut citer [14] :

- la méfiance des utilisateurs par rapport à ces nouvelles architectures de machines-outils qui bouleverse complètement les usages ;
- un ratio « empreinte au sol / volume de travail », en défaveur des architectures parallèles ;
- des procédures d'étalonnage plus complexe ;
- la présence d'articulations passives à plusieurs degrés de liberté ;
- des capacités d'orientation limitées à cause des singularités parallèles [18] ;
- la présence de moteurs qui entraîne la production d'énergie thermique dans les corps en mouvement occasionnant des problèmes de dilatation.



**Figure 1.13.** Machine-outil Variax (Gidding & Lewis)

Néanmoins, une étude comparative entre des machines-outils, respectivement à structure parallèle et à structure sérielle [16], a montré que les machines-outils à structure parallèle ont de réelles aptitudes pour le fraisage de formes complexes, ainsi que pour le fraisage à grande vitesse. Les résultats obtenus par certaines machines-outils à structure parallèle sont d'un niveau de qualité équivalent à ceux obtenus par des machines-outils à structure sérielle. Dans la référence [17] une étude comparative à caractère général est réalisée entre deux machines-outils fondées pour l'une, sur une structure hexapode dont les jambes ont une longueur variable, et pour l'autre sur un hexapode dont les jambes ont une longueur fixe. Cette étude montre qu'en termes de rigidité la structure possédant des jambes de longueur fixe présente un meilleur potentiel. Mais malgré tout, la rigidité d'une telle structure varie fortement dans l'espace de travail. L'auteur conclut, que sur la base d'une conception très rigoureuse, une machine-outil de ce type peut avoir, pour certaines applications spécifiques, des caractéristiques identiques aux machines-outils à structure sérielle.

### 1.3.3) Les robots manipulateurs à architecture hybride

Une tendance actuelle dans le monde de la machine-outil est de proposer des machines à structure hybride permettant de cumuler les avantages associés à chaque type de structure. En fait, la définition exacte d'un mécanisme hybride est évasive. Dans cette classification, nous considérerons comme hybride un mécanisme qui comporte plusieurs chaînes cinématiques reliant le bâti à l'organe terminal et dont une des chaînes au moins comporte plus d'un actionneur.

La machine Verne (Figure 1.14) de la société Fatronik [18] utilise une table porte-outil sérielle procédant deux axes de rotations et une structure porte outil parallèle à trois degrés de liberté. Dans ce cas de figure, la structure sérielle est utilisée afin d'augmenter les capacités d'orientation relative pièce/outil.

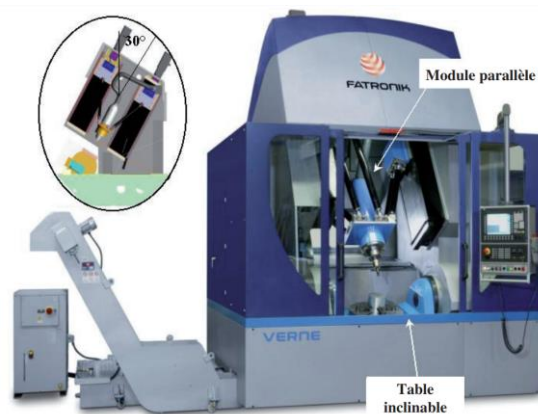


Figure 1.14. Vue d'ensemble de la MOCN VERNE

La combinaison du module sériel et du module parallèle permet à la MOCN VERNE de disposer de capacités cinématiques élevées et homogènes, à savoir une vitesse maximale de 50 m/min, une accélération par axe de 15 m/s<sup>2</sup> par axe ainsi que la possibilité de travailler avec des valeurs de Jerk supérieures à 100 m/s<sup>3</sup> sur chaque axe de translation. La broche installée dans le module parallèle présente une puissance nominale de 40 kW et une vitesse

de rotation maximale de 24000 tr/min. Parallèlement à cela, l'espace de travail est un cube de 600 mm de côté. Par conséquent, cette machine-outil est tout à fait apte à réaliser des applications d'Usinage à Grande Vitesse.

Dans le même ordre d'idée, l'Exechon concept [19] [20], est une nouvelle structure porte-outil à 5 degrés de liberté conçue à partir d'une structure parallèle possédant 3 degrés de libertés (2 rotations et 1 translation) à laquelle est associé un poignet série qui possède 2 degrés de libertés en rotation (Figure 1.15). Cette nouvelle structure est la remplaçante de la machine Tricept de Neos Robotics [21] qui représente à ce jour 70 % du marché des machines-outils à structure parallèle. L'avantage principal du l'Exechon est de ne posséder que 6 liaisons passives à l'origine de 10 degrés de liberté inactifs alors que la structure de la Tricept possédait 7 liaisons passives et 18 degrés de liberté inactifs. La limitation du nombre des liaisons passives permet de réduire les jeux dans les liaisons, d'augmenter la rigidité de la structure, d'en simplifier la conception et donc d'en réduire le coût [19].

Le robot Tricept de Neos (Figure 1.16) est une machine hybride : un mécanisme porteur d'architecture parallèle porte un poignet série. Une particularité de ce robot est la présence d'une patte passive (non-actionnée) qui contraint la cinématique du robot. Ce robot a connu un réel succès commercial mais il semble que dans le domaine de la machine-outil les performances obtenues soient en deçà de celles espérées.

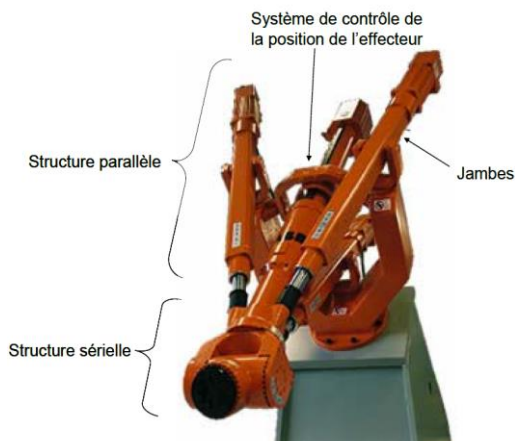


Figure 1.15. L'Exechon concept

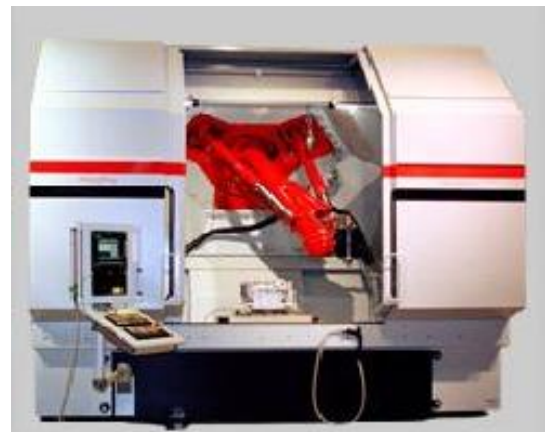


Figure 1.16. Tricept 845 (Neos Robotics)

#### 1.4) Conclusion

Dans ce chapitre, nous avons présenté un bref aperçu sur l'évolution de l'emploi des robots dans le milieu industriel. Ensuite, nous avons donné quelques définitions utiles à l'état de l'art des robots utilisés en usinage, tout en citant quelques exemples avec différentes architectures. Les notions introduites dans ce chapitre vont servir de base pour le développement des futurs chapitres. Dans le prochain chapitre, nous aborderons la modélisation analytique d'un robot parallèle plan à deux degrés de liberté qui va servir de base pour la construction de notre mini-CNC à architecture parallèle à trois degrés de liberté.

## *Chapitre II :*

# *Modélisation d'un robot parallèle plan a deux degrés liberté*

## 2.1) Introduction

Les travaux de Gough (1956) et Stewart (1965) ont permis d'établir les principes de base pour concevoir les manipulateurs à cinématique parallèle qui ont une structure cinématique en boucle fermée, ce manipulateur peut générer une position et une orientation spécifiées d'une plate-forme mobile dans l'espace [9] [22]. Depuis lors, les manipulateurs parallèles ont été largement étudiés par de nombreux chercheurs et plusieurs architectures, ayant différents degrés de liberté (*ddl*) ont vu le jour.

Les manipulateurs parallèles les plus étudiés sont ceux à 6 *ddl*. Ces manipulateurs parallèles possèdent les avantages d'une rigidité élevée, d'une faible inertie et d'une grande capacité de charge utile. Cependant, ils souffrent des problèmes d'un espace de travail utile relativement petit et des difficultés de conception et de singularité [23]. De plus, leur cinématique directe pose un problème très difficile lors de la résolution. Par ailleurs, pour les manipulateurs parallèles avec deux ou trois *ddl*, ces problèmes peuvent être décrits sous forme fermée et simple à traiter [24]. D'une manière générale, les singularités des manipulateurs parallèles à 6-*ddl* ne peuvent pas être trouvées facilement. Mais pour un manipulateur parallèle à deux ou trois *ddl*, les singularités sont toujours facilement identifiables. Pour ces raisons, les manipulateurs parallèles avec moins de six *ddl*, en particulier deux ou trois *ddl*, ont de plus en plus attiré l'attention des chercheurs.

En ce qui concerne les applications industrielles des robots ayant moins de 6 *ddl*, on peut citer les manipulateurs parallèles avec 3*ddl* de translation qui ont joué un rôle important dans les applications industrielles [25-26]. Par exemple le manipulateur de Tsai (1996), dans lequel chacune des trois jambes se compose d'un parallélogramme, est la première conception pour résoudre le problème de la chaîne cinématique cardan-cardan (UU). Un manipulateur parallèle 3-*ddl* de translation baptisé, *Star*, a été conçu par Hervé (1992) sur la base de la théorie des groupes. De tels manipulateurs parallèles pouvant avoir de larges applications dans le monde industriel, par exemple, les applications pick-and-place, machines d'usinage et dispositifs médicaux spéciaux [25-26].

En outre, des manipulateurs parallèles planaires à 2-*ddl* existent [28-29], particulièrement on a les mécanismes à cinq barres bien connu avec des actionneurs prismatiques ou des actionneurs à révolution. Dans le cas d'un manipulateur à actionneurs de révolution, le mécanisme se compose de cinq liaisons de révolution et les deux liaisons de la base sont actionnées. Dans le cas du manipulateur à actionneurs prismatiques, le mécanisme se compose de trois liaisons rotatives et de deux liaisons prismatiques actionnées. La fonction de sortie du manipulateur est le mouvement de translation d'un point sur l'effecteur terminal. Cela veut dire que l'orientation de l'effecteur terminal change en conséquence du mouvement. Malheureusement, il y a peu de manipulateurs parallèles de translation à 2-*ddl* pour lesquels l'effecteur final se déplace avec une orientation constante. Un tel manipulateur peut être utilisé dans la conception d'une machine-outil ayant une structure portique au lieu d'une structure traditionnelle avec des chaînes en série pour améliorer sa rigidité et ses caractéristiques d'inertie [28-29].

Dans ce chapitre, un manipulateur parallèle de la famille des robots biglide ayant deux *ddl* de translation est étudié. La cinématique de ce manipulateur se base sur celle des

mécanismes de type parallélogramme qui est adopté dans chacun des deux jambes du robot. Le mouvement de la plate-forme est un mouvement de translation planaire d'un corps rigide à orientation constante. La modélisation analytique de ce robot est abordée en détail.

## 2.2) Description du manipulateur parallèle 2-DoF

Les manipulateurs parallèles à deux *ddl* sont des systèmes très importants dans la famille des manipulateurs parallèles. Les manipulateurs planaires parallèles à *2ddl* existants ne peuvent positionner qu'un point et non un corps rigide dans un plan [29].

L'architecture du manipulateur parallèle à *2-ddl* étudié dans le cadre de ce projet est illustrée par la (figure 2.1). Le schéma cinématique du manipulateur est donné dans la (figure 2.1), où la base est étiquetée 1 et la plate-forme mobile est étiquetée 2. La plate-forme mobile est connectée au bâti par deux pieds identiques. Chaque jambe est constituée d'un parallélogramme parallèle planaire à quatre barres : corps 2, 3, 4 et 5 pour la première jambe ; corps 2, 6, 7 et 8 pour la deuxième jambe. La chaîne cinématique du manipulateur peut être notée 2-Pa, où "Pa" représente un parallélogramme plan à quatre barres avec quatre liaisons pivot. Les biellettes sont respectivement actionnées par des actionneurs prismatiques. En conséquence de cet agencement cinématique, les mouvements de la plate-forme mobile sont obtenus par la combinaison des mouvements des biellettes virtuelles 3 et 8 pouvant être transmis à la plate-forme par le système des deux parallélogrammes. Du fait de la structure, on peut voir que la plate-forme mobile ou le corps rigide 2 a deux degrés de liberté purement en translation par rapport à la base en raison des Parallélogrammes à barres utilisés. Ainsi, le manipulateur peut positionner un corps rigide dans un espace 2D avec une orientation constante.

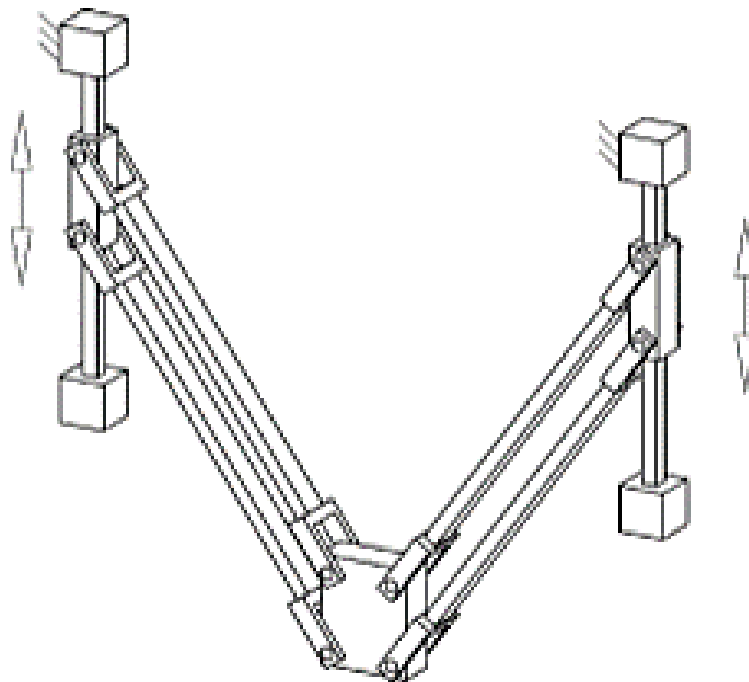


Figure 2.1. Architecture du manipulateur parallèle planaire 2-ddl étudié



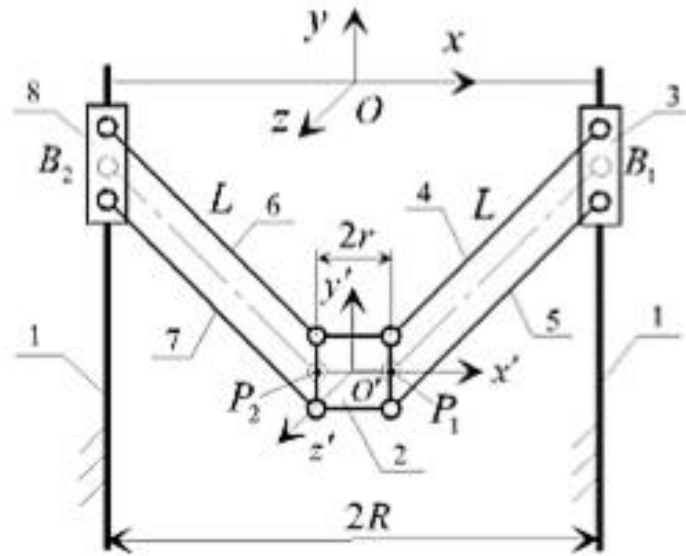


Figure 2.2. Schéma cinématique du manipulateur et paramétrage [30]

Ce type d'architecture peut être appliquée à la manipulation des pièces et utilisée comme base mobile pour un manipulateur spatial. En outre, un manipulateur ayant une cinématique similaire à celle de la figure 2.2 peut être utilisé pour réaliser une machine-outil avec une structure de portique où les deux degrés de liberté de translation planaires sont généralement nécessaires. Donc, sur la base de ce type manipulateur, on peut envisager :

- une machine-outil à **cinq axes**, où le troisième *ddl* de translation est attaché à la table de travail et deux *ddl* d'orientation sont fournis par un poignet série à deux axes.
- une machine-outil à **trois axes**, où le troisième *ddl* de translation est attaché à la table de travail et l'outil est directement attaché à la nacelle.

C'est ce dernier cas qui nous intéresse dans le contexte de notre projet de fin d'études.

### 2.3) Modélisation géométrique du robot

Pour mener à bien cette analyse du robot, on se basera sur les paramètres de la figure 2.2. Donc, soit le référentiel  $\mathcal{R}(O, xyz)$  qui est fixé à la base et un repère mobile  $\mathcal{R}'(O', x'y'z')$  qui est attaché à la plate-forme mobile, où  $O'$  est le point de référence sur la plate-forme mobile. Pour la caractéristique d'un plan parallélogramme à quatre barres, on peut considérer les chaînes fictives  $P_1B_1$  et  $P_2B_2$ , comme indiqué sur la figure 2.2.

Pour résoudre la cinématique du manipulateur. Les vecteurs  $P_{i\mathcal{R}}$  et  $P_{i\mathcal{R}'}$  ( $i = 1, 2$ ) seront définis comme étant les vecteurs positions des points  $P_i$  dans les repères  $\mathcal{R}$  et  $\mathcal{R}'$  respectivement. Aussi, les vecteurs  $b_{i\mathcal{R}}$  ( $i = 1, 2$ ) sont définis comme étant les vecteurs positions des points  $B_i$  dans le repère  $\mathcal{R}$ .

Les paramètres géométriques du manipulateur sont  $P_iB_i = L$  ( $i = 1, 2$ ),  $P_1P_2 = 2r$ , et la distance entre deux rails  $2R$ . La position du point  $O'$  dans le repère fixe est notée vecteur  $c_{\mathcal{R}}$  suivant :

$$c_{\mathcal{R}} = (x, y)^T. \quad (2.1)$$

Les vecteurs de  $\mathbf{b}_{iR}$  dans le cadre fixe peuvent être écrit comme suit :

$$\mathbf{b}_{1R} = (R \quad y_1)^T, \quad \mathbf{b}_{2R} = (-R \quad y_2)^T, \quad (2.2)$$

Où  $y_i$  sont les entrées actionnées pour les deux glissières. Les vecteurs  $\mathbf{P}_{iR}$  dans le cadre fixe  $\mathfrak{R}$  peut être écrit comme :

$$\mathbf{P}_{iR} = \mathbf{P}_{iR'} + \mathbf{c}_R, \quad i = 1, 2, \quad (2.3)$$

$$\text{avec } \mathbf{P}_{iR'} = (\mathbf{r} \quad \mathbf{0})^T, \quad \mathbf{P}_{iR''} = (-\mathbf{r} \quad \mathbf{0})^T. \quad (2.4)$$

Le modèle géométrique inverse du manipulateur peut être résolu en écrivant l'équation de contrainte suivante :

$$\|\mathbf{P}_{iR} - \mathbf{b}_{iR}\| = L, \quad i = 1, 2, \quad (2.5)$$

En conséquence :

$$y_1 = \pm \sqrt{(L^2) - (\mathbf{r} + \mathbf{x} - R)^2 + \mathbf{y}} \quad , \quad (2.6)$$

$$y_2 = \pm \sqrt{(L^2) - (\mathbf{r} - \mathbf{x} + R)^2 + \mathbf{y}} \quad , \quad (2.7)$$

À partir des relations (2.6) et (2.7) nous pouvons voir qu'il existe quatre solutions pour le modèle inverse du manipulateur. Par conséquent, pour un manipulateur donné et pour les valeurs prescrites de la position de la plate-forme mobile, les entrées actionnées requises peuvent être directement calculées à partir des équations (2.6) et (2.7). Pour obtenir la configuration illustrée à la figure 2.1, le signe «  $\pm$  » dans les équations (2.6) et (2.7) doit être «  $+$  ».

L'objectif de la résolution du modèle géométrique directe est de définir une fonction mathématique à partir de l'ensemble connu des entrées actionnées (variables articulaires) jusqu'à la pose inconnue de la plate-forme dans l'espace opérationnelle. À partir des équations (2.6) et (2.7), le modèle géométrique direct du manipulateur peut être décrit comme suit : [30]

$$\mathbf{x} = \mathbf{a}\mathbf{y} + \mathbf{b}, \quad (2.8)$$

Où :

$$\mathbf{a} = \frac{y_1 - y_2}{2(R - r)}, \quad \mathbf{b} = \frac{y_1^2 - y_2^2}{4(R - r)} \quad (2.9)$$

Et

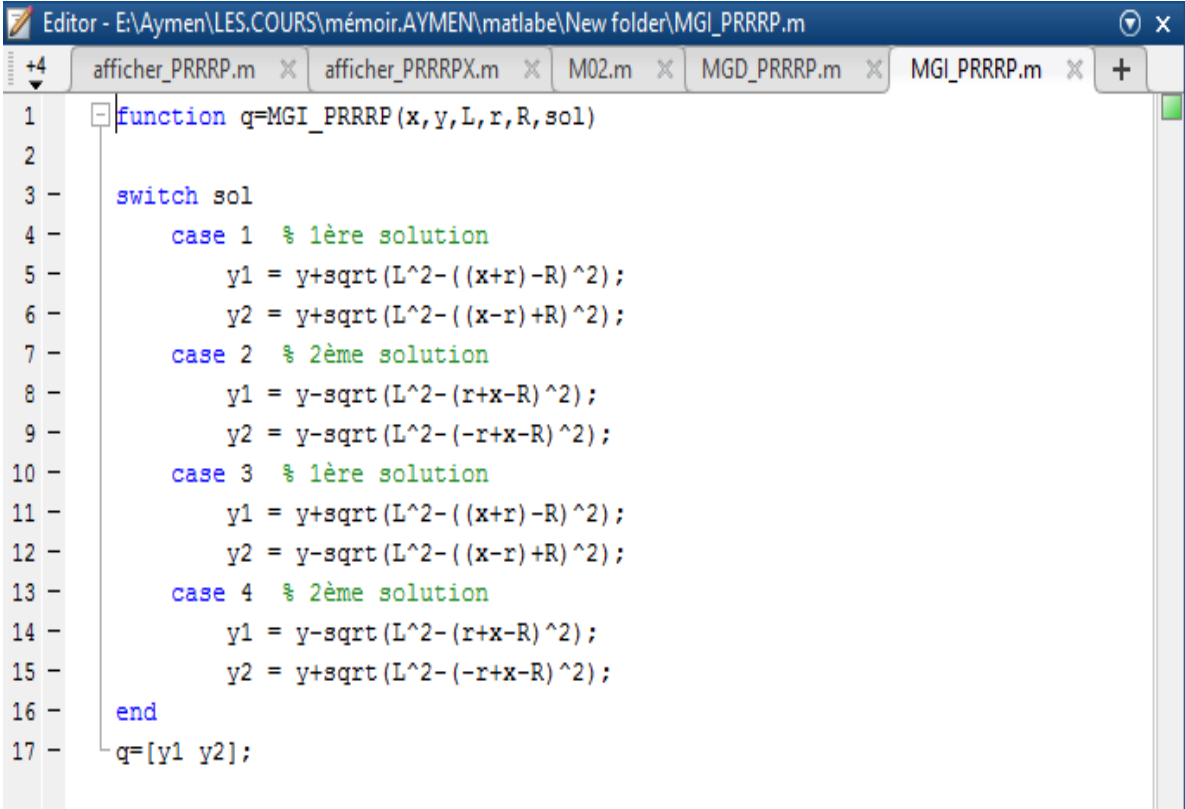
$$\mathbf{y} = \frac{-f \pm \sqrt{f^2 - 4eg}}{2e} \quad (2.10)$$

Avec :

$$\mathbf{e} = \mathbf{a}^2 + 1, \quad \mathbf{f} = 2\mathbf{a}(\mathbf{r} + \mathbf{b} - R) - 2y_1, \quad \mathbf{g} = (\mathbf{r} + \mathbf{b} - R)^2 + y_1^2 - L^2 \quad (2.11)$$

Il existe deux solutions pour le modèle géométrique directe pour le manipulateur. Pour obtenir la configuration directe illustrée à la figure 2.1, le signe «  $\pm$  » dans l'équation (2.10) doit être «  $-$  ».

À partir des équations ci-dessus, nous pouvons voir que les problèmes de géométrie inverse et directe du manipulateur sont très faciles et peuvent être décrits comme des formes fermées. Par ailleurs, nous avons implémenté les relations précédentes sous Matlab pour simuler le fonctionnement de notre robot. Les figures 2.3.-2.4 sont des captures d'écran pour les programmes réalisés.



```
1 function q=MGI_PRRRP(x,y,L,r,R,sol)
2
3 switch sol
4 case 1 % 1ère solution
5     y1 = y+sqrt(L^2-((x+r)-R)^2);
6     y2 = y+sqrt(L^2-((x-r)+R)^2);
7 case 2 % 2ème solution
8     y1 = y-sqrt(L^2-(r+x-R)^2);
9     y2 = y-sqrt(L^2-(-r+x-R)^2);
10 case 3 % 1ère solution
11     y1 = y+sqrt(L^2-((x+r)-R)^2);
12     y2 = y-sqrt(L^2-((x-r)+R)^2);
13 case 4 % 2ème solution
14     y1 = y-sqrt(L^2-(r+x-R)^2);
15     y2 = y+sqrt(L^2-(-r+x-R)^2);
16 end
17 q=[y1 y2];
```

Figure 2.3. Fonction Matlab implémentant le modèle géométrique inverse

```

1  function X = MGD_PRRRP(y1, y2, L, r, R, sol)
2
3  -   a = 0.5*(y2-y1)/(R-r);
4  -   b = 0.25*(y1^2-y2^2)/(R-r);
5
6  -   ep = a^2+1;
7  -   f = 2*a*(r+b-R)-2*y1;
8  -   g = (r+b-R)^2+y1^2-L^2;
9
10 -   switch sol
11 -       case 1
12 -           y = (-f-sqrt(f^2-4*ep*g))/(2*ep);
13 -       case 2
14 -           y = (-f+sqrt(f^2-4*ep*g))/(2*ep);
15 -   end
16
17 -   x = a*y+b;
18 -   X = [x y];
    
```

Figure 2.4 Fonction Matlab implémentant le modèle géométrique directe

## 2.4) Modélisation cinématique du robot

### 2.4.1) Modélisation cinématique indirecte (MCI)

L'équation (2.5) peut être dérivée par rapport au temps pour obtenir les équations de vitesse. Cela conduit à une équation de la forme :

$$\mathbf{A} \begin{pmatrix} \dot{y}_1 \\ \dot{y}_2 \end{pmatrix} = \mathbf{B} \begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix}, \quad (2.12)$$

Où  $\mathbf{A}$  et  $\mathbf{B}$  sont, respectivement, les matrices Jacobéennes inverse et directe  $2 \times 2$  du manipulateur et peuvent être exprimées comme :

$$\mathbf{A} = \begin{bmatrix} y - y_1 & \mathbf{0} \\ \mathbf{0} & y - y_2 \end{bmatrix}; \mathbf{B} = \begin{bmatrix} r + x - R & y - y_1 \\ x - r + R & y - y_2 \end{bmatrix}. \quad (2.13)$$

Avec :  $\det(\mathbf{A}) \neq 0$

Si la matrice  $\mathbf{A}$  est non singulière la matrice Jacobéenne du manipulateur peut être obtenue comme suit:

$$\mathbf{J} = \mathbf{A}^{-1} \cdot \mathbf{B} = \begin{bmatrix} (r + x - R)/(y - y_1) & \mathbf{1} \\ (x - r + R)/(y - y_2) & \mathbf{1} \end{bmatrix}. \quad (2.14)$$

### 2.4.2) Modélisation cinématique directe (MCD)

Ce modèle peut être établi à partir de la relation (2.13), si la matrice  $\mathbf{B}$  est non singulière alors, la matrice Jacobéenne du manipulateur peut être obtenue comme suit :

$$\mathbf{J} = \mathbf{B}^{-1} \cdot \mathbf{A} \quad (2.15)$$

donc 
$$\mathbf{J} = \begin{bmatrix} (r + x - R)/(y - y_1) & \mathbf{1} \\ (x - r + R)/(y - y_2) & \mathbf{1} \end{bmatrix}. \quad (2.16)$$

Soit l'équation aux valeurs propres suivante :

$$|\mathbf{I}\lambda - \mathbf{J}| = 0$$

Dans laquelle on a la matrice identité  $\mathbf{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$  et donc on aboutit à :

$$\lambda^2 + s\lambda + t = 0 \quad (2.17)$$

Avec :

$$s = \left( \frac{r+x-R}{y-y_1} + 1 \right), \quad t = \frac{r+x-R}{y-y_1} - \frac{x-r+R}{y-y_2} \quad (2.18)$$

Ensuite, les valeurs singulières de la matrice Jacobéenne  $\mathbf{J}$  peuvent être obtenues comme suit :

$$\lambda = \frac{-s \pm \sqrt{s^2 + 4t}}{2} \quad (2.19)$$

A ce stade, on peut parler de la dextérité du manipulateur parallèle qui peut être évaluée en utilisant l'indice de conditionnement  $1/k$  qui est l'inverse du nombre de conditionnement  $k$  [31] [32] qui s'écrit :

$$1/k = \lambda_1 / \lambda_2 \quad (2.20)$$

Où  $\lambda_1$  et  $\lambda_2$  sont les valeurs singulières maximale et minimale la matrice Jacobéenne  $\mathbf{J}$ , qui peuvent être obtenues à partir de l'équation (2.18), du manipulateur, respectivement. Par ailleurs, l'indice de conditionnement global correspondant sera :

$$\eta = \int_w 1/k \, dw / \int_w dw \quad (2.21)$$

Où  $W$  est l'espace de travail accessible du manipulateur.

Dans les indices de conditionnement, les configurations singulières sont à éviter. À partir de l'analyse précédente, on peut voir que le manipulateur est d'une cinématique simple et l'analyse de la singularité sera également simple, ce qui peut être atteint à partir des matrices  $\mathbf{A}$  et  $\mathbf{B}$  [33] :

- Quand  $|\mathbf{A}| = 0$  et  $|\mathbf{B}| \neq 0$ , d'après l'équation (2.13), il y a  $\mathbf{y} = \mathbf{y}_1$  ou  $\mathbf{y} = \mathbf{y}_2$ , ce qui signifie que la première ou la deuxième jambe est parallèle à l'axe x. Cela correspond au premier type de singularité.
- Si  $|\mathbf{B}| = 0$  et  $|\mathbf{A}| \neq 0$ , le deuxième type de singularité se produit, c'est-à-dire  $\mathbf{r} + \mathbf{x} = \mathbf{R}$  pour la première jambe lorsque x est positif et  $\mathbf{R} + \mathbf{x} = \mathbf{r}$  pour la deuxième jambe lorsque x est négatif. Dans un tel cas, le manipulateur est dans la configuration où quatre barres du parallélogramme dans l'une des deux branches sont parallèles entre elles.
- Si  $|\mathbf{B}| = 0$  et  $|\mathbf{A}| = 0$  conduisent au troisième type de singularité, dans lequel les deux jambes sont parallèles à l'axe x. Cette condition de singularité est réalisée quand  $\mathbf{r} + \mathbf{L} = \mathbf{R}$ .

À partir des équations précédentes, nous avons réalisé sous Matlab les fonctions suivantes qui nous permettent de réaliser l'analyse cinématique de notre robot :

```

1      %%%%%%%%% (MGD) %%%%%%%%%
2      function Ja=JACD(x,y,r,R,y1,y2)
3
4      A = [y-y1 0;0 y-y2];
5      B = [r+x-R y-y1;x-r+R y-y2];
6      if det(A)==0
7          disp('configuration singulière')
8          Ja=[0 0;0 0];
9      else
10         Ja = inv(A)*B;
11     end
12 end
    
```

Figure 2.5 : Fonction Matlab implémentant la Jacobienne du modèle MCD

```

1      %%%%%%%%% (MGI) %%%%%%%%%
2      function Ja=JACI(x,y,r,R,y1,y2)
3
4      A = [y-y1 0;0 y-y2];
5      B = [r+x-R y-y1;x-r+R y-y2];
6      if det(B)==0
7          disp('configuration singulière')
8          Ja=[0 0;0 0];
9      else
10         Ja = inv(B)*A;
11     end
12 end
    
```

Figure 2.6. Fonction Matlab implémentant la Jacobienne du modèle MCI

La figure 2.7 illustre un programme Matlab faisant appel aux différentes fonctions développées pour modéliser la géométrie et la cinématique de notre robot plan biglide.

```

Editor - E:\Aymen\LES.COURS\memoir.AYMEN\matlab\New folder\Model2_PRRRP.m
JACD.m x JACI.m x Model2_PRRRP.m x +
4 - H = 220; % Hauteur de la machine
5 - h = 30.5; % 1/2 Hauteur de la nacelle
6 - R = 146; % 1/2 largeur de la machine
7 - r = 41; % 1/2 largeur de la nacelle
8 - L = 185; % longueur des bras L>R
9 % variables opérationnelles
10 - x = 60 % -0.2*R
11 - y = -190 % -1.5*H
12 - plot(x,y)
13 % variables articulaires MGI
14 - sol = 1 % (on utilisera sol =1)
15 - q = MGI_PRRRP(x,y,L,r,R,sol)
16 - y1 = q(1);
17 - y2 = q(2);
18 % variables opérationnelles MGD
19 - sol = 1 % (on utilisera sol =1)
20 - X = MGD_PRRRP(y1,y2,L,r,R,sol)
21 - afficher_PRRRP(y1,y2,x,y,r,R,h,H)
22 % modekisation cinématique
23 - vitx = 0.3; % composante de la vitesse opérationnelle selon x
24 - vity = -0.25; % composante de la vitesse opérationnelle selon y
25 % inverse
26 - jac2 = JACI(x,y,r,R,y1,y2)
27 - vitq = jac2*[vitx;vity] % vitesses articulaires
28 % directe
29 - jac1 = JACD(x,y,r,R,y1,y2)
30 - vitxy = jac1*vitq; % vitesses articulaires

```

Figure 2.7. Programme Matlab utilisant les différentes fonctions Modélisant notre robot

### 2.5) Espace de travail du manipulateur

L'un des problèmes les plus importants dans le processus de conception d'un manipulateur parallèle est son espace de travail. Pour les manipulateurs parallèles, ce problème peut être plus critique car les manipulateurs parallèles auront parfois un espace de travail plutôt limité. L'espace de travail du robot parallèle planaire 2-ddl est souvent représenté comme une région dans le plan. La détermination de l'espace de travail est relativement simple, et peut être obtenue géométriquement à partir de l'équation cinématique inverse. A partir de l'équation (2.5), on obtient :

$$(r + x + R)^2 + (y - y_1)^2 = L^2 \tag{2.22}$$

$$(x - r + R)^2 + (y - y_2)^2 = L^2 \tag{2.23}$$

Ce qui signifie que si  $y_i$  est spécifié, les équations (2.22) et (2.23) représentent deux cercles centrés en  $B_1$  et  $B_2$ , respectivement (Figure 2.8). Leurs rayons sont  $L$ . Si  $y_i \in [y_{i \min}, y_{i \max}]$ , les équations (2.22) et (2.23) représentent deux surfaces enveloppantes dont chacune est le lieu d'un cercle (le rayon est  $L$ ), lorsque le centre roule sur les segments de droite ayant comme abscisse  $x = R - r$  et  $x = r - R$  (pour  $y_i \in [y_{i \min}, y_{i \max}]$ ), respectivement. L'intersection des deux surfaces enveloppantes est l'espace de travail du manipulateur.

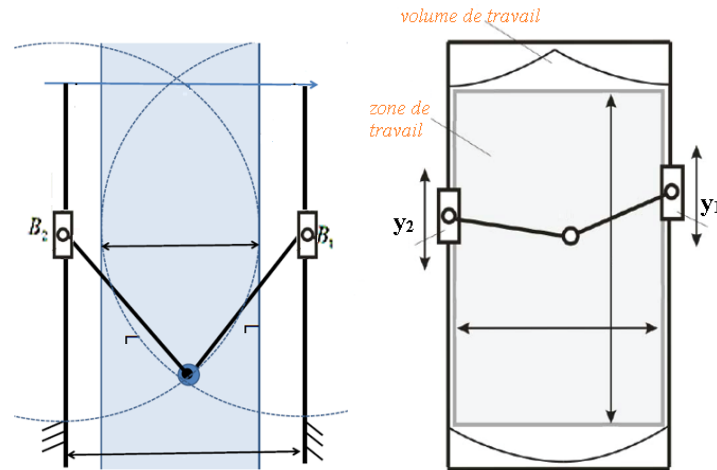


Figure 2.8 : Limites du volume de travail du robot

## 2.6) Conclusion

Les manipulateurs cinématiques parallèles avec moins de 6 ddl sont devenues récemment très attractifs de point de vue applications industrielles. Dans ce chapitre, un manipulateur translationnel 2-ddl (biglide) a été étudié. Un avantage du manipulateur est qu'il peut positionner un corps rigide dans un plan 2D tout en conservant une orientation constante. Les problèmes de cinématique inverse et directe, l'espace de travail, les indices de conditionnement, la singularité sont présentés dans ce chapitre. Le manipulateur proposé peut être appliqué à la manipulation de pièces, utilisé comme base mobile pour un manipulateur spatial, et utilisé dans le domaine des machines-outils. C'est ce cas de figure qui nous intéresse dans la suite de ce travail.



## ***Chapitre III :***

# ***conception de la machine***

### 3.1) Introduction

Notre projet de fin d'étude a pour objectif la mise en œuvre d'une mini machine-outil 3 axes à architecture parallèle à commande numérique. Dans ce chapitre, nous allons traiter la conception de la machine (du moins les parties les plus importantes) et on détaillera donc le processus de conception. Pour cela, on traitera les points suivants :

- L'élaboration du cahier des charges
- Le choix des solutions technologiques
- Le dimensionnement des principaux organes.

En réalité, l'un des facteurs déterminant dans une démarche rationnelle de conception de produit est l'établissement d'un cahier des charges fonctionnel CdCF. Le CdCF est en fait le document par lequel le demandeur exprime son besoin en termes de fonctions de service et de contraintes. C'est un outil méthodologique nécessaire pour détecter et formuler fonctionnellement le besoin. La rédaction du CdCF comporte deux étapes : Analyse du besoin et Etude de faisabilité.

### 3.2) Analyse du besoin

L'opération d'analyse du besoin consiste à saisir, énoncer et valider le besoin. En général, c'est le service « marketing » qui réalise une étude du marché pour identifier le besoin qui sera par la suite énoncé en utilisant par exemple un diagramme Bête à cornes. Le contrôle de validité permet de valider le besoin exprimé. Pour notre étude, nous avons identifié les besoins potentiels, ils sont résumés dans le schéma de la figure 3.1. Ce sont ces besoins qui font que ce projet ait un sens et peut exister.

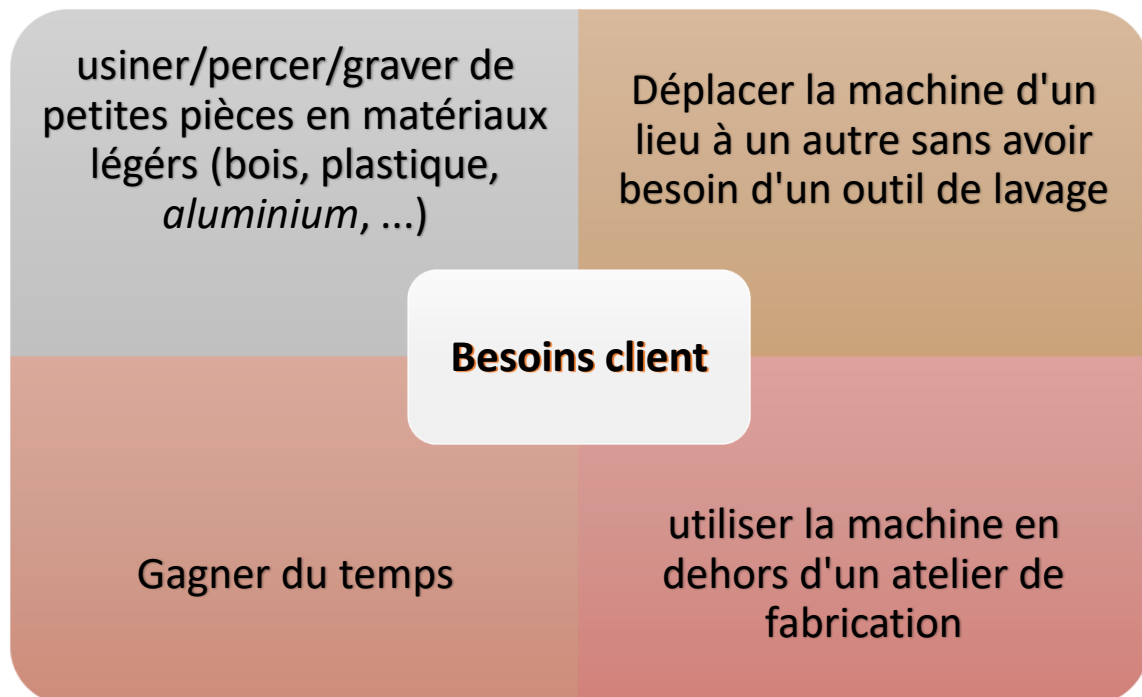


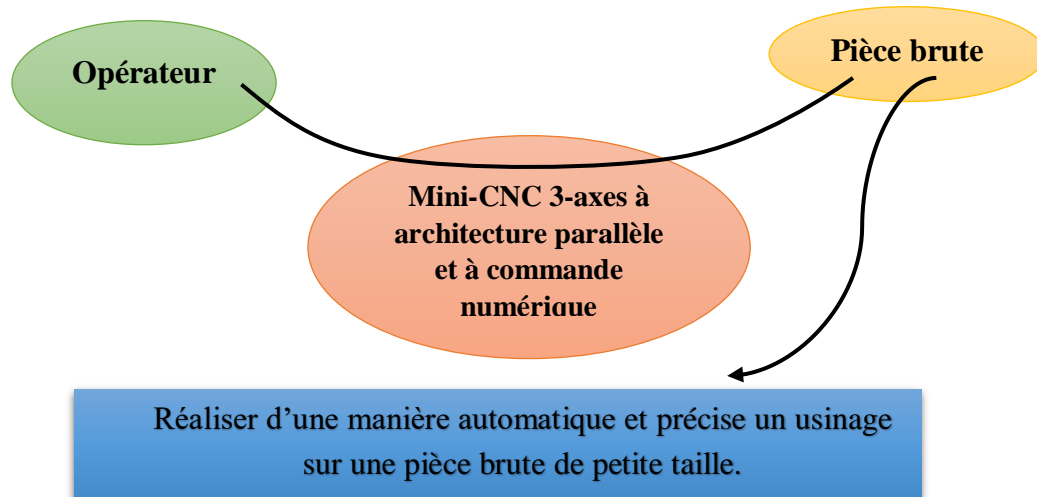
Figure 3.1. Les besoins client identifiés

Par ailleurs, énoncer le besoin consiste à répondre aux questions regroupées dans le tableau 3.1.

**Tableau 3.1** : tableur questions/réponses relatif au besoin

Questions	Réponses
1) <i>A qui le produit rend-il service ?</i>	Opérateur
2) <i>Sur quoi le produit agit-il ?</i>	Pièce brute
3) <i>Dans quel but ?</i>	Pièce Usinée

Le diagramme « bête à cornes » permet de formaliser graphiquement les réponses précédentes comme illustré sur la figure 3.2. Cet outil se situe dans la première étape de la méthode d'analyse fonctionnelle et d'analyse de la valeur (APTE). Il a pour objectif de représenter graphiquement l'expression du besoin du client (ou utilisateur) à travers les 3 questions simples du tableau 3.1.



**Figure 3.2.** Diagramme bête à corne

Pour valider l'expression du besoin, il faut se poser les questions complémentaires suivantes :

**Pourquoi le système existe-t-il ?**

- Usinage avec précision des pièces prismatiques de forme complexe qu'on ne peut pas réaliser sur des machines conventionnelles.
- Réalisation de la gravure sur des surfaces planes pour des objets décoratifs.
- Réalisation de gabarit en bois pour le moulage en sable.
- Opérations de sculptage sur des pièces en bois pour la menuiserie.

### Qu'est-ce qui pourrait faire disparaître ou évoluer le besoin ?

Cette question permet de valider la stabilité du besoin, elle permet aussi d'anticiper les évolutions du besoin.

- Une forte demande du marché dans le domaine de la gravure et la décoration.
- Un coût réduit par rapport à une machine 4 ou 5 axes.
- Un investissement accessible aux artisans (bijoutiers, menuisiers, plâtriers, etc.) et aux professionnels (fabrication mécanique, moulage, etc.).

### 3.3) Etude de la faisabilité

A partir du besoin validé précédemment, on va étudier la faisabilité du projet en passant par les étapes suivantes :

- Recenser les éléments de l'environnement du produit ciblé. On doit donc définir la frontière de l'étude avec exactitude et dresser la liste des éléments de l'environnement en interaction directe avec le produit dans différentes situations de vie.
- Identifier les fonctions de service en utilisant le graphe d'interaction pieuvre. Ainsi, on doit mettre les éléments de l'environnement en relation avec le produit en recherchant les liaisons entre : i) plusieurs éléments de l'environnement par l'intermédiaire du produit, cela traduit une fonction principale **FP**, ii) un élément unique de l'environnement et le produit, cela traduit une fonction contrainte **FC**.
- Caractériser les fonction de service. Cela permet d'exprimer les performances, attendues par l'utilisateur, de chacune des fonctions de service en terme de finalité (et non pas de moyens)

Rappelons qu'un produit peut être considéré comme le support matériel d'un certain nombre de fonctions. L'analyse fonctionnelle consiste à identifier, caractériser, ordonner, hiérarchiser et valoriser toutes les fonctions d'un produit pendant tout son cycle de vie. Pour un produit donné, l'analyse fonctionnelle utilise deux points de vue interdépendants :

- le points de vue **externe** qui est celui de l'utilisateur qui attend du produit des **services**, ou des **fonction de service**.
- Le point de vue **interne** qui est celui du concepteur qui réalise des **fonctions techniques** capables d'assurer les fonctions de service.

Selon qu'on s'intéresse aux fonctions de service ou aux fonctions techniques, on parle d'analyse fonctionnelle externe ou interne. La figure 3.3 montre les différentes étapes de l'analyse fonctionnelle qu'on doit réaliser.

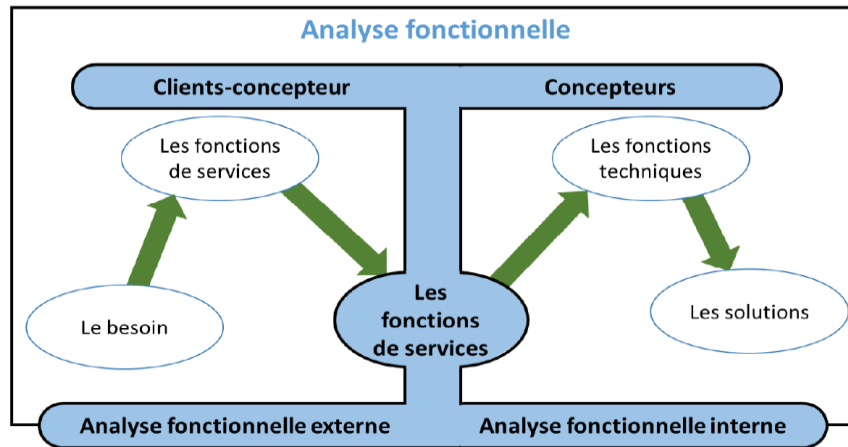


Figure 3.3. étapes de l'analyse fonctionnelle

### 3.3.1) Analyse fonctionnelle externe

On a identifié les principaux éléments du milieu extérieur du produit ciblé ainsi que les différentes fonctions de services escomptées, ils sont regroupés dans le diagramme Pieuvre de la figure 3.4. La Fonction principale du produit ciblé est de permettre à un opérateur de réaliser un usinage précis d'une manière automatique sur une pièce brute. Les fonctions de service sont brièvement décrites dans le tableau 3.2.

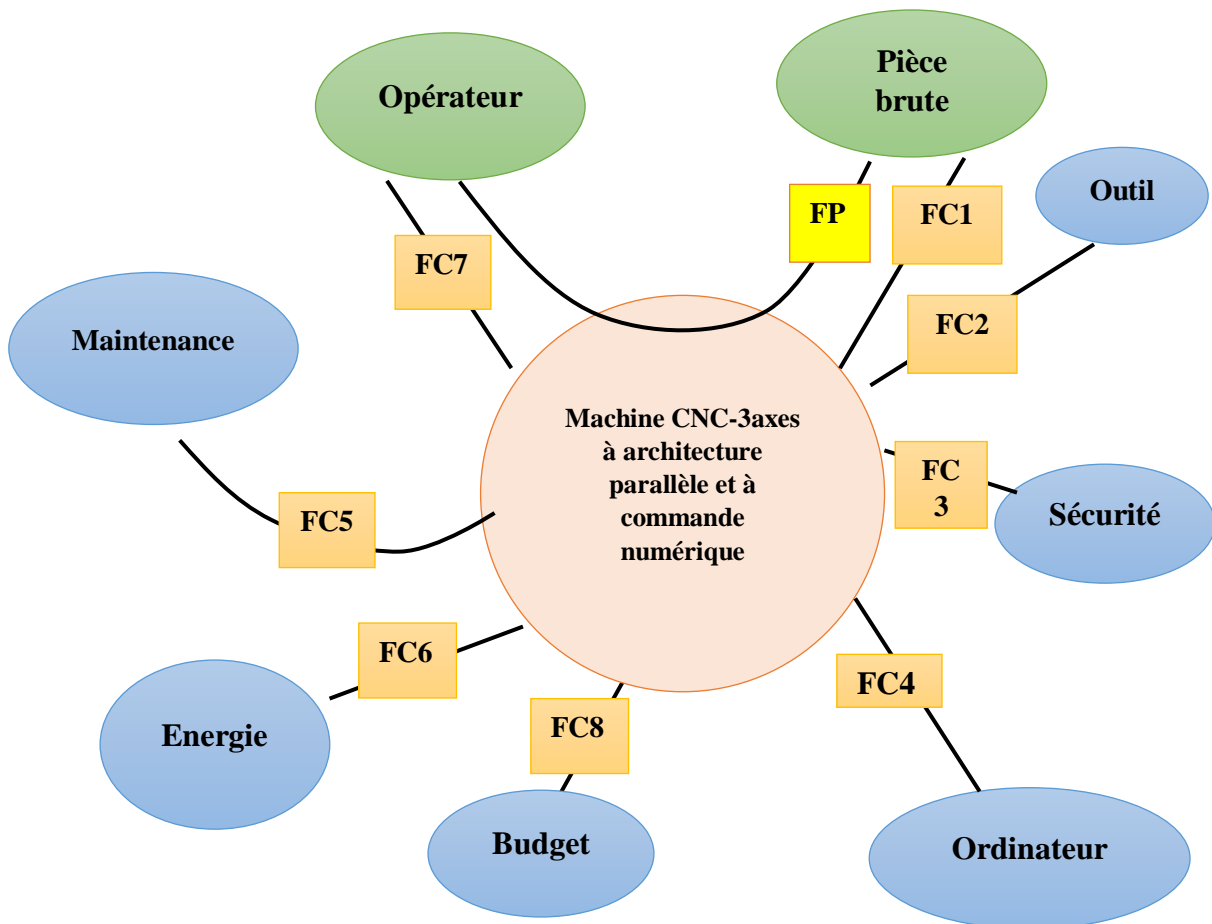


Figure 3.4. Diagramme Pieuvre

Tableau 3.2 . Description des fonctions de service

<b>FP</b>	Réaliser des opérations d' : usinage (fraisage/perçage/gravure), avec précision, sur des pièces brutes de petite taille
<b>FC1</b>	Subir les efforts de coupe
<b>FC2</b>	Faciliter le montage / démontage des outils pour différentes tailles et opérations
<b>FC3</b>	Respecter les normes de sécurité
<b>FC4</b>	Utiliser les logiciels de CFAO pour faciliter la programmation en Code G
<b>FC5</b>	Permettre une maintenance périodique aisée
<b>FC6</b>	S'adapter au réseau électrique domestique.
<b>FC7</b>	Etre facile à utiliser et à déplacer
<b>FC8</b>	Avoir un prix raisonnable

### 3.3.2) Analyse fonctionnelle interne

L'analyse fonctionnelle interne définit la vision concepteur qui doit mettre en œuvre le produit devant répondre aux besoins exprimés. Le système n'est plus considéré comme une boîte noire, mais au contraire l'analyse va porter sur l'intérieur de la boîte pour comprendre ses fonctionnalités internes.

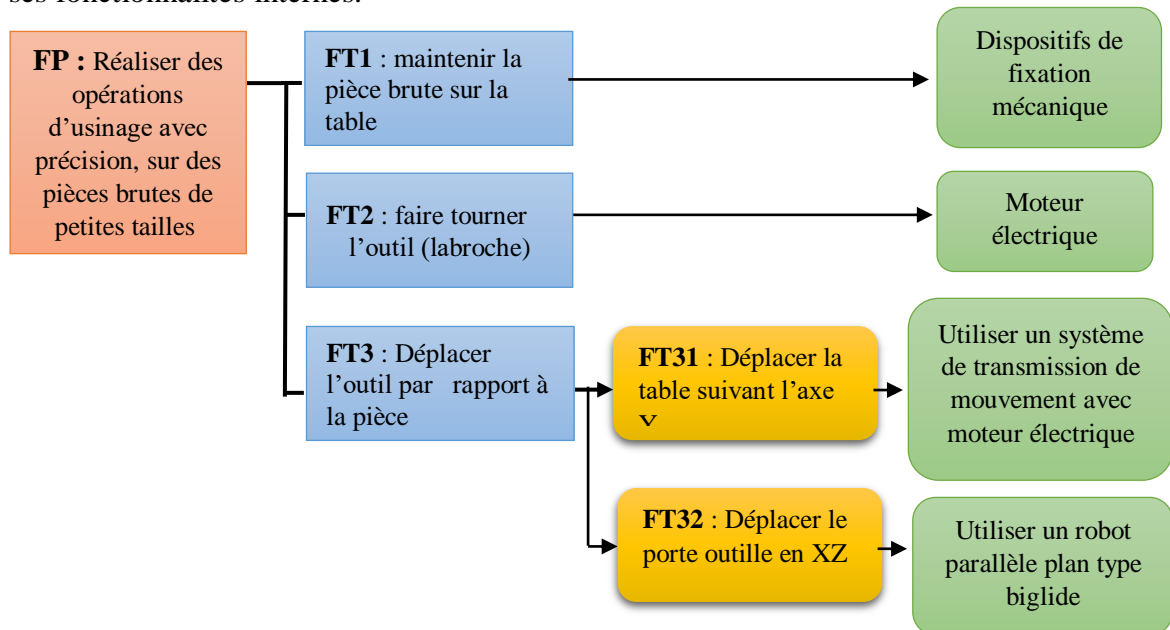


Figure 3.5. Diagramme FAST de la fonction principale du produit ciblé

Le système est considéré comme un assemblage de constituants dont chacun remplit certaines fonctions vis-à-vis des autres. L'analyse fonctionnelle interne met en évidence les fonctions techniques en faisant appel à plusieurs techniques parmi lesquelles figure le diagramme FAST (Function Analysis System Technique). Ce diagramme permet de traduire

chacune des fonctions de service en fonctions techniques, puis matériellement en solutions techniques. Le diagramme FAST de la fonction principale « **Réaliser des opérations d’usinage avec précision, sur des pièces brutes de petite taille** » est donné par la figure 3.5. La fonction « **FT31 : Déplacer la table suivant l’axe Y** » peut être décomposée d’avantage comme indiqué sur la figure 3.6.

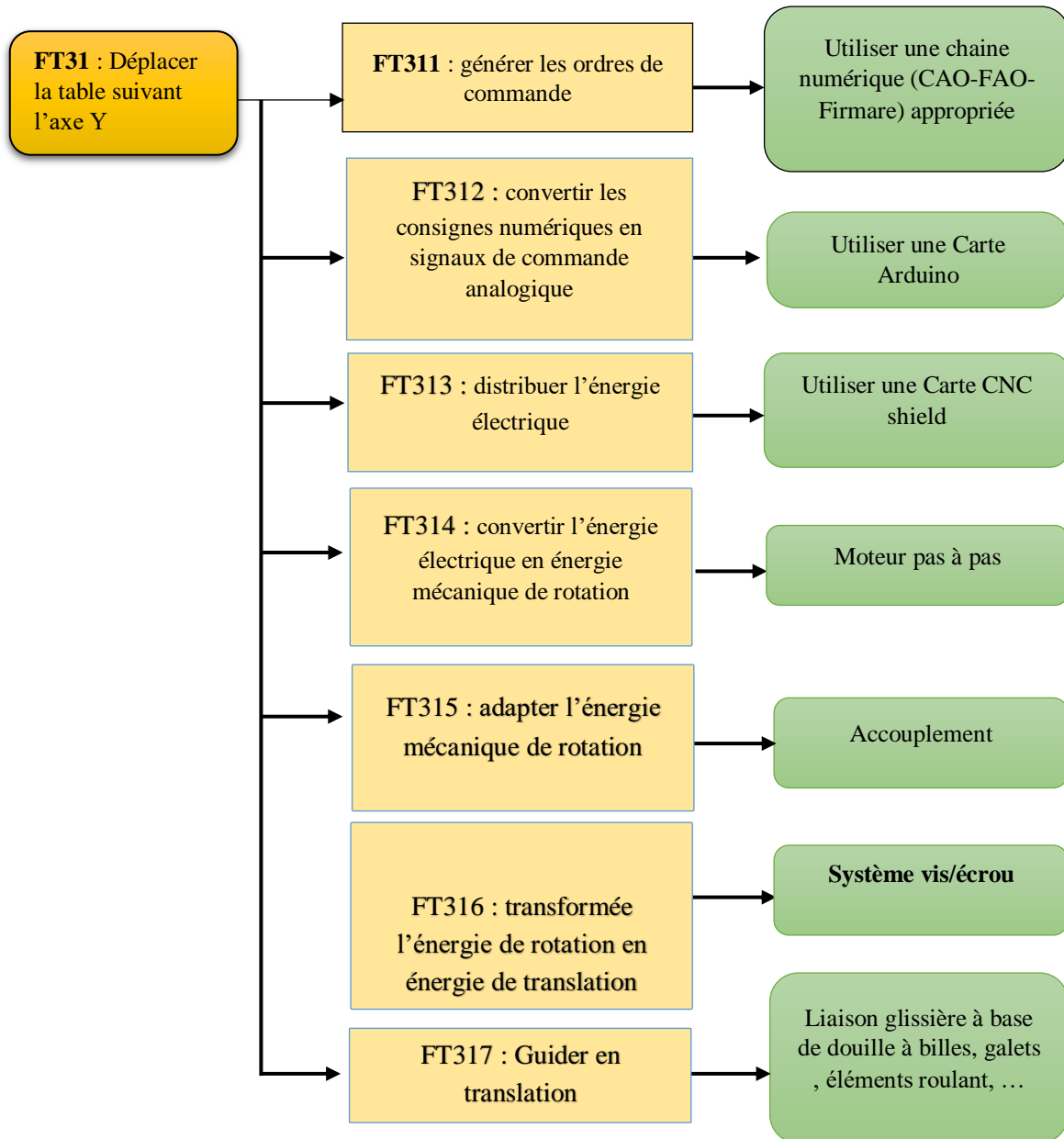


Figure 3.6. Diagramme FAST de la fonction FT31

### 3.4 Cahier des charges fonctionnel

Le cahier des charges fonctionnelles (CDCF) est le document qui récapitule la démarche et les résultats de l’analyse fonctionnelle du besoin. Il porte donc essentiellement sur les fonctions de service identifiées précédemment. Le tableau 3.3 représente la caractérisation de toutes les fonctions de services en conformité avec les besoins client préalablement établis.

Tableau 3.3. Caractérisation de fonctions de services.

Fonction	Critère	Niveau
Fp1	Vitesse d'usinage	$f_{max} = 1000 \text{ mm/min}$
	Formes de fraisage	Fraise de forme (V, arrondie, cylindrique)
Fc1	pression spécifique de coupe de bois	$K_c = 80 \text{ N/mm}^2$
Fc2	diamètre de l'outil	$D = 5\text{mm}$
Fc3	Les normes de sécurité	ISO 16090- 1:2017
Fc4	Communication	Communication série
	Synchronisation	Paquet de deux lignes de commande
Fc5	Fiabilité	R = 90%
Fc6	Type d'énergie électrique	AC 220V/1A
Fc7	Interface Homme-machine	Ecran d'ordinateur
Fc8	La taille maximale brute	90mm x 35mm x 180mm

### 3.5) Solutions technologiques

Dans cette partie les solutions technologiques possibles seront présentées dans le but de sélectionner les plus pertinentes.

#### 3.5.1) Choix N°1

Ce choix est illustré par la figure 3.7, il présente les caractéristiques suivantes :

- **Les axes :** Dans ce choix, nous avons une table mobile sur l'axe (Z), elle est considérée comme indépendante du mouvement des autres axes (X) et (Y). en effet, les deux axes(X) (Y) sont indirectement liés.
- **La nacelle :** Une pièce centrale de la machine car elle transporte la broche et la connecte aux axes (X) et (Y), la tête de la broche est en mouvement par rapport à la table mobile selon l'axe (Z)
- **Les bras :** Ils ont un rôle important dans le déplacement du mouvement des axes vers la nacelle et aident également à déterminer le domaine du travail et de l'équilibre de la broche car la forme du parallèle aide à la maintenir en position droite pour la table.

**C'est ce choix qui a été retenu par la suite au vue des moyens disponibles**



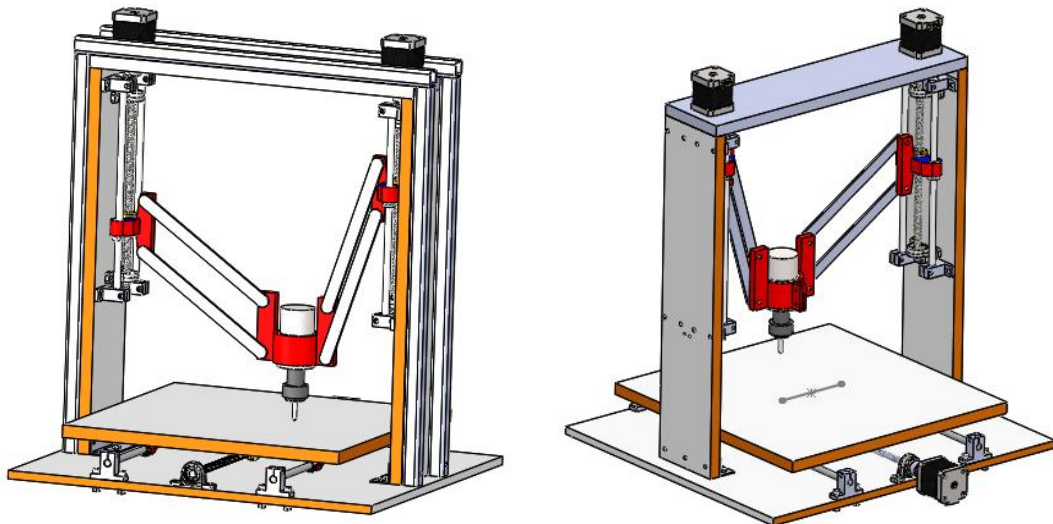


Figure 3.7. Choix technologique N°1

### 3.5.2) Choix N°2

Ce choix est illustré par la figure 3.8. En termes de forme et agencement d'axes, c'est la même chose que le premier choix, mais la différence réside dans les points suivants :

1. Le nombre de points d'attache de la nacelle (trois au lieu de quatre)
2. Il y'a trois bras pour lier la nacelle aux deux axes.
3. un volume de travail plus grand que le choix N°1.

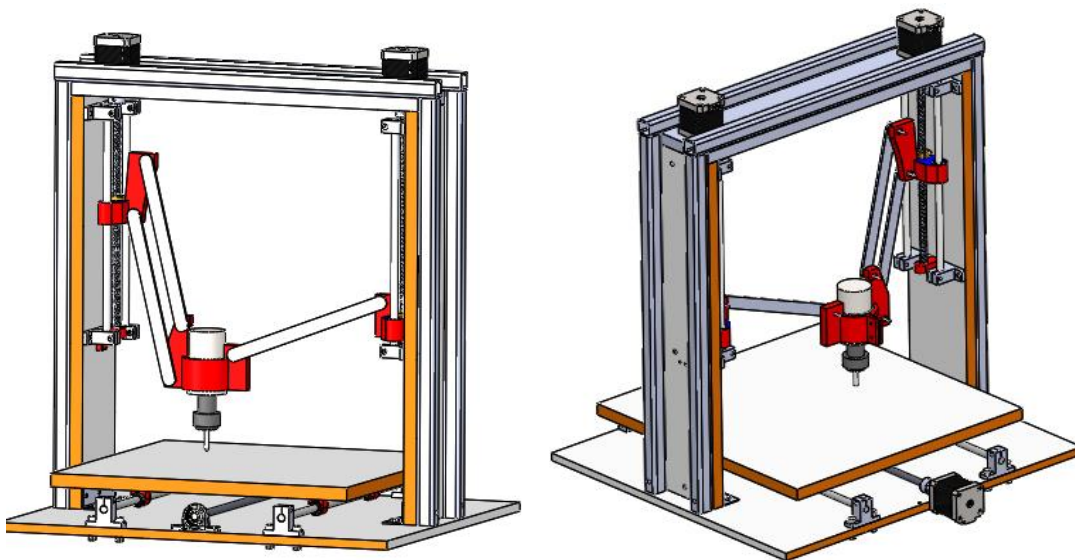


Figure 3.8 . Choix technologique N°2

### 3.5.3) Choix N°3

Ce choix est illustré par la figure 3.9, il est similaire à celui de la figure 3.7. La différence essentielle réside au niveau de la conception des biellettes d'attache de la nacelle. Elle a été renforcée pour supporter davantage d'effort

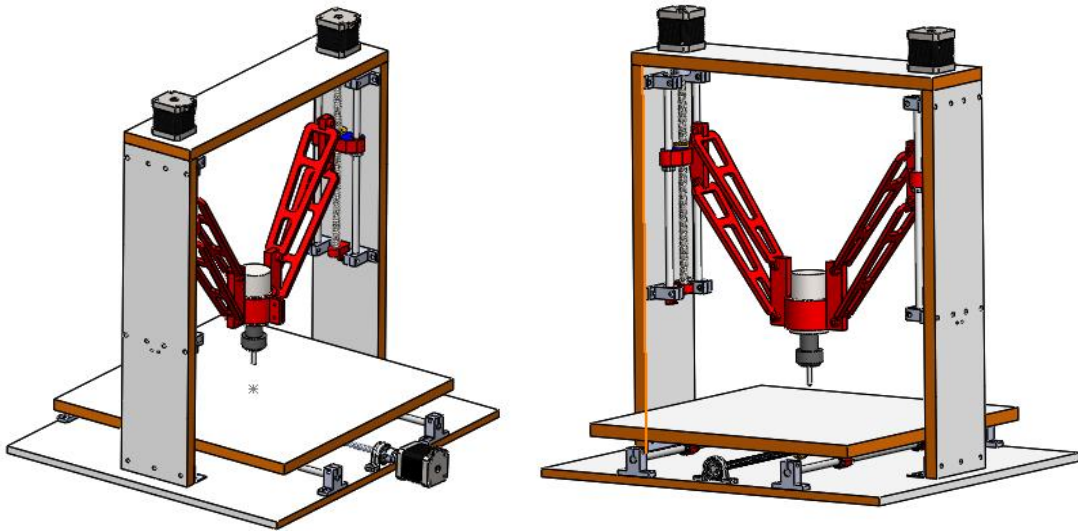


Figure 3.9. Choix technologique N°3

### 3.5.4) Sous systèmes de déplacement (X, Y et Z)

Le sous-système de déplacement linéaire est assuré par les fonctions techniques suivantes :

- Transformer l'énergie électrique en énergie mécanique.
- Transformer le mouvement de rotation en mouvement de translation.
- Guider les axes (X, Y et Z) en translation.
- Lier l'arbre moteur au système de transformation de mouvement.
- Contrôler le déplacement des axes (X, Y et Z).

#### 3.5.4.1) Transformer le mouvement de rotation en mouvement de translation

La transformation du mouvement est une fonction mécanique qui consiste à transmettre un mouvement d'une pièce à une autre, tout en modifiant sa nature. Le mouvement recherché dans notre cas, s'obtient en transformant un mouvement de rotation donné par le moteur pas à pas en un mouvement de translation de la table ou de l'outil. Parmi les systèmes de transformation du mouvement, on retrouve les systèmes suivants :

➤ **Système vis-écrou** : Le système vis-écrou permet de transformer un mouvement de rotation en un mouvement de translation en combinant les mouvements d'une vis et d'un écrou. Ce mécanisme permet d'exercer des forces et des pressions importantes. Il permet aussi des ajustements fins mais sa défaillance peut entraîner des problèmes de guidage. Les deux principaux types de mécanismes vis-écrous sont : les vis à billes et les vis trapézoïdales. Une vis à billes utilise des roulements à billes à recirculation. Le mécanisme est plus cher que le système à vis trapézoïdales mais avec un bon rendement (frottement faible), une précision et une espérance de vie plus élevées que les vis trapézoïdales. Rappelons que le pas de vis trapézoïdal est utilisé pour la transmission d'efforts importants et précis. On distingue deux sortes de filets trapézoïdaux : (i) Le pas **métrique** avec des flancs formant

un angle de  $30^\circ$ , (ii) le système Acmé dont le pas est exprimé en pouces avec des flancs formant un angle de  $29^\circ$ .



Figure 3.10. Système vis-écrou (vis à billes et vis trapézoïdale)

➤ **Système pignon-crémaillère** : Le système pignon-crémaillère transforme le mouvement de rotation du pignon en un mouvement de translation de la crémaillère. Il n'y a aucun glissement lors de la transformation de ce mouvement. La force motrice de ce système est relativement grande. Utilisé pour les grandes courses sans risques de déformations importantes, contrairement à la transmission vis-écrou. Parmi les inconvénients on peut citer :

- Nécessiter d'une lubrification importante,
- un ajustement précis à cause des dents entre la roue et la crémaillère,
- Il y a beaucoup d'usure.



Figure 3.11. Système pignon-crémaillère courroie

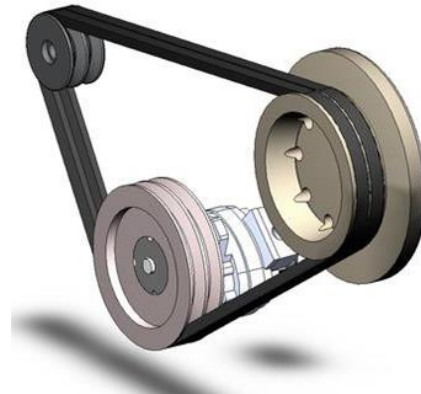


Figure 3.12. Système poulies

➤ **Système poulies-courroie** : Les poulies crantées et la courroie associée assurent une transformation de mouvement sans glissement. Les avantages du mécanisme : souplesse de la transmission due à l'élasticité de la courroie, possibilité de faire varier l'entraxe, pas de lubrification, silencieux. Les transmissions poulies-courroie sont efficaces dans les applications dont la charge utile à déplacer est faible (robots cartésiens). Les vitesses linéaires de déplacement et les accélérations sont élevées. Les inconvénients sont : usure de la courroie, entretien régulier.

Dans notre cas, Le choix s'est porté sur une transmission vis-écrou (vis trapézoïdale) pour les trois moteurs. Le choix de la solution est basé essentiellement sur une comparaison de coût et de disponibilité sur le marché.

### 3.6) Analyse de l'espace de travail du robot

L'espace de travail de la machine est borné en raison des limitations sur les longueurs des vis trapézoïdales utilisées ainsi que les inter-collisions entre les différents segments de la machine durant le mouvement. En effet, on a des limitations sur le débattement des articulations passives ainsi que par les collisions entre segments. Dans cette section, on va voir de manière explicite les limites du volume de travail du robot porteur d'outil en fonction des choix technologiques effectués dans les étapes précédentes en prenant en compte les limitations sur les longueurs des segments, les débattements des articulations passives ainsi que les intersections entre segments.

#### 3.6.1) Limites de l'espace de travail du robot dans le plan (xy)

Dans ce cas, L'analyse du volume de travail sera faite dans des états particuliers correspondant à des valeurs précises des variables articulaires  $y_1$  et  $y_2$  vues dans le chapitre 2 de la modélisation.

❖ Limite 1 :

cette limite correspond à l'état 0 de la figure 3.13, elle correspond au repliement complet du robot vers la position Home supérieure pour laquelle On a :  $y_1 = y_2 = -60\text{mm}$ ,

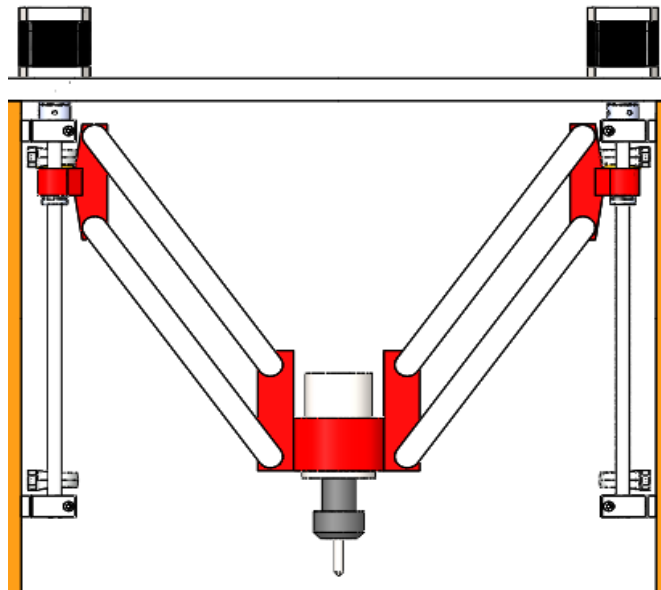
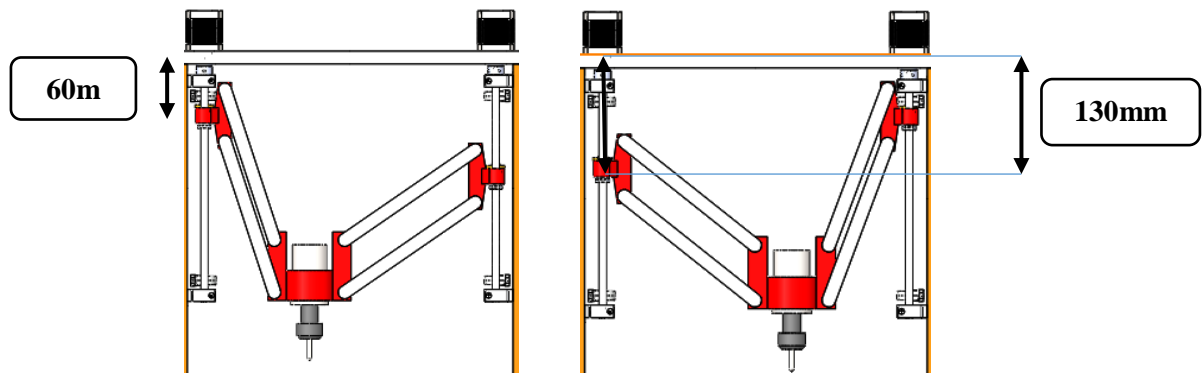


Figure 3.13. Etat 0 du robot

❖ Limite 2 :

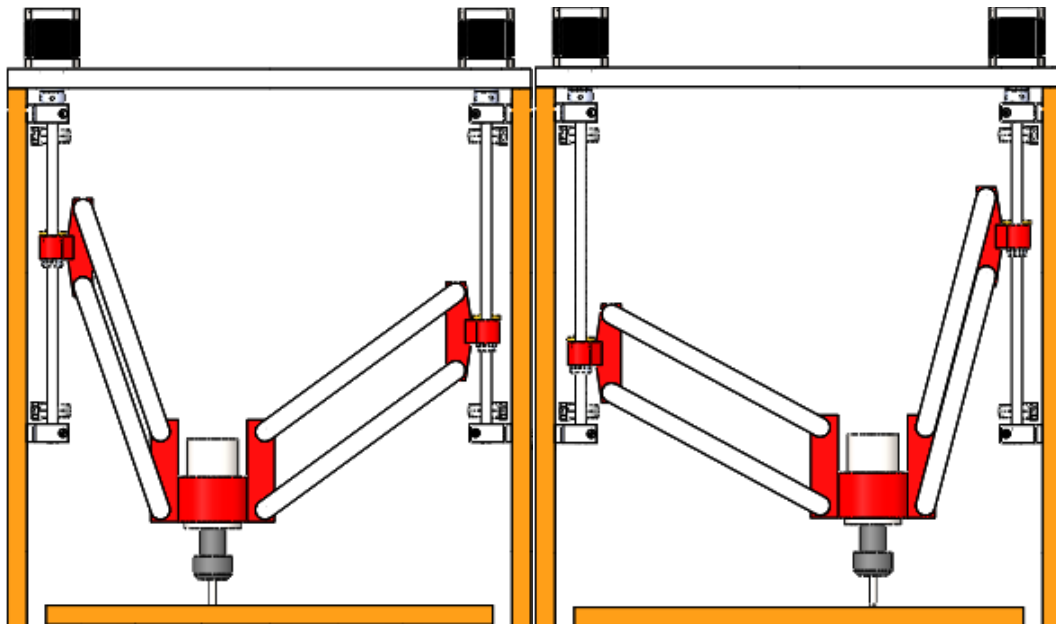
Cette limite correspond au déplacement maximal tolérable selon l'axe (Ox). Elle est obtenue en fixant l'une des variables  $y_i$  à sa valeur minimal tolérable (-60mm) et en faisant varier l'autre variable à sa valeur limite qui correspond à une inter-collision du parallélogramme correspondant comme indiqué sur la figure 3.14. Dans ce cas, on obtient le déplacement maximal selon x pour que les deux bras ne se touchent pas, on doit avoir  $x \in [-50,50]\text{mm}$ , avec  $y_{i\_max} = -130\text{ mm}$



**Figure 3.14.** Valeurs limites des variables  $y_i$  correspondant au déplacement extrême selon l'axe x

❖ **Limite 3 :**

Cette limite correspond au déplacement maximal tolérable selon l'axe (Ox) et l'axe (Oy) ayant l'outil tangent à la table.



**Figure 3.15.** Cas limites correspondants au contact de l'outil avec la Table

Pour confirmer ces limites identifiées dans SolidWorks, nous avons procédé à la programmation dans **Matlab**, à l'aide des modèles MGD et MGI développés précédemment, différents mouvements limites correspondant à ces situations. Toutes les données ont été regroupées sous Excel pour générer la frontière du volume de travail de notre robot. La figure 3.16 représente les limites du volume de travail **du robot pour un point de la nacelle sans outil**. En présence d'un outil, L'espace de travail réel doit être translaté vers le bas selon la longueur de l'outil utilisé. Par exemple, pour un outil de longueur apparente de **115 mm**, il faut décalé le volume de travail comme indiqué sur la figure 3.17. Pour préserver la machine et avoir un espace de travail régulier on a décidé de le limité comme indiqué sur la figure 3.18.

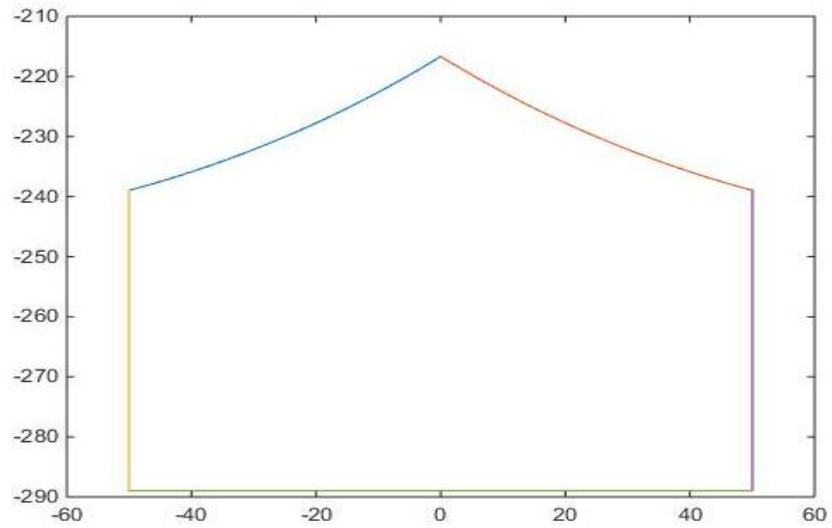


Figure 3.16. Les limites du volume de travail du robot pour un point de la nacelle

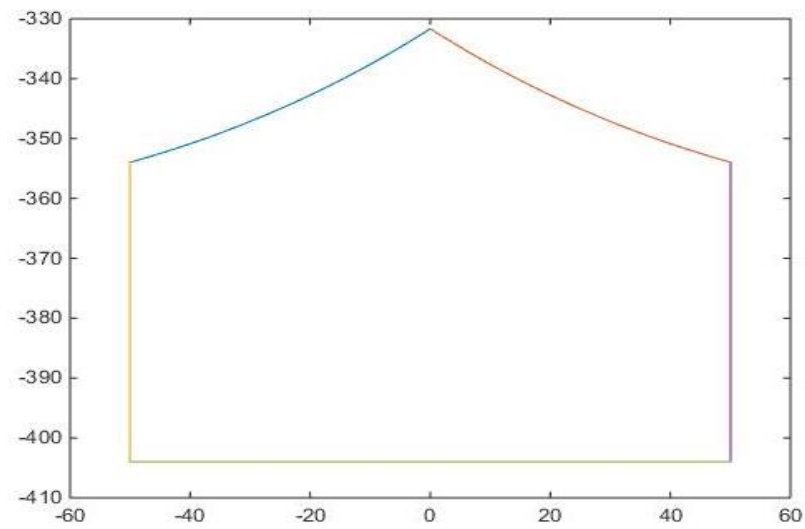


Figure 3.17. Les limites du volume de travail du robot avec un outil de 115 mm

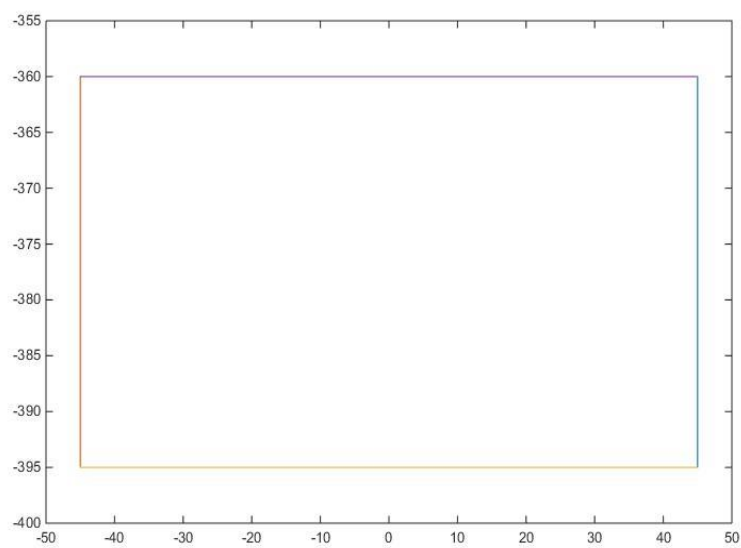


Figure 3.18. Volume de travail final

### 3.6.2) Limites de l'espace de travail du robot selon l'axe z

Ces limites définissent les mouvements admissibles de la table suivant l'axe Z. Par rapport à une position médiane ( $z=0$ , correspondant au plan vertical contenant le robot biglide), on a un déplacement de  $\pm 90$ mm, donc une course de 180mm. Elles sont clairement représentées sur la figure 3.19.

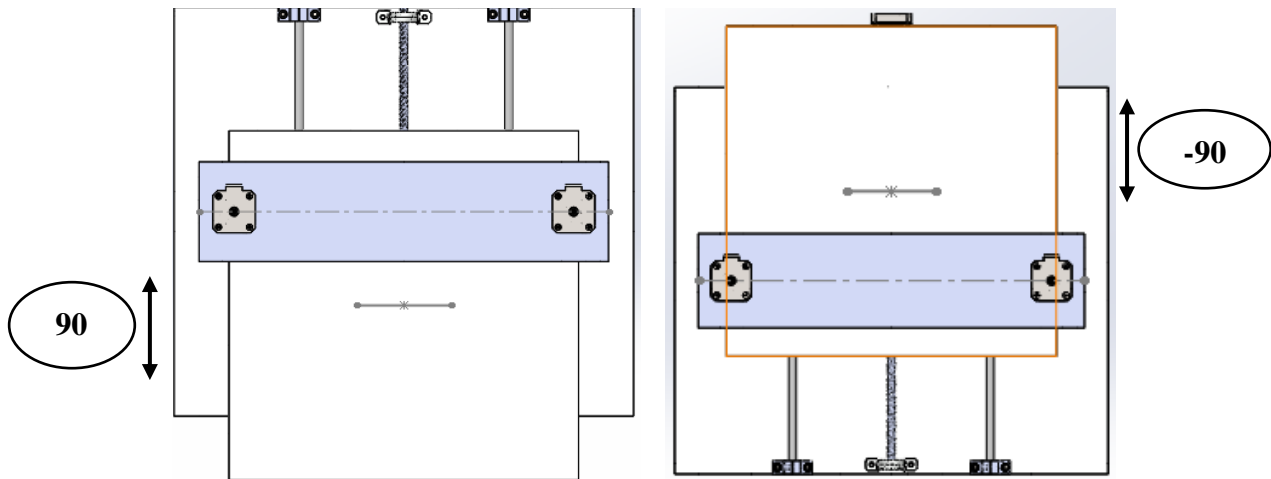


Figure 3.19. Limites du volume de travail selon l'axe z

### 3.7) Etude et calcul des principaux organes de la machine

Cette partie a pour but de dimensionner et vérifier les différents éléments fonctionnels de notre machine pour en assurer le bon fonctionnement. Il sera donc question de calculs de structures (RDM) et de puissances. Les éléments à dimensionner sont les suivants :

- Les vis trapézoïdales ;
- Les roulements ;
- Les axes de guidages.

#### 3.7.1) Estimation des masses des éléments de la machine

La machine comporte différents éléments :

- Des éléments normalisés (vis, écrous, roulements, ...) dont les masses sont rapportées dans les différents catalogues.
- Des moteurs d'entraînement ainsi que la broche, dont les masses sont aussi fournies par les fournisseurs directement sur la boîte du produit ou alors données sur leurs site internet.
- Des pièces spécialement conçues pour le cas de notre machine et dont les masses sont approximées par le logiciel de CAO SolidWorks. Celui-ci prend en compte les dimensions de la pièce et le matériau de construction.

Le tableau 3.4 regroupe les estimations des masses des différents éléments.

Tableau 3.4. Les masses des différents composants du dispositif

Déplacements	Nom de pièce	Nombre de pièces	$m_i$	Masse (kg)
Direction Y	la broche	1	$m_1$	0,5
	paliers à douilles à billes	2	$m_2$	0.05
	support broche	1	$m_3$	0.08
	les barres du manipulateur	4	$m_4$	0.08
Direction Z	La table	1	$m_5$	0.3
	paliers de transmission	1	$m_6$	0.05

### 3.7.2) vérification des vis trapézoïdales

#### 3.7.2.1) Calcul de la charge $F$ applique sur la vis suivant les axes Y et Z

##### ❖ Suivant l'axe Y :

Pour l'axe Y il y a des vis trapézoïdale parallèle et symétriques par rapport à l'axe machine, en plus de symétrie des masses, donc nous allons dimensionne une seule vis.

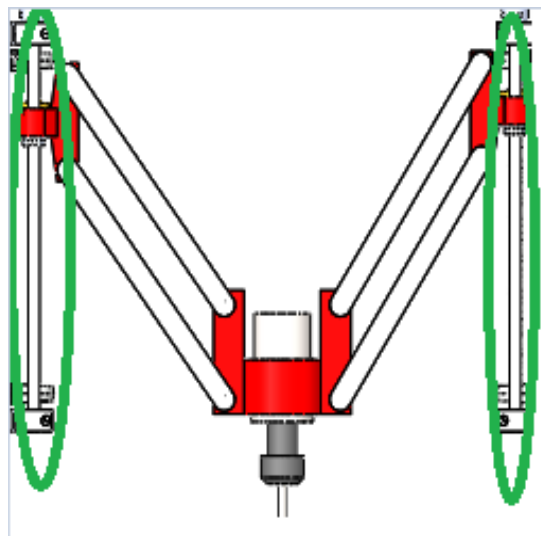


Figure 3.20. Système de translation suivant Y

La charge  $F$  suivant Y est purement axiale et comporte en plus du poids des composants ( $F_1$ ), les forces d'inertie dues aux accélérations et aux décélérations ( $F_2$ ) ainsi que les efforts de coupes et de pénétration ( $F_3$ ). On aura donc :

$$F_y = F_1 + F_2 + F_3 \quad (3.1)$$

Où  $F_1 = \sum . m_i . g$  : la masse du  $i$  élément,  $g$  : La gravité. Donc



$$m_i = m_1/2 + m_3/2 + m_4 \times 4 + m_2 = 0.5 \text{ kg} \quad \text{Alors, } F_1 = 4,9 \text{ N}$$

$$\text{Pour la force} \quad F_2 = \sum m_i \times a \quad (3.2)$$

Où  $a$  : représente l'accélération et elle est calculée comme suit :

$$a = \frac{\Delta V}{\Delta t} \quad (3.3)$$

$\Delta V$  : est la variation de vitesse linéaire. On aura donc

$$\Delta V = \frac{p.n}{60} = \frac{2.1000}{60} = 33.33 \text{ mm/s} \quad (3.4)$$

Avec :

- $p$  est le pas du filetage [mm]. Dans notre cas  $p = 2 \text{ mm}$ .
- $n$  est la vitesse de rotation de la vis [tr/min]. On la fixe dans notre cas à  $n = 1000 \text{ tr/min}$

On prend  $\Delta t = 0,1 \text{ s}$  qui est un temps de réponse assez court.

On aura ainsi :  $a = 0,3 \text{ m/s}^2$  et  $F_2 = 16,66 \text{ N}$

$$F_3 \text{ représente l'effort de pénétration } F_p: \quad F_3 = F_p = \frac{1}{3} \times \frac{F_c}{2}$$

Où  $F_c$  est l'effort de coupe

$$F_c = K_c. a. f \quad (3.5)$$

$$f = \frac{p.n}{N} = \frac{1000.2}{10000} = 0,2 \text{ mm}$$

Avec :

- $f$  : Avance de l'outil par rotation de la broche en [mm]
- $K_c$  : Coefficient spécifique de coupe en [daN/mm<sup>2</sup>]. On prend  $K_c = 80 \text{ daN/mm}^2$  pour usine Alliages d'aluminium
- $a$  : Profondeur de passe en [mm] dans notre cas  $a = 1 \text{ mm}$ .
- $N$  : Vitesse de rotation de la broche en [tr/min]. On prend  $N = 10000 \text{ tr/min}$ .

Les valeurs obtenues sont :  $F_c = 320 \text{ N}$ , Donc :  $F_3 = 53,33 \text{ N}$

La force axiale totale  $F$  est :  $F_y = 74,89 \text{ N}$

#### ❖ Suivant l'axe Z

On raisonnera de la même façon que pour l'axe Y sauf que cette fois la somme des forces sera divisée par trois car celle-ci est supportée par les deux axes de guidage et la vis,

$$F_z = \frac{1}{3} (F_1 + F_2 + F_3) \quad \text{Avec} \quad m_i = m_1 + 2m_2 + m_3 + 4m_4 + m_5 + m_6 = 1,35 \text{ kg}$$

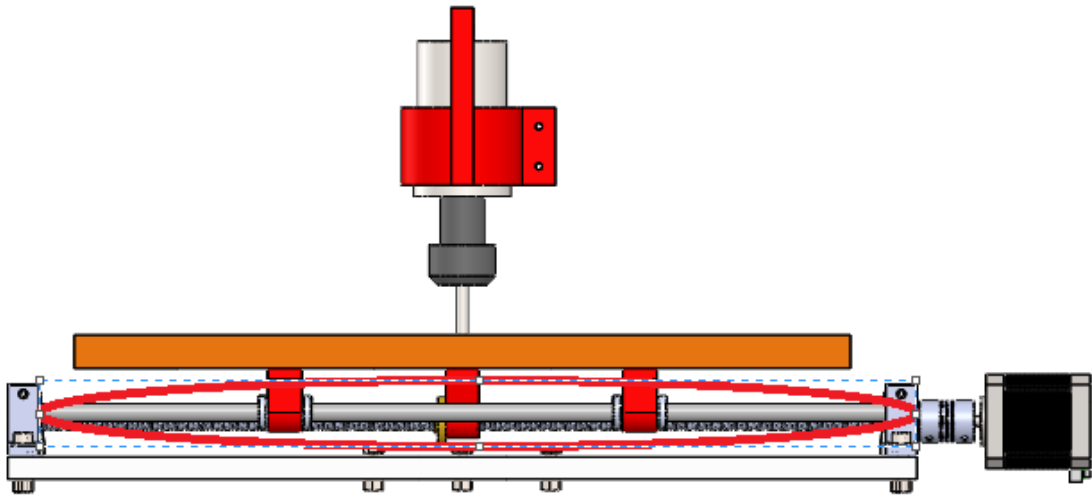


Figure 3.21 : Système de translation suivant Z

La différence apparait aussi dans la masse des parties mises en mouvements. Ici nous aurons :

$$F_1 = 13,23 \text{ N}$$

$F_2$  Change aussi vue qu'elle est fonction de la masse (l'accélération ne varie pas).

Donc :

$$F_2 = 45 \text{ N}$$

$F_3$  représente ici l'effort de coupe  $F_c$ . Donc :

$$F_3 = F_c = 320 \text{ N}$$

On aura donc :

$$F_z = 126,07 \text{ N}$$

**Remarque :**  $F_1$  ,  $F_2$  ,  $F_3$  suivant l'axe Z ce ne sont pas les même forces suivant l'axe Y.

### 3.7.2.2) vérification des vis trapézoïdales à l'usure

La paire vis/écrou trapézoïdale est un système utilisé depuis longtemps pour la transformation dans de nombreuses applications d'un mouvement rotatif en un mouvement linéaire. La puissance totale  $P$  transmise par la vis à l'écrou est transformée en puissance  $P_u$ . Le rapport  $P_u / P$  = définit le rendement du système qui dépend du coefficient de frottement entre les surfaces de contact entre la vis, l'écrou et l'angle d'hélice du filetage. Nous sommes en présence d'un frottement de glissement. . Une partie de la puissance est donc transformée en chaleur à chaque mouvement.

Il est possible de paramétrer ce frottement de glissement pour évaluer le bon fonctionnement de l'ensemble. Le critère est de limiter la pression superficielle de contact sur le côté du filetage afin de permettre un glissement doux entre les deux surfaces et d'éviter les frottements .Nous limitons aussi le produit  $P \times V_{st}$  ( $p$ = pression superficielle de contact et  $V_{st}$  = vitesse de frottement sur le diamètre moyen du filetage) afin de limiter la puissance perdue sous forme de chaleur. [34]

Dans notre cas on doit aussi prendre en compte l'influence des forces d'inerties engendrées par les accélérations et les décélérations du système. Celles-ci seront représentées par un coefficient de sécurité  $f_i$  Dont les valeurs seront tirées du **Tableau 3.5** On calculera donc le produit

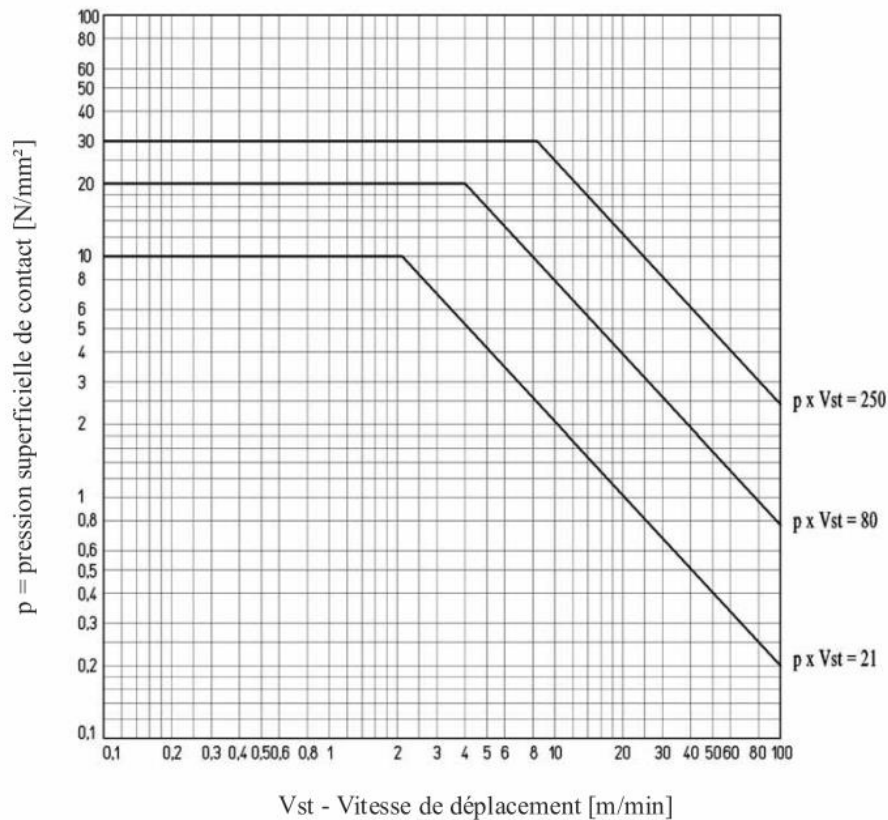
$$P \cdot V_{st} \cdot f_i \tag{3.6}$$

**Tableau 3.5.** Coefficients de sécurité pour différents types de charges [34]

Type de charge	$f_i$
Charge constante avec rampe d'acc/déc contrôlée	De 1 à 0,5
Charge constante avec démarrages et arrêts à déchirure	De 0,5 à 0,33
Charge et vitesses très variables	De 0,33 à 0,25
Charge avec chocs et vibrations	De 0,25 à 0,17

On prendra donc dans notre cas  $f_i = 0,75$

En ce qui concerne les écrous en bronze (qui est notre cas), le produit  $P \times V_{st}$  permet de tracer le graphique présenté dans la Figure où trois zones sont mises en évidence, chacune caractérisée par certaines conditions d'utilisation, entre la vitesse de glissement et la pression.



**Figure 3.22 :** Condition de glissement des écrous de bronze sur les vis en acier [34]

➤ **Zone A** : La zone A est la limite de  $P \times V_{st} = 21$  [N/mm<sup>2</sup>.m/min]. Dans cette zone, le fonctionnement est dans les meilleures conditions. L'« utilisation continue » est possible car la quantité de chaleur produite entre ces limites de  $P \times V_{st}$  est plutôt maîtrisée. La durée de vie de l'écrou est très bonne.

➤ **Zone B** : La zone B est la limite de  $P \times V_{st} = 80$  [N/mm<sup>2</sup>.m/min] Dans cette zone, le fonctionnement se trouve dans des conditions difficiles. Les conditions de glissement sont telles qu'une constante lubrification est nécessaire pour maîtriser l'érosion bronze afin d'obtenir une bonne durée de vie de l'écrou. L'utilisation « continue » est possible seulement pour des périodes limitées car la quantité de chaleur produite est telle qu'elle provoque un échauffement très marqué de l'écrou, et dépend aussi directement de la qualité du lubrifiant utilisé, nous sommes dans un cas où la durée de vie de l'écrou est limitée.

➤ **Zone C** : La zone C est la limite  $P \times V_{st} = 250$  [N/mm<sup>2</sup>.m/min] Dans cette zone, le fonctionnement est dans conditions très sévères. Avec ces valeurs de  $P \times V_{st}$ , travailler en « utilisation continue » n'est pas possible. Même en présence d'une bonne lubrification, il se produit un échauffement intense et une usure prématurée de l'écrou car le glissement entre les surfaces de contact est tel qui provoque une érosion rapide de l'écrou.

Les formules à utiliser sont:

✓ **Pression superficielle de contact P [34]:** 
$$P = \frac{F}{A_t} \quad (3.7)$$

où : - **F** est la force axiale en [N].

- **A<sub>t</sub>** Surface d'appui totale entre les dents de la vis et les dents de l'écrou. sur le plan perpendiculaire à l'axe. [mm<sup>2</sup>].

Et : 
$$A_t = \pi \cdot d_m \cdot Z \cdot H1 \quad (3.8)$$

Avec :

- **d<sub>m</sub>** = diamètre moyen du filetage [mm].
- **H1** = dimension radiale d'appui entre les dents de la vis et les dents de l'écrou [mm].
- **Z** = nbre des dents en prise.

$$Z = \frac{\text{h ecrou}}{\frac{\text{pas effectif}}{\text{n° principes}}} \quad (3.9)$$

Dans notre cas la valeur de **A<sub>t</sub>** est tirée du catalogue **A<sub>t</sub> = 250 mm<sup>2</sup> [34]**.

✓ **Vitesse de glissement V<sub>st</sub> [34]:** 
$$V_{st} = \frac{n \cdot p}{1000 \cdot \sin \alpha} \quad (3.10)$$

Où **α** est l'angle d'hélice du filetage. Il est tiré du catalogue [36]. Dans notre cas **α = 4.02°**

✓ **Applications numériques :**

❖ Suivant l'axe Y

La pression superficielle de contact est d'après (3.7) est  $P = \frac{F_y}{A_t} = 0,3$

La vitesse de glissement est d'après (3.10)  $V_{st} = 28,52 \text{ m/min}$

Le produit (3.6) nous donne :  $P \cdot V_{st} \cdot f_t = 6,41 \text{ Mpa. m/min}$

En conséquence, D'après le diagramme de la **Figure**, la vis travaille dans la zone A. Les conditions de fonctionnement sont idéales et donc l'usure est faible même en utilisation prolongée.

❖ Suivant l'axe Z

La pression superficielle de contact est d'après (3.7) est :  $P = \frac{F_z}{A_t} = 0,50 \text{ MPa}$

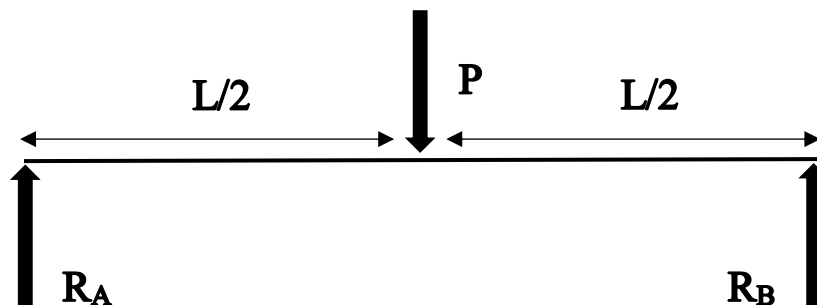
La vitesse de glissement est d'après (3.10),  $V_{st} = 28,52 \text{ m/min}$

Le produit (3.6) nous donne :  $P \cdot V_{st} \cdot f_t = 10,69 \text{ Mpa. m/min}$

**Donc**, Idem que pour l'axe Y, la vis trapézoïdale travaillant suivant l'axe Z évolue dans la zone A du diagramme et donc le fonctionnement est dans les meilleures conditions et l'influence de l'usure est minime.

**3.7.2.3) vérification des vis trapézoïdales à la flexion**

La vis trapézoïdale peut être représentée par une poutre de section circulaire faite en acier inoxydable de  $D = D_{min} = 8,739$  (disponibilité), encastrée aux extrémités et subissant un effort au milieu qui tend à la faire fléchir, comme décrit dans la Figure 3.23.



**Figure 3.23.** Diagramme des forces et des réactions.

**P** représente le poids des éléments soutenus par la vis. Ce poids est représenté au milieu de la vis où la flèche est maximale.

Le système étant symétrique on déduit que :  $R_A = R_B = \frac{P}{2}$  (3.15)

$$M_{fA} = M_{fB}$$

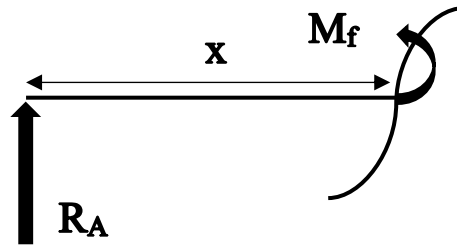


Figure 3.24. Diagramme du moment fléchissant

$$M_{fA} = M_f = R_A \cdot x$$

Le cas plus danger on  $x = \frac{L}{2}$  donc :  $M_{fA} = M_{fmax} = \frac{P}{4}L$  (3.16)

On calcule ensuite la contrainte de flexion  $\sigma_f$  pour la comparée à la contrainte  $\sigma_{fadm}$  admissible.

Sachant que :  $\sigma_f = \frac{M_{fA}}{R_f}$  (3.17)

Où  $R_f$  est le moment de résistance à la flexion :  $R_f = \frac{\pi \cdot d^3}{32} = 64.48 mm^3$  (3.18)

La contrainte de flexion admissible est calculée par la formule :

$$\sigma_{fadm} = \frac{R_e}{S_f} \quad (3.19)$$

Tel que  $S_f = 2$  qu'est le coefficient de sécurité. Et  $R_e = 360 MPa$ ,

Pour le matériau utilisé, acier E360 , on a  $\sigma_{fadm} = 180 MPa$

L'application numérique nous donne

$$R_A = R_B = \frac{F_1}{3} \times \frac{1}{2} = 0, 57 N$$

On a divisé  $F_1$  par trois car la masse est supportée par les deux axes de guidages et la vis trapézoïdales alors :  $M_{fA} = 114 N \cdot mm$   $\sigma_f = 1,769 MPa$

Comparant cette valeur à celle de  $\sigma_{fadm}$   $\sigma_f \ll \sigma_{fadm}$

Donc la tige filetée reste dans le domaine de sollicitation élastique.

### ➤ Vérification des axes de guidages

Les chariots de la mini-fraiseuse sont guidés en translation sur des tiges en acier E360 lisse et par des douilles à billes. Ces tiges sont soumises à la flexion principalement, on suivra donc la même procédure de calcul que pour le dimensionnement des vis trapézoïdales. Sachant que  $d = 8 mm$  (disponibilité), ce qui nous donne  $R_f = 50.24 mm^3$ . Aussi, on a  $R_e = 185 MPa$  (acier inoxydable).

Le reste des valeurs demeure inchangé. Alors,  $\sigma_{fadm} = \frac{R_e}{S_f} = 92, 5 MPa$

❖ Selon l'axe Z       $M_{fA} = 114 \text{ N. mm}$        $\sigma_f = 2,26 \text{ MPa}$

Comparant cette valeur à celle de  $\sigma_{fadm}$        $\sigma_f \ll \sigma_{fadm}$

Donc la tige reste dans le domaine de sollicitation élastique.

### 3.8) Choix des moteurs

Pour choisir les moteurs d'entraînement des vis trapézoïdales il faut déterminer les couples et les puissances nécessaires de chaque moteur. Les formules sont les suivantes [34]:

$$C_Z = \frac{F.P}{2\pi\eta \times 1000} \quad (3.11)$$

Avec :       $\eta = \frac{1-f \tan \alpha}{1+\frac{f}{\tan \alpha}} \quad (3.12)$

$$P_{uz} = C_z. \omega \quad (3.13)$$

Avec :

- $f$  est le coefficient de frottements dynamiques. Pour le frottement acier-bronze,  $f = 0.18$

- $\omega$  est la vitesse angulaire de la vis       $\omega = \frac{2\pi n}{60} = 104,66 \text{ rd/s} \quad (3.14)$

d'après (3.12) on a alors       $\eta = 0,27$

#### ❖ Détermination du couple et la puissance nécessaire selon l'axe Y

On a :  $F = 74,89 \text{ N.}$

Le couple de torsion nécessaire au déplacement du chariot suivant Y est donc selon (3.11):

$$C_z = 0,08 \text{ N. mm}$$

La puissance nécessaire sera donc de :  $P_{uz} = 9,24 \text{ W}$

On choisit alors, un moteur Nema 17 de couple  $C_z = 310 \text{ N. mm.}$

#### ❖ Détermination du couple et la puissance nécessaire selon l'axe Z

On a  $F = 378.21 \text{ N.}$

Le couple de torsion nécessaire au déplacement du chariot suivant Y est donc selon (3.11):

$$C_z = 0,45 \text{ N. mm}$$

La puissance nécessaire sera donc de :  $P_{uz} = 47,09 \text{ W}$

On choisit alors un moteur Nema 17 de couple  $C_z = 310 \text{ N. mm.}$

### 3.9) Dimensionnement des roulements

On dimensionne les roulements placés aux paliers des vis suivant deux critères

- La capacité de charge dynamique. C'est-à-dire  $C_{roulement} > C_{nécessaire}$ ,
- La durée de vie. C'est-à-dire  $L_h_{roulement} > L_h_{nécessaire}$  Pour cela on utilisera les formules suivantes [35] [36] :

$$L_H = \left(\frac{C}{P}\right)^k \cdot \frac{10^6}{60 \cdot N} \quad (\text{heures}) \quad (3.20)$$

$$C = \frac{f_L \cdot f_Z \cdot (P)}{f_t \cdot f_n \cdot f_H} \quad (3.21)$$

Ou:

- $f_Z$  : Coefficient supplémentaire des efforts dynamique. On prend  $f_Z = 1$ .
- $f_H$  : Coefficient de dureté. On considère  $f_H = 1$
- $f_t$  : Coefficient de température. Dans notre cas  $f_t = 1$
- $k = 3$  pour les roulements à billes [35]
- $N$  : vitesse de rotation en tr/min. = 1000
- $f_L$  : Coefficient de durée de vie. **On prend**  $L_h_{nécessaire} = 40000 \text{ h}$

$$f_L = \sqrt[3]{\frac{L_H_{nécessaire}}{500}} = 4.3 \quad (3.22)$$

- $f_n$  : Coefficient de nombre de tours.

$$f_n = \sqrt[3]{\frac{100}{3 \cdot n}} = 0.32 \quad (3.23)$$

#### ❖ Dimensionnement des roulements selon l'axe Y

Sachant que  $P = \frac{74,89}{2} = 37,44 \text{ N}$

On aura donc :  $C = 0,5 \text{ KN}$

On choisit donc le roulement à billes NTN 6800 (d =10, D =19)

Avec  $C_{dyn} = 1.83 \text{ KN}$  ,  $C_{stat} = 0.925$  [37]

On vérifie maintenant la durée de vie pour le roulement choisi.

$$L_h_{roulement} \gg L_h_{nécessaire}$$

$$L_h_{roulement} = 1946231 \text{ (heures)} \gg 40000 \text{ (heures)}$$

#### ❖ Dimensionnement des roulements selon l'axe Z

Sachant que  $P = F/3/2 = 55,84 \text{ N}$ . Où  $F = 335,09 \text{ N}$

On aura donc :  $C = 0,75 \text{ KN}$

**Remarque :** On a divisé la charge supportée par le roulement en 2 car elle est répartie sur deux paliers de roulements.



On choisit le même roulement à billes NTN 6800 ( $d = 10$ ,  $D = 19$ )  
avec  $C_{dyn} = 1.83 \text{ KN}$ ,  $C_{stat} = 0.925 \text{ KN}$ . [37]

On vérifie maintenant la durée de vie pour le roulement choisi.

***Lh roulement >> Lh nécessaire***

***Lh roulement = 586632 (heures) >> 40000 (heures)***

### **3.10) Conclusion**

Dans ce chapitre, nous avons traité le volet conception de la machine. Nous avons d'abord abordé l'analyse fonctionnelle pour aboutir à un CdCF réalisable pour laquelle on a sélectionné un ensemble de solutions technologiques. Par la suite on a eu à dimensionner les parties les plus sensibles de notre système (vis, tiges, Roulements, ...). On a pu constater que tous les éléments résistaient parfaitement aux différentes sollicitations auxquelles elles sont soumises. Dans le dimensionnement à l'usure des vis sans fins, on a constaté que le système fonctionne Dans les meilleures conditions de température et de frottement possibles où le fonctionnement Continu est possible et les pertes de puissances sont minimales. Globalement, les matériaux utilisés ainsi que les techniques employées à leurs réalisations Assurent un fonctionnement optimale et sans risques.

## *Chapitre IV :*

# *Simulation du comportement de la machine*

### 4.1) Introduction aux études de mouvement sous SolidWorks

Les études de mouvement sont des simulations graphiques du mouvement dans des modèles d'assemblage qui se fait sous SolidWorks à l'aide de l'outil *Motion* (Figure 4.1). C'est un outil de prototypage virtuel qui prend en charge l'animation, l'analyse et la conception de mécanismes. Au lieu de construire et de tester physiquement des prototypes de mécanismes, on peut utiliser *SOLIDWORKS Motion* (*SW-Motion*) pour évaluer et affiner le mécanisme avant de finaliser la conception et réaliser le prototypage fonctionnel. Cet outil a été utilisé d'une façon intensive dans notre projet pour valider les différentes solutions mécaniques.

Les fonctionnalités disponibles dans *SW-Motion* aident également à rechercher de meilleures alternatives de conception. Un meilleur l'alternative de conception dépend beaucoup du problème. Il est essentiel qu'un objectif de conception soit clairement défini par le concepteur avant de rechercher de meilleures alternatives de conception.

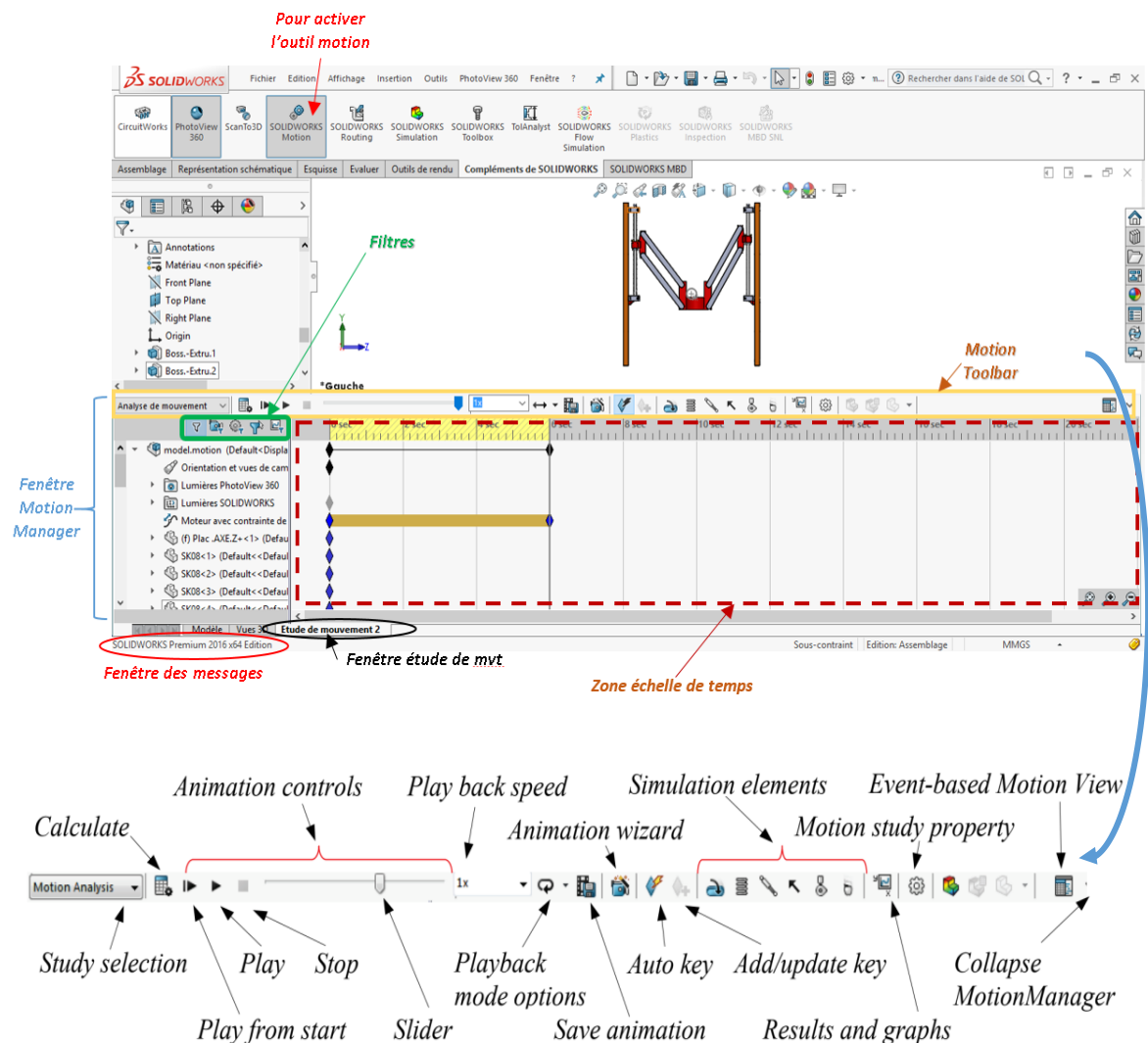


Figure 4.1. Éléments de base de l'Interface utilisateur pour SolidWorks Motion

Les études de mouvement sous *SW-Motion* ne modifient pas le modèle d'assemblage ou ses propriétés. Elles simulent et animent le mouvement prescrit. Les contraintes de *SW-Motion* peuvent être utilisées pour restreindre le mouvement des composants dans un assemblage lors de la modélisation du mouvement.

Le processus global d'utilisation de *SW-Motion* pour analyser un mécanisme consiste de trois étapes principales : génération du modèle, analyse (ou simulation), et visualisation des résultats. Les entités clés qui constituent un modèle de simulation comporte des servomoteurs qui pilotent le mécanisme pour une analyse cinématique, des charges externes (force et couple), des composants imposant des forces tels que le ressort et l'amortisseur, et les conditions initiales du mécanisme. Sans oublier, les contraintes d'assemblage qui doivent être correctement définies pour que le mécanisme simulé reprenne le plus fidèlement possible le comportement physique mécanisme réel.

## 4.2) Vue d'ensemble de la fenêtre étude de mouvement

A partir d'une étude de mouvement, on peut utiliser le Motion Manager, une interface sous forme de chronogramme, qui comprend les outils d'étude de mouvement suivants :

### ➤ Animation

L'animation est utilisée pour animer le mouvement d'assemblages après avoir

- Ajouté des moteurs pour piloter le mouvement d'une ou de plusieurs pièces dans un assemblage.
- Spécifié la position des composants de l'assemblage à divers moments en utilisant des clés. Animation utilise l'interpolation pour définir le mouvement des composants entre les clés.

### ➤ Mouvement standard

Cette option permet de simuler les effets des moteurs, ressorts, contacts et de la gravité sur des assemblages. Le mouvement standard prend en compte la masse dans le calcul du mouvement. L'analyse étant relativement rapide, cette option peut être utilisée pour créer des animations de qualité professionnelle basées sur les lois de la physique.

### ➤ Analyse de mouvement

On peut utiliser Analyse de mouvement pour simuler et analyser les effets d'éléments de mouvement (forces, ressorts, amortisseurs et frottement) sur un assemblage. Cette option utilise des solveurs de cinématique puissants et tient compte des propriétés du matériau, ainsi que de la masse et de l'inertie, dans l'analyse. L'Analyse de mouvement peut aussi être utilisée pour tracer les résultats de la simulation en vue d'une analyse plus poussée.

La barre d'outils Motion Manager sert à :

- Modifier les points de vue.
- Afficher les propriétés.
- Créer des animations de qualité professionnelle, facilement distribuables, montrant votre assemblage en mouvement.

C'est cet outil qui va être utilisé pour analyser les mouvements de notre machine CNC.

On peut utiliser l'Analyse de mouvement pour simuler et analyser le mouvement d'un assemblage avec précision tout en incorporant les effets d'éléments d'étude de mouvement (forces, ressorts, amortisseurs et frottement).

Une étude d'analyse de mouvement combine les éléments d'étude de mouvement avec les contraintes dans les calculs du mouvement. Par conséquent, les contraintes de mouvement, les propriétés du matériau, la masse et les contacts entre composants sont pris en compte dans l'analyse réalisée par le solveur de cinématique SOLIDWORKS Motion.

Une étude d'analyse de mouvement vous permet également de calculer les chargements pouvant être utilisés dans la définition de scénarios de chargement à des fins d'analyse structurelle.



Ainsi, pour utiliser le solveur SOLIDWORKS Motion dans une étude de mouvement, il faut sélectionner l'Analyse de mouvement dans la liste type d'études de mouvement du Motion Manager. Par ailleurs, on peut utiliser le Mouvement standard pour simuler les effets de moteurs, de ressorts, de collisions et de la gravité sur des assemblages. Mouvement standard prend en compte la masse dans le calcul du mouvement. L'analyse étant relativement rapide, Mouvement standard peut être utilisé pour créer des animations de qualité professionnelle basées sur les lois de la physique.

Par exemple, pour créer une simulation reposant sur les calculs de Mouvement standard :

1. Sélectionnez Mouvement standard dans la liste Type d'étude du Motion Manager.

Les types d'études de mouvements disponibles sont Animation, Mouvement standard, et Analyse de mouvement.

2. Sélectionnez des outils dans la barre d'outils Motion Manager pour inclure des éléments de simulation. 












3. Cliquez sur Calculer  (barre d'outils Motion Manager) pour lancer les calculs relatifs à la simulation.
4. Cliquez sur Lecture du début  (barre d'outils Motion Manager) pour lire la simulation du début.

Le tableau 4.1 suivant décrit les outils disponibles dans les études de mouvement. Un (X) dans une colonne indique que l'outil est disponible pour ce type d'étude de mouvement.

### 4.3) Eléments mécaniques pour la mise en œuvre d'une étude mouvement

L'utilisateur de SolidWorks Motion, dispose d'un ensemble d'outils pour mener à bien une modélisation précise des actions mécaniques appliquées sur un mécanisme. Les types d'éléments de force suivants sont disponibles dans les Etudes de mouvement :







Tableau 4.1. Principaux outils de Motion Manager

Outil	Fonction	Animation	Mouvement standard	Analyse de mouvement
	<b>Calculer.</b> Calcule la simulation en cours. Si vous modifiez la simulation, elle doit être recalculée avant la relecture.	X	X	X
	<b>Lecture du début.</b> Réinitialise les composants et lit la simulation. A utiliser une fois la simulation calculée.	X	X	X
	<b>Lecture.</b> Lit la simulation en commençant à l'emplacement actuel de la barre de temps.	X	X	X
	<b>Arrêter.</b>	X	X	X
	<b>Mode de lecture</b> Mode de lecture	X	X	X
	<b>Normal.</b> Lecture du début à la fin, une fois.	X	X	X
	<b>En continu.</b> Lecture en boucle.	X	X	X
	<b>Va-et-vient.</b> Lecture en continu, du début à la fin, puis de la fin au début.	X	X	X
	<b>Enregistrer l'animation</b>	X	X	X
	<b>Assistant d'animation</b>	X	X	X
	<b>Clé automatique.</b> Cliquez sur cette option pour automatiquement placer une nouvelle clé lorsque vous déplacez ou modifiez des composants. Cliquez à nouveau pour la désactiver.	X		
	<b>Ajouter/Mettre à jour une clé.</b> Cliquez pour ajouter une nouvelle clé ou actualiser les propriétés d'une clé existante.	X	X	X

- **Gravité.**
- **Ressorts**, amortisseurs, frottement et plots élastiques. Les forces s'appliquent aux ressorts en translation, ressorts de torsion, amortisseurs en translation, amortisseurs de torsion, frottement statique, frottement dynamique, frottement dans les liaisons et plots élastiques.
- **Forces.** Les forces appliquées dans SOLIDWORKS Motion définissent les chargements de certaines pièces afin de les faire bouger dans certaines directions. Vous devez sélectionner le type de Fonction de force et les valeurs des paramètres associés. Vous pouvez aussi utiliser des expressions mathématiques. Vous pouvez appliquer des combinaisons de Force linéaire ou Couple.
- **Contact.** Des forces sont générées entre les composants en contact ou bien un contact continu est imposé aux composants. Définissez un contact pour empêcher les composants de se traverser pendant l'analyse de mouvement.


En règle générale, les éléments de force n'empêchent pas et ne prescrivent un mouvement imposé par un moteur. De ce fait, ils n'ajoutent pas de degrés de liberté au modèle étudié et n'en suppriment pas non plus. Les forces peuvent résister au mouvement ou elles peuvent l'induire.

Tableau 4.2. Principaux éléments de force dans SolidWorks motion




	Moteurs
	Gravité (Analyse de mouvement et Mouvement standard uniquement)
	Ressorts (Analyse de mouvement et Mouvement standard uniquement)
	Amortisseurs (Analyse de mouvement uniquement)
	Forces (Analyse de mouvement uniquement)
	Contact (Mouvement standard et Analyse de mouvement uniquement)

### 4.3.1) Les Moteurs :

L'utilisation de moteurs dans les études de mouvement vous permet d'appliquer un mouvement à un composant, sans prendre en compte la masse ou l'inertie. Le mouvement généré par un moteur élimine le mouvement généré par tout autre élément d'étude de mouvement. Tout élément d'étude de mouvement qui tend à résister au mouvement du moteur augmente la consommation de puissance de dernier, mais ne le ralentit pas. Cet effet peut par exemple être observé dans les résultats de l'analyse de mouvement. Cependant, si la référence de direction du moteur change pour une raison quelconque, le mouvement du moteur est appliqué dans la nouvelle direction. Vous pouvez sélectionner le mouvement du moteur à partir différents types de moteur définis ou utiliser des expressions mathématiques. En conséquence, les moteurs sont des éléments d'étude de mouvement qui déplacent les composants dans un assemblage en simulant les effets de différents types de moteurs. Pour ouvrir le **PropertyManager Moteur** il faut, pour un assemblage donné, cliquez sur **Moteur**




 (barre d'outils Motion Manager) et choisir par les éléments suivants :

#### Type de moteur




	<b>Moteur circulaire</b>	Spécifie un moteur circulaire.
	<b>Moteur linéaire (actionneur)</b>	Spécifie un moteur linéaire.
	<b>Moteur avec contrainte de trajectoire</b> (Analyse de mouvement uniquement)	Pour une contrainte de trajectoire donnée dans l'assemblage, spécifie le déplacement, la vitesse ou l'accélération pendant le déplacement d'un corps le long d'une trajectoire.

Il faut aussi choisir les bonnes contraintes et directions selon les indications suivantes

#### Contrainte/Direction








	<b>Contrainte de trajectoire</b>	Sélectionne la contrainte de trajectoire qui définit la trajectoire de mouvement.
	<b>Inverser la direction</b>	Bascule la direction du mouvement.
	<b>Composant</b>	Sélectionne le composant par rapport auquel le mouvement est appliqué.

## Composant/Direction

 <b>Emplacement du moteur</b>	Sélectionne la fonction au niveau de laquelle positionner le moteur.
 <b>Inverser la direction</b>	Bascule la direction du mouvement.
<b>Direction du moteur</b>	Sélectionne la fonction, telle qu'une face ou une arête, qui définit l'axe directionnel du moteur.
 <b>Composant</b>	Sélectionne le composant par rapport auquel le mouvement est appliqué.

Les propriétés du mouvement d'un moteur sont décrites selon la nomenclature suivante

## Mouvement

<b>Vitesse constante</b>		Spécifie un moteur à vitesse constante.
 <b>Vitesse</b>		Détermine la vitesse d'un moteur à vitesse constante.
<b>Distance</b>		Spécifie que le moteur est actionné sur une distance et une durée précises.
 <b>Déplacement</b>		Détermine la distance d'opération d'un moteur de type distance.
 <b>Heure de démarrage</b>		Détermine l'heure à laquelle le moteur de type distance est actionné.
 <b>Durée</b>		Détermine la durée d'opération d'un moteur de type distance.
<b>Oscillatoire</b>		Spécifie un profil de moteur oscillatoire.
 <b>Déplacement</b>		Définit l'amplitude du profil de décalage pour un moteur oscillatoire.
 <b>Fréquentielle</b>		Définit la fréquence du profil pour un moteur oscillatoire.
 <b>Déphasage</b>		Définit le déphasage du profil pour un moteur oscillatoire.
<b>Servomoteur</b>		Spécifie un servomoteur que vous pouvez utiliser dans les études de mouvement séquentielles.
	<b>Déplacement</b>	Spécifie un servomoteur de type déplacement.
	<b>Vitesse</b>	Spécifie un servomoteur de type vitesse.
	<b>Accélération</b>	Spécifie un servomoteur de type accélération.
<b>Segments</b>	Sélectionne le composant par rapport auquel le mouvement est appliqué.	Définit le profil comme fonctions continues du temps ou de l'angle du cycle.
<b>Points de données</b>		Définit le profil à partir d'un ensemble de données interpolées comme fonction du temps, de l'angle du cycle ou des résultats de l'étude de mouvement.
<b>Expression</b>		Définit le profil comme expression mathématique du temps, de l'angle du cycle ou des résultats de l'étude de mouvement.
<b>Charger une fonction à partir d'un fichier</b>		Permet d'importer une fonction créée avec le Créateur de fonctions à partir d'un fichier .sidfcn.
<b>Effacer des fonctions</b>		Permet de supprimer des fonctions créées avec le Créateur de fonctions.
<b>Modifier</b>		Permet de modifier le type de profil sélectionné.

Les moteurs déplacent les composants dans une direction donnée, mais ils ne constituent pas une force. La force des moteurs ne varie pas en fonction de la taille ou de la masse des composants. Par exemple, un petit cube se déplace à la même vitesse qu'un gros cube si la vitesse du moteur est réglée sur la même valeur dans les deux cas. Si le point de référence



de la direction du moteur change pour une raison quelconque, le moteur continue à déplacer le composant dans la nouvelle direction. Ceci peut arriver, par exemple, en cas de collision entre le composant déplacé par le moteur et un autre composant.

#### 4.4) Chronogramme

Situé à droite de l'arbre de création Motion Manager, il affiche la chronologie et le type d'événements d'animation dans l'étude de mouvement. Lorsque vous positionnez la barre de temps, déplacez les composants dans la zone graphique, ajoutez des éléments de simulation ou changez les propriétés visuelles, le chronogramme affiche les changements à l'aide de clés et de barres de changement.

Le chronogramme (Fig. 4.2) est divisé par des lignes de grille verticales correspondant aux marqueurs numériques montrant le temps. Les marqueurs numériques commencent à 00 :00 :00 et sont espacés en fonction de la taille de la fenêtre et du niveau de zoom.

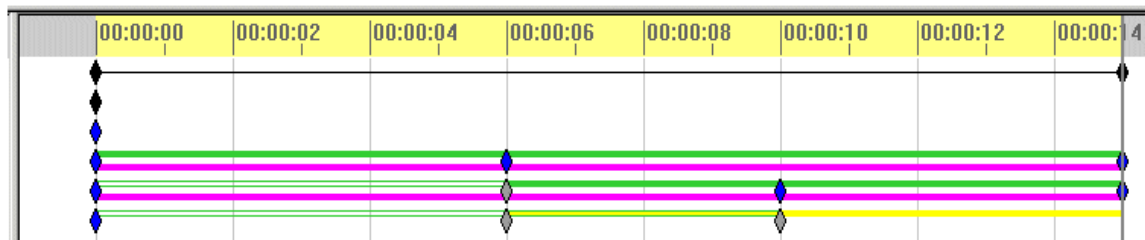


Figure 4.2. Les lignes d'un chronogramme

Pour afficher les composants à un moment donné, cliquez sur le moment correspondant dans le chronogramme. La barre de temps est alors réglée sur ce moment et un aperçu de la position des composants s'affiche.

- La ligne verticale pleine gris foncé qui se trouve sur le chronogramme est la barre de temps. Elle représente le temps courant.

Pour déplacer la barre de temps, effectuer une des actions suivantes :

- Faites-la glisser n'importe où le long du chronogramme.
- Cliquer n'importe où dans le chronogramme, sauf sur une clé.

Lorsque vous déplacez la barre de temps, vous changez le temps courant dans l'animation et mettez à jour le modèle.



- Les barres de changement sont les barres horizontales qui relient les clés. Elles indiquent un changement entre les clés.

Au nombre de ces changements se trouvent les suivants :

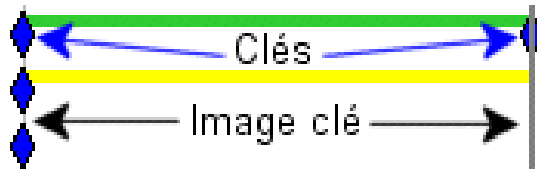
- Durée de l'animation

- Mouvement des composants
- Modifications des propriétés d'un élément de simulation
- Orientation de la vue, comme par exemple la rotation
- Propriétés visuelles telles que la couleur ou la vue

En fonction des entités, les barres de changement utilisent des couleurs qui identifient visuellement le composant et le type de changement.

- Les clés représentent le début ou la fin d'un changement de position ou d'attributs à un moment donné dans l'animation. Chaque fois que vous positionnez une nouvelle clé, celle-ci correspond à un mouvement ou à des changements des attributs visuels.

Les images clés définissent la portion du chronogramme qui sépare les clés.






Une clé est l'entité qui correspond aux positions des composants de l'assemblage, aux propriétés visuelles ou à l'état des éléments de simulation, tels qu'ils ont été définis.

L'image clé est la zone comprise entre des clés, qui peut être d'une durée quelconque. Elle définit la période sur laquelle le mouvement des composants de l'assemblage ou les modifications des propriétés visuelles interviennent.




#### 4.5) Arbre de création Motion Manager

L'arbre de création Motion Manager comprend :



- Orientation et vues de caméra Orientation et vues de caméra
- Le dossier Lumières, caméras et scène 
- Les entités de composants qui apparaissent dans l'arbre de création FeatureManager de SOLIDWORKS
- Les éléments de simulation que vous ajoutez (moteurs, forces ou ressorts)

Au cours d'une animation, vous pouvez faire pivoter, zoomer ou translater le modèle entier à l'aide des outils disponibles dans les menus contextuels pour Orientation et vues de caméra  . Vous pouvez également utiliser Orientation et vues de caméra  , pour définir la vue à travers une caméra que vous ajoutez au modèle. Cliquez à l'aide du bouton droit de la souris pour Orientation et vues de caméra activer ou désactiver ces options :

- Désactiver la lecture des clés de vue. Sélectionnez cette option pour que la vue du modèle ne change pas pendant la lecture ou l'édition de l'animation. L'icône change de forme en fonction de l'activation/désactivation de cette option :

- La lecture des clés de vue est activée. 
  - La lecture des clés de vue est désactivée .
- Désactiver la création de clés de vue. Sélectionnez cette option pour verrouiller l'animation spécifiée avec Orientation et vues de caméra . Les modifications apportées au modèle avec les outils de rotation, de zoom ou de translation ne sont pas enregistrées comme images clés dans l'animation.

Le mode **Clé automatique** est uniquement disponible avec Animation. Dans ce mode **Clé automatique**, Animation :

- Place automatiquement des clés, en fonction de la position de la barre de temps
  - Calcule l'animation de sorte que les composants atteignent les positions spécifiées
  - Calcule le mouvement des composants d'assemblage d'une position à l'autre en utilisant l'interpolation
- Lorsque vous sélectionnez **Clé automatique** , le Motion Manager place des clés automatiquement. Ces dernières sont placées sur les moments correspondant aux actions réalisées sur le modèle.
- Lorsque vous désactivez **Clé automatique** , vous devez placer les clés manuellement.

#### 4.6) Contraintes dans les études de mouvement

D'une façon générale, on dispose dans SolidWorks de trois familles de contraintes : Standards, avancées et mécaniques, elles sont regroupées dans la figure 4.3.

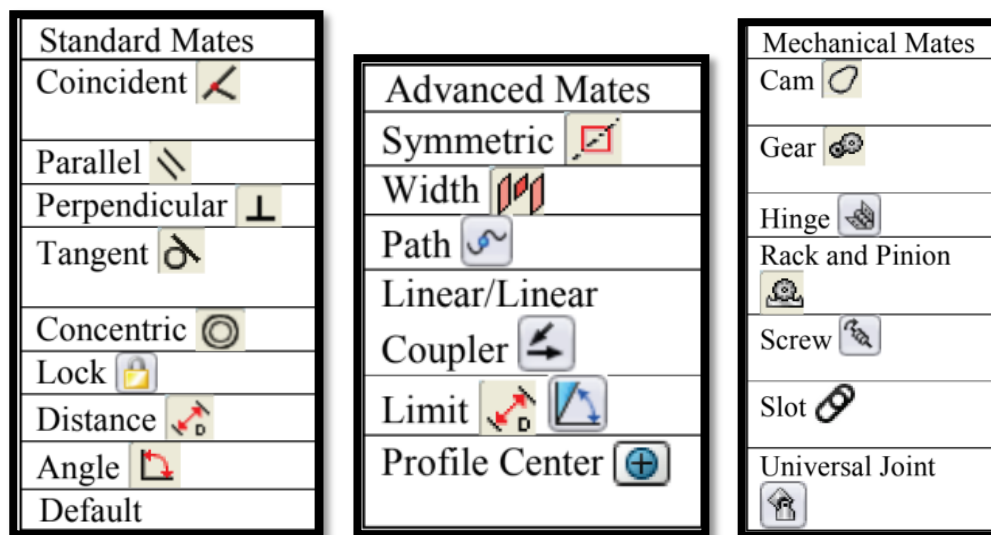
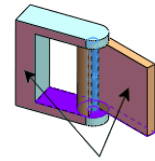


Figure 4.3. Ensemble de contraintes disponibles dans SolidWorks.

Elles sont utilisées pour restreindre le mouvement entre des composants dans les études de mouvement. On peut définir les valeurs pour les contraintes de distance et d'angle et modifier ces valeurs à différents points dans l'animation. Par ailleurs, il faut faire attention au

problème de l'hyperstaticité. Dans les études d'analyse de mouvement, l'hyperstaticité équivaut à sur-contraindre un modèle SOLIDWORKS.

Par exemple, pour créer des contraintes mécaniques spécifiques à une étude de mouvement il faut d'abord sélectionner une étude de mouvement puis, ajouter des contraintes à l'assemblage à partir de l'étude de mouvement :



1. **Contrainte**  (barre d'outils Assemblage) ou sur **Insertion > Contrainte**.



2. Dans le PropertyManager **Contraintes mécaniques**, choisir la bonne contrainte


Les contraintes spécifiques à l'étude de mouvement sont uniquement répertoriées dans l'arbre de création de **MotionManager**. La chaîne Local précède le nom des contraintes spécifiques à une étude de mouvement. Lorsque les contraintes sont regroupées par état, les contraintes spécifiques à l'étude de mouvement ne sont pas incluses dans les dossiers des états. Ces contraintes sont répertoriées individuellement sous le dossier Contraintes dans l'arbre de création MotionManager. A titre indicatif, nous expliquons comment réaliser une contrainte pivot et une Contrainte de trajectoire dans les études de mouvement.

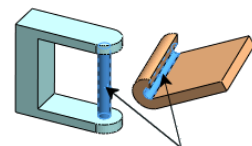
On va maintenant illustrer comment imposer une contrainte pivot


#### 4.6.1) Contrainte pivot

Une **contrainte pivot** limite le mouvement entre deux composants à un degré de liberté en rotation. Elle revient à ajouter une **contrainte coaxiale** et une **contrainte coïncidente**. On peut également limiter le mouvement angulaire entre les deux composants. **Pour ajouter une contrainte de pivot :**

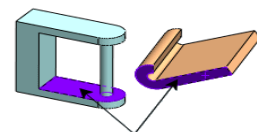
- Cliquez sur **Contrainte**  (barre d'outils Assemblage) ou sur **Insertion > Contrainte**.
- Dans le PropertyManager **Contraintes mécaniques**, sous **Pivot**, cliquez sur .
- Faites vos sélections et définissez les options sous **Sélection des contraintes** :

- **Sélections coaxiales**  . Sélectionnez deux entités. Les sélections valides sont les mêmes que pour les contraintes coaxiales.



- **Sélections coïncidentes**  . Sélectionnez deux entités. Les sélections valides incluent un plan ou une face plane et:

- Un plan ou une face plane
- Arête
- Point



- **Spécifier les limites d'angle.** Sélectionnez cette option pour limiter la rotation angulaire entre les deux pièces. Définissez l'étendue de la rotation :

- **Sélection d'angles** . Sélectionnez deux faces.

- **Angle.** Indiquez l'angle nominal entre les deux faces.

- **Valeur maximale** 


- **Valeur minimale** 

3. Cliquez sur .

**Pivot**  apparaît dans le dossier Contraintes dans l'arbre de création FeatureManager.

#### 4.6.2) Contraintes de trajectoire dans les études de mouvement





On peut aussi inclure des contraintes de trajectoire dans les études **d'Analyse de mouvement** comme suit

1. Sélectionner **Insertion > Contrainte** .
2. Définissez une contrainte de trajectoire pour les composants.
3. Sélectionner l'onglet Etude de mouvement, puis choisir **Analyse de mouvement** sous **Type d'étude**.
4. Inclure, si nécessaire, des éléments d'études de mouvement.

Vous devez inclure un **Moteur** , une **Force**  ou une **Gravité**  pour forcer le mouvement.

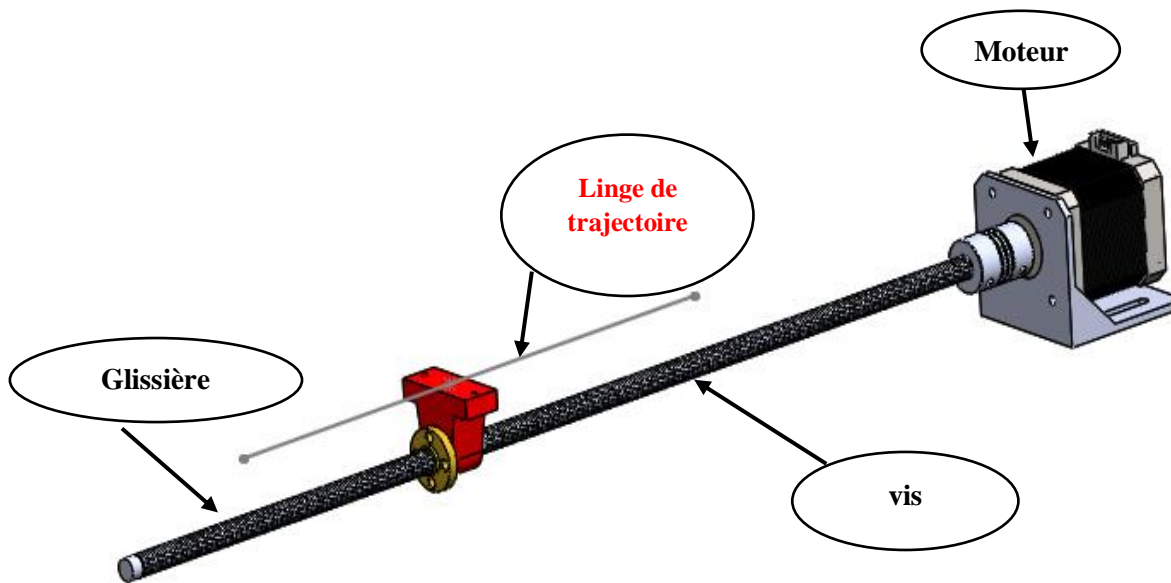
5. Exécuter l'étude.

Pour les études **d'Analyse de mouvement**, on peut définir un moteur avec contrainte de trajectoire afin d'indiquer les valeurs de déplacement, de vitesse ou d'accélération pendant le déplacement d'un corps le long d'une trajectoire. Voici comment il faut le faire

1. A partir d'une étude **d'Analyse de mouvement**, cliquez sur **Moteur** .
2. Pour **Contrainte de trajectoire** , sous **Contraintes** dans l'arbre de création FeatureManager, sélectionnez une **contrainte de trajectoire**.
3. Sélectionnez le profil du **moteur**.
  - **Pour définir le profil du moteur à partir d'une fonction ou à partir de données, cliquez sur Expression, Points de données ou Segments.**
4. Sélectionnez des options et cliquez sur .
5. **Pour inverser la direction du moteur, pour Contraintes/Direction, cliquez sur Inverser la direction** .

## 4.7) Application au cas de notre machine CNC




### 4.7.1) Simulation du sous-ensemble Moteur / glissière



**Figure 4.4.** Sous-Ensemble Moteur-vis-écrou

La cinématique de la machine CNC qu'on désire réaliser est tributaire du bon fonctionnement du sous-ensemble *Moteur-vis-écrou*. En effet, il y en a trois sous-ensembles dans la machine de ce type-là. L'étude de ce mécanisme est donc importante pour déterminer la loi d'entrée-sortie des sous-ensembles de la machine et éventuellement estimer la puissance des moteurs à utiliser pour atteindre des performances bien ciblées.

Comme étude préliminaire, nous donnerons au moteur couplé avec la vis, une certaine vitesse de rotation pour voir comment l'écrou glissant se déplace. Étant donné que la vis est connectée directement au moteur, la vitesse ou le mouvement de rotation que nous donnons au moteur sera la même pour la vis. En outre, nous suivrons les étapes suivantes :

- 1) Sélectionnez Insertion > Contrainte  après sélectionnez contrainte mécanique.
  - 2) Cliquez sur l'icône de la vis hélicoïdale .
  - 3) Sélectionnez une face du filetage de l'écrou et une face du filetage de la vis.
- Après avoir appliqué la contrainte, nous allons commencer l'étude de mouvement comme suit
- 4) Sélectionnez le profil du moteur.
  - 5) Sélectionnez le moteur circulaire et sélectionnez le profil selon lequel il doit tourner.
  - 6) Cliquez sur Calculer  de la barre d'outils Motion Manager pour effectuer les calculs nécessaires à la simulation ;

Ensuite, après la fin des calculs, nous allons afficher les résultats et en utilisant différents graphes.

Il est possible de construire le profil de mouvement qu'on désire en utilisant l'outil créateur de fonction comme indiqué sur la (figure 4.5)

The image shows a software interface for creating a motion profile for a motor. It is divided into several sections:

- 1) sélectionner les contraintes**: A panel on the left titled "Sélection des contraintes" with a red star labeled '3'. It lists various constraints: "Face <1> @vis a chonfen (YZ)", "Face <2> @LeadScrew Nut 8r", "Contraintes standard", "Contraintes avancées", and "Contraintes mécaniques". Under "Contraintes mécaniques", there are icons for "Came", "Rainure", "Pivot", "Engrenage", "Pignon crémaillère", and "Hélicoidale" (marked with a red star labeled '2').
- 4) Choix du moteur**: A panel on the right titled "Type de moteur" with a red star labeled '5'. It offers three options: "Moteur circulaire", "Moteur linéaire (actionneur)", and "Moteur avec contrainte de trajectoire". Below this is the "Composant/Direction" section with a blue bar and a red star labeled '5'. The "Mouvement" section has a dropdown set to "Vitesse constante" and a value of "100 RPM". A red star labeled '5' is also present here. A callout box on the right says "Sélectionner le profil du mouvement de rotation". Another callout box below it says "Sélectionner segment".
- Créateur de fonctions**: A window at the bottom titled "Créateur de fonctions" with a red star labeled '2'. It has tabs for "Segments", "Points de données", and "Expression". The "Segments" tab is active, showing a table with columns "X initial", "X final", "Valeur", and "Type de saigne". The table contains three rows:
 

X initial	X final	Valeur	Type de saigne
0s	2s	0.00deg/s	Linear (Defa)
2s	4s	100.00deg/s	Linear (Defa)
4s	5s	0.00deg/s	Linear (Defa)

 A callout box below the table says "Définir les valeurs numériques pour le profil de mouvement sélectionné". To the right of the table are three graphs: "Vitesse (deg/s)" vs "Temps (s)", "Accélération (deg/s^2)" vs "Temps (s)", and "Jerk (deg/s^3)" vs "Temps (s)". The top graph shows a trapezoidal velocity profile. The bottom-left graph shows a step function for acceleration. The bottom-right graph shows a step function for jerk.

Figure 4.5. Différentes étapes suivies pour effectuer la simulation du sous-ensemble Moteur-Vis-Ecrou

A l'issue de la simulation, on peut afficher par le biais de graphes différents résultats. Par exemple on a :

- **Graphe 01** de la figure 4.6 représente l'évolution temporelle du déplacement linéaire de l'écrou (on a choisi un point sur l'écrou)
- **Graphe 02** de la figure 4.7 représente l'évolution temporelle du déplacement angulaire de la vis

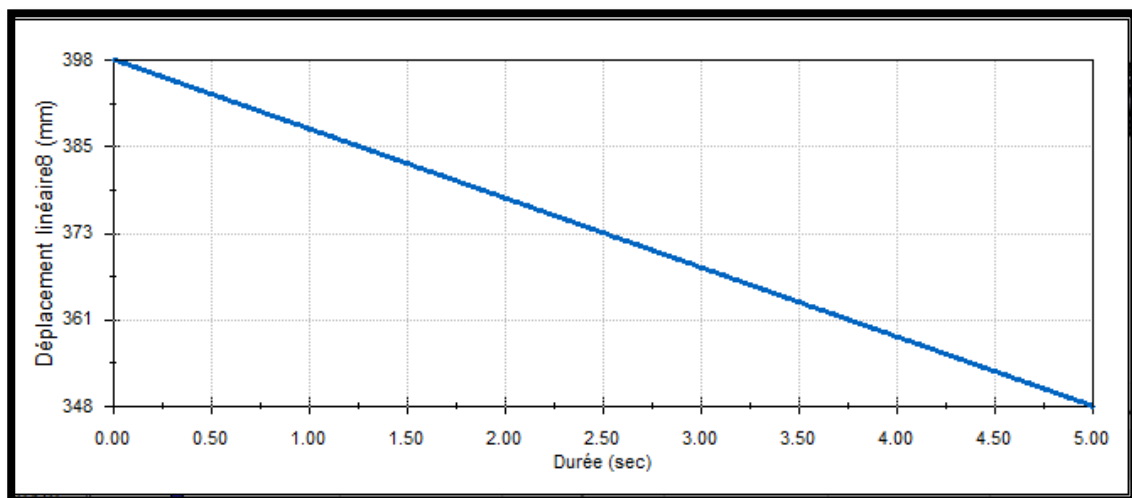


Figure 4.6. L'évolution temporelle du déplacement de l'écrou de la vis

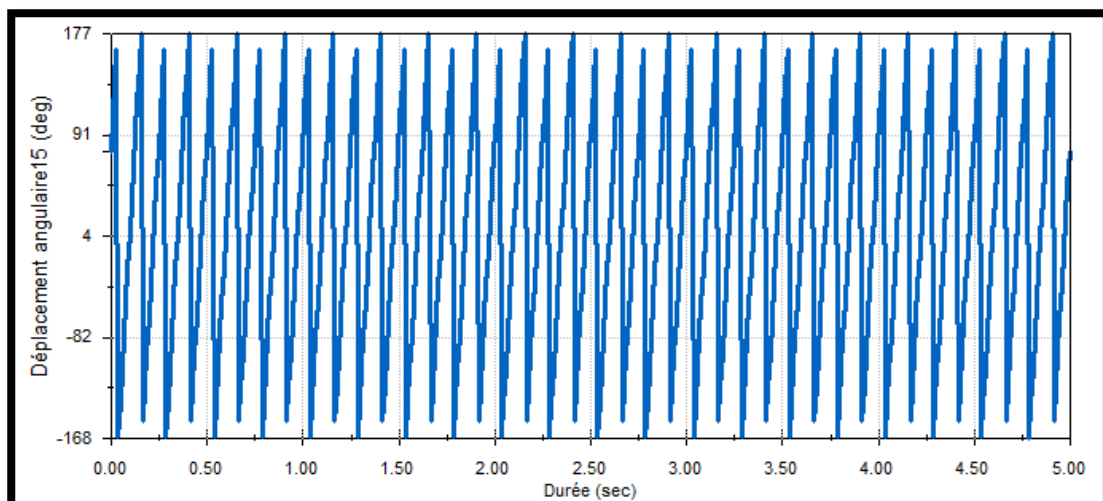


Figure 4.7. L'évolution temporelle du déplacement angulaire de la vis



### 4.7.2) Simulation du sous-ensemble Table-porte-pièce

La simulation de mouvement du sous ensemble table-porte-pièce permet de vérifier que la puissance du moteur est suffisante.

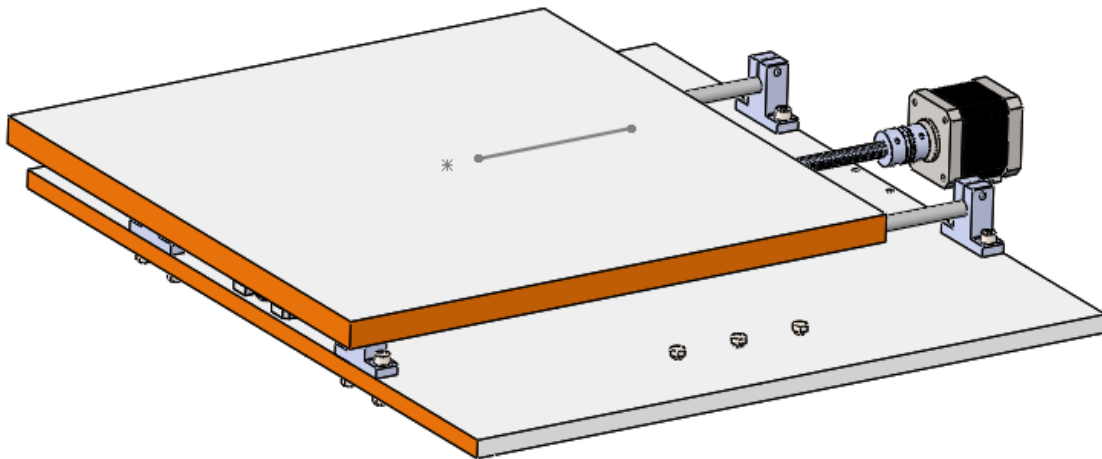


Figure 4.8. Sous-Ensemble Moteur-vis-écrou-table

Pour la mise en œuvre de cette simulation, on dessine d’abord une ligne qui est appliquée comme contrainte de trajectoire. La longueur de la ligne est  $L=130\text{mm}$ . Puis on applique les caractéristiques de vitesse appropriées. Nous avons essayé différentes vitesses et selon différents profils à l’aide de l’outil de la figure 4.9. ce qui nous permet d’estimer la puissance moteur comme indiqué sur la figure 4.10.

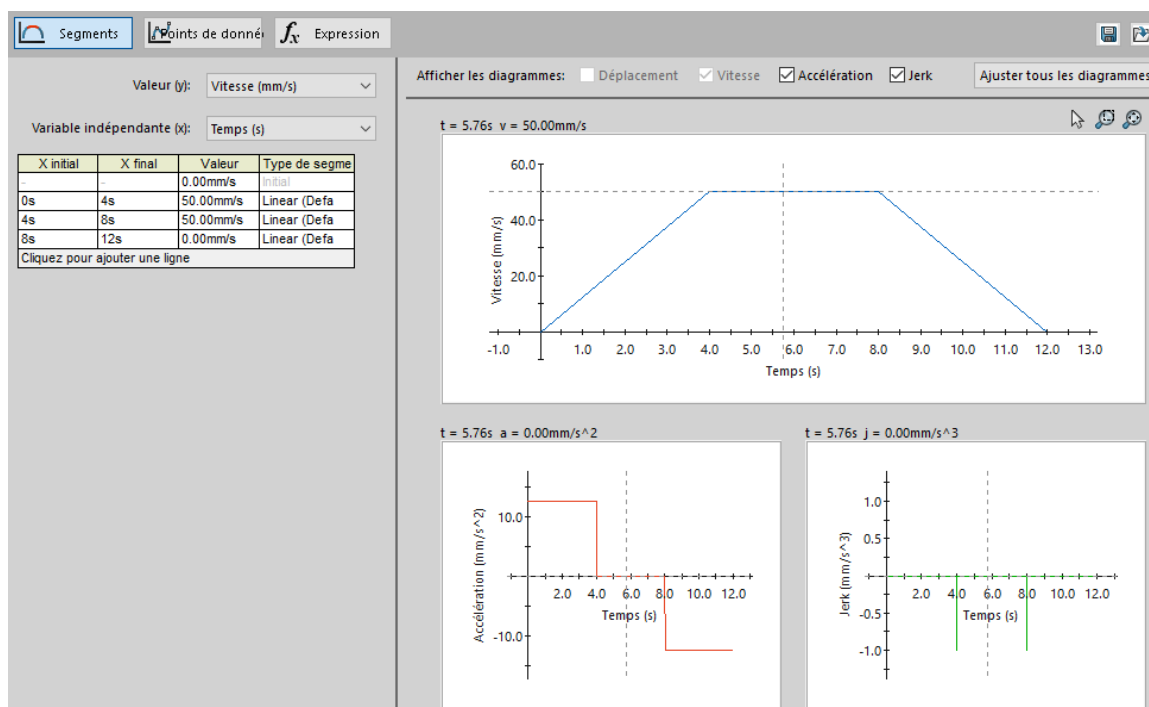


Figure 4.9. Caractéristique de la vitesse imposée (profil trapézoïdal)

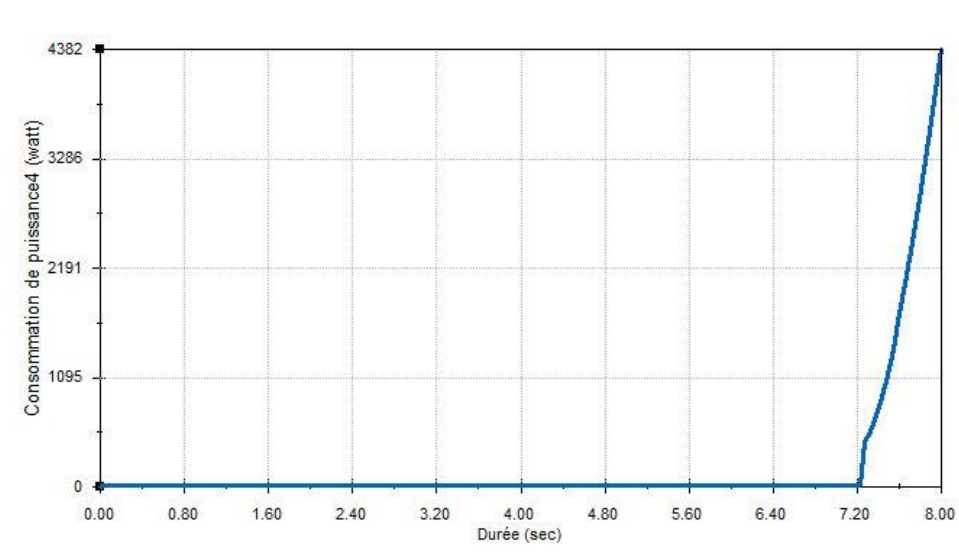


Figure 4.10. La puissance du moteur

### 4.7.3) Simulation du sous-ensemble robot parallèle

Dans ce paragraphe, nous allons décrire l'étude de mouvement réalisée pour la partie robot parallèle. Pour cette analyse de mouvement, allons utiliser un moteur avec une contrainte de trajectoire afin d'imposer les valeurs de déplacement, de vitesse ou d'accélération pendant le déplacement d'un corps le long d'une trajectoire.

A ce titre, nous allons étudier le mouvement des pièces importantes (**Vis, nacelle, écrou**) comme indiqué sur la figure 4.11. On va voir comment imposer une trajectoire circulaire à suivre selon une cinématique bien déterminée. Avant cela il faut activer l'analyse en cliquant sur le bouton SolidWorks motion (1) de la barre du menu et activer l'étude de mouvement en appuyant sur l'onglet correspondant (2).

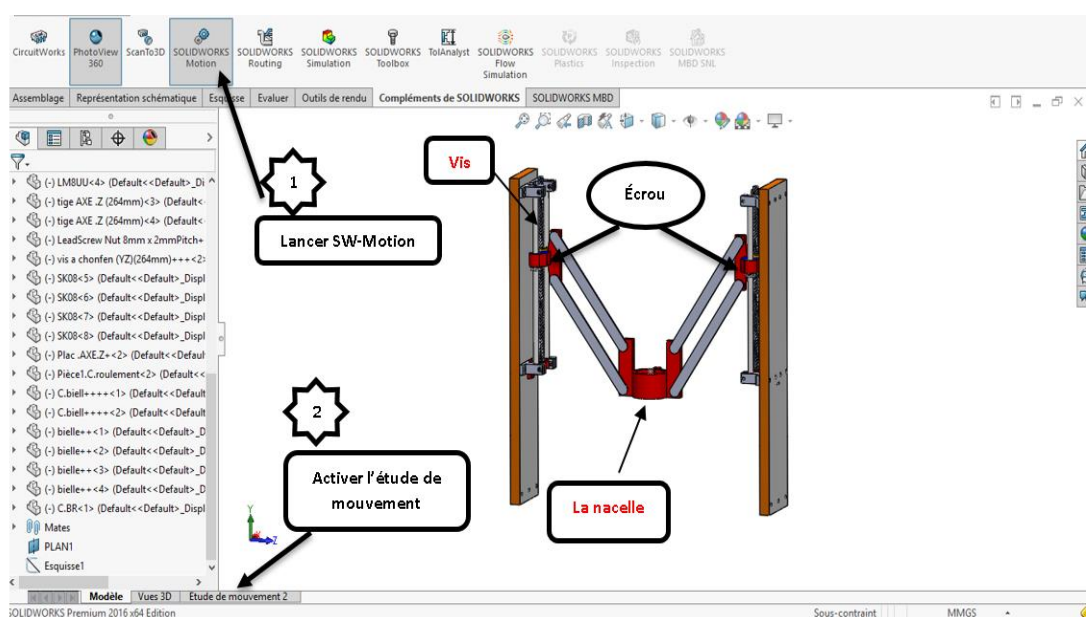


Figure 4.11. Assemblage étudié lors du suivi d'une trajectoire.

Par la suite il faut réaliser les étapes suivantes (figure 4.12) :

- Choisir Analyse de mouvement sous Type d'étude (3).
- Spécifier le type de Moteur à utiliser (4).

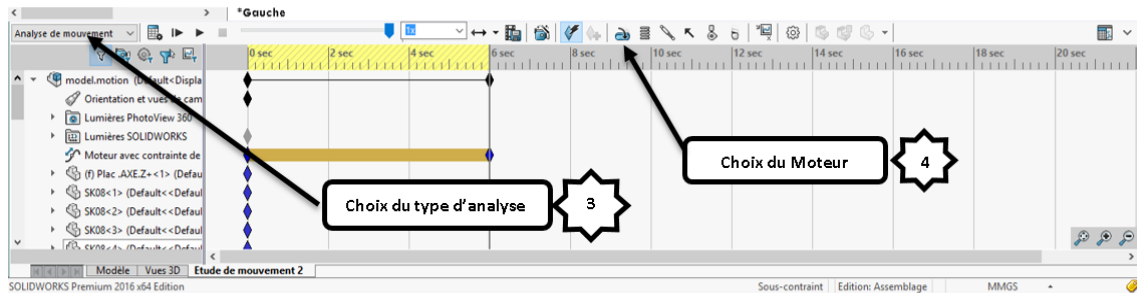


Figure 4.12. choix du type d'analyse et du moteur à utiliser

Pour contraindre le mouvement des composants de l'assemblage sur une trajectoire nous avons suivi les étapes suivantes (Fig.4.13) :

1. Nous traçons la trajectoire à suivre.
2. Nous dessinons un point sur la pièce que nous voulons faire suivre la trajectoire que nous avons tracé plus tôt. Pour cet exemple, on va dessiner le point sur la nacelle.

**Remarque :** attention la **trajectoire** et le **point** doivent être dans le même **PLAN**, c'est-à-dire que le plan de la trajectoire est **parallèle** et la distance avec le plan du point est inexistante (**zéro**).

Nous avons tracé un cercle de diamètre 30mm

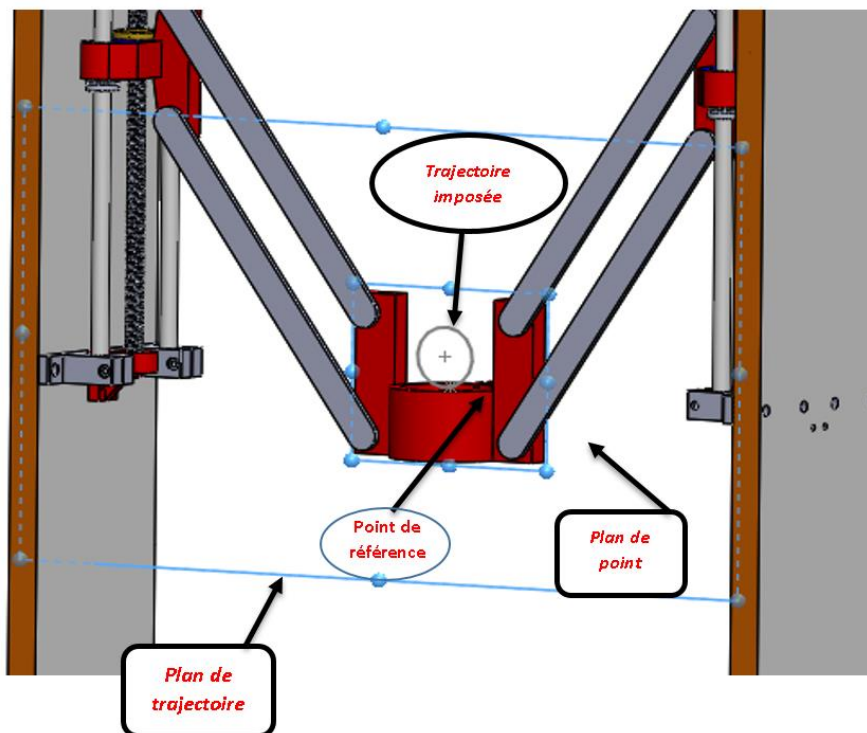
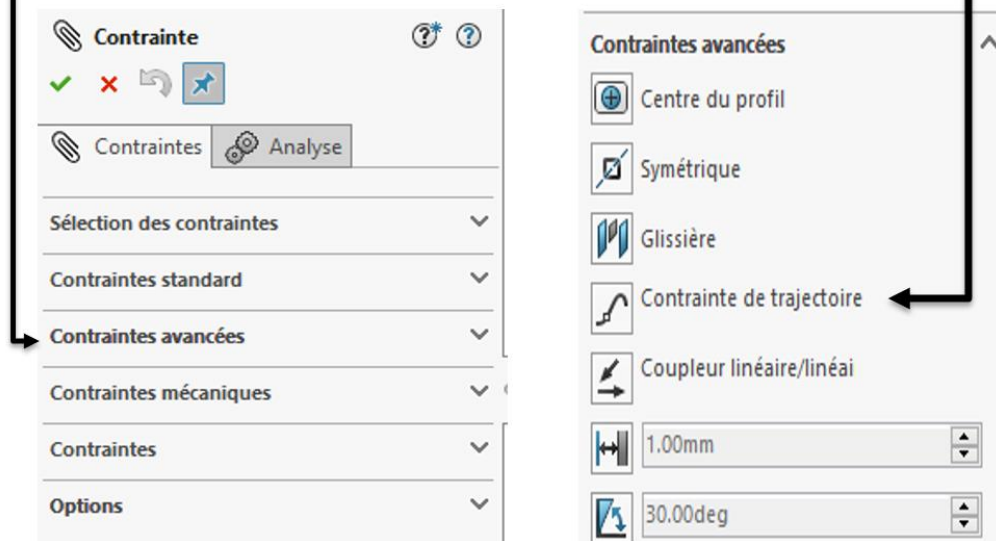


Figure 4.13a. Trajectoire circulaire imposée à la nacelle

1. Sélectionnez Insertion > Contrainte  après sélectionnez contrainte avancer.
2. Définissez une contrainte de trajectoire pour les composants.

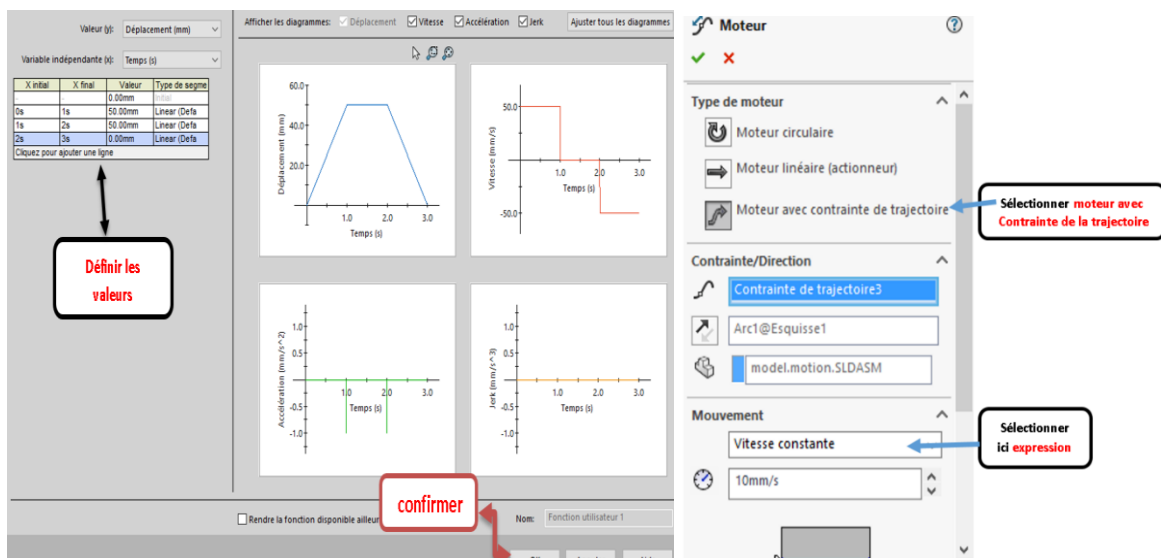


**Figure 4.13b.** Étapes suivies pour imposer le suivi d'une trajectoire circulaire par un point de référence pris sur la nacelle

Pour **Contrainte de trajectoire**, sous **Contraintes** dans l'arbre de création **FeatureManager**, sélectionner une contrainte de trajectoire.

1. Sélectionner le profil du moteur.
2. Pour définir le profil du moteur à partir d'une fonction ou à partir de données, cliquer sur Expression, Points de données ou Segments.
3. Sélectionnez des options et cliquez sur.

Pour inverser la direction du moteur, pour Contraintes/Direction, cliquez sur Inverser la direction



**Figure 4.14.** Étapes suivies pour définir le profil de mouvement sur la trajectoire imposée

Après avoir terminé ces étapes, nous allons directement lancer la simulation pour obtenir les résultats (Figure 4.15). Pour cela, il faut Cliquer sur **calculer** et attendre que le programme termine les calculs. Ensuite, nous allons afficher les résultats et cliquer sur **résultats** en graphe. Après, nous demandons au programme quel genre de résultats que nous voulons afficher.

Les graphes affichés sont:

- 1) Graphe 01 : position/vitesse/ accélération d'un point de la nacelle
- 2) Graphe 02 : position/vitesse/ accélération d'un point des deux écrous q1/q2
- 3) Graphe 03 : position/vitesse/ accélération des moteurs,
- 4) Graphe 04 : évaluer les couples moteurs et la puissance consommée pour la trajectoire spécifiée

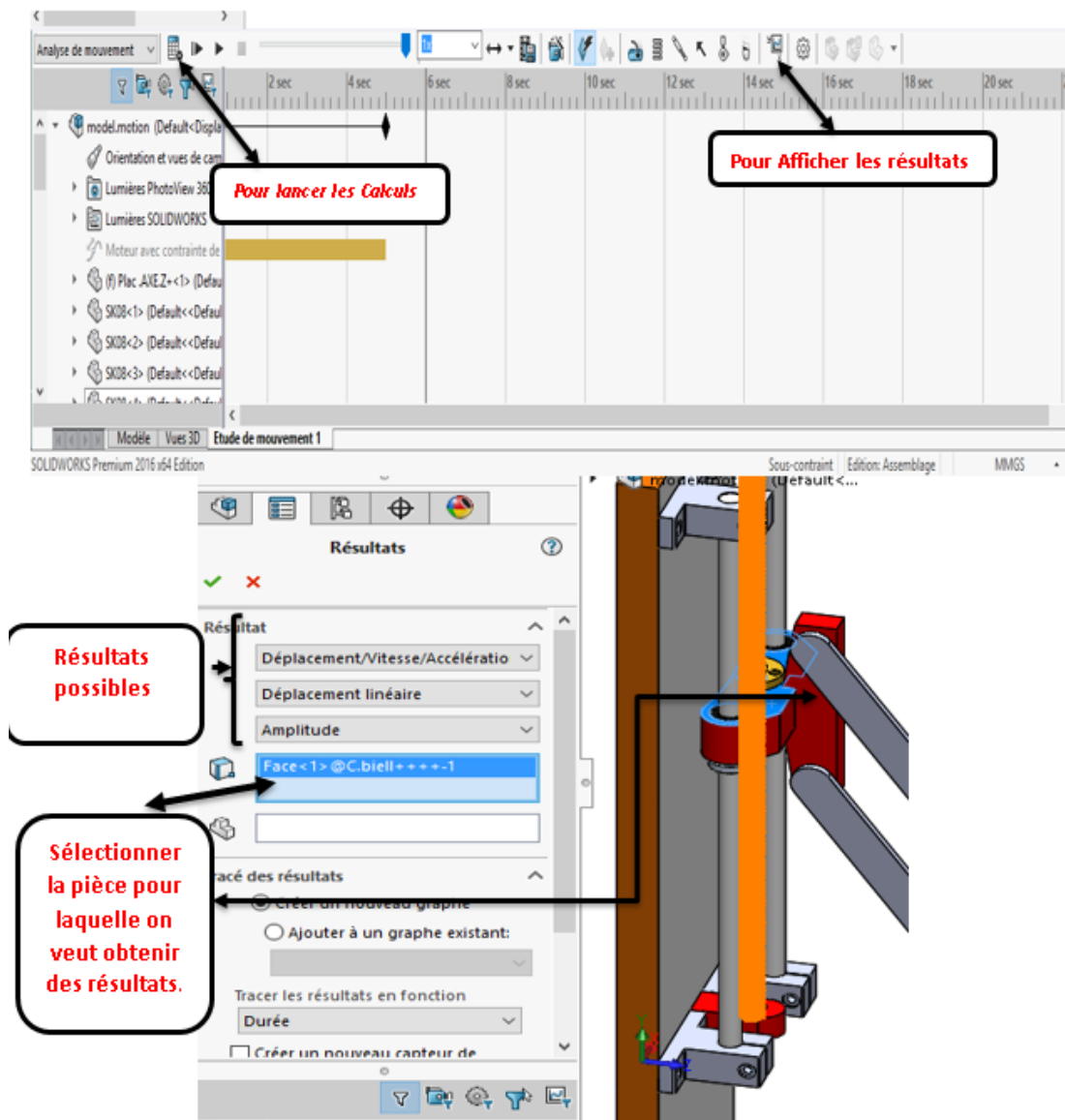


Figure 4.15. Étapes suivies pour afficher les résultats de simulation

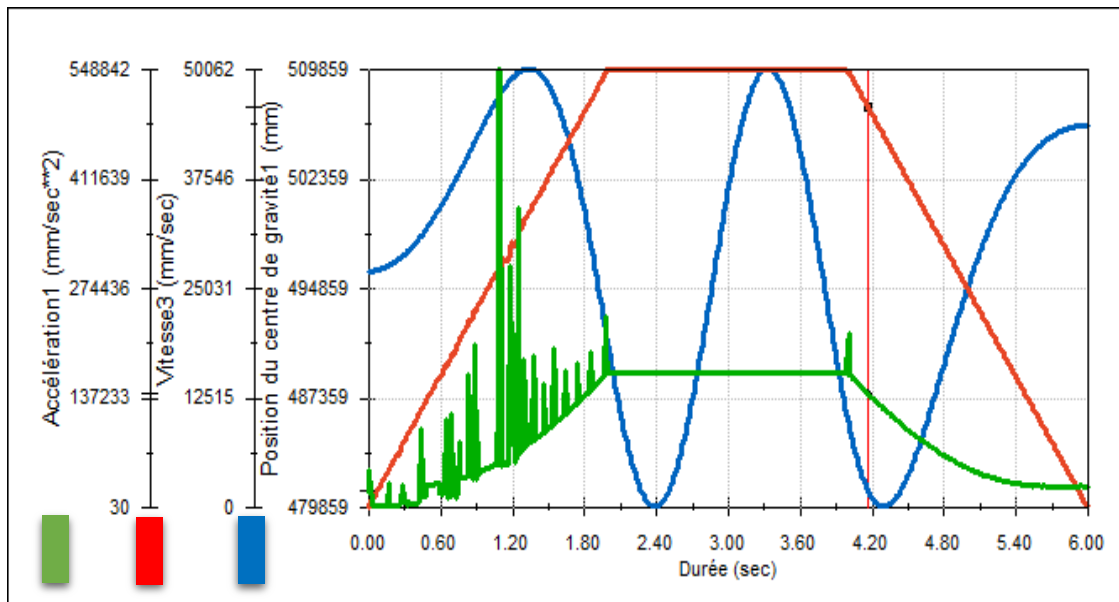


Figure 4.16. Graphe position/vitesse/ accélération d'un point de la nacelle

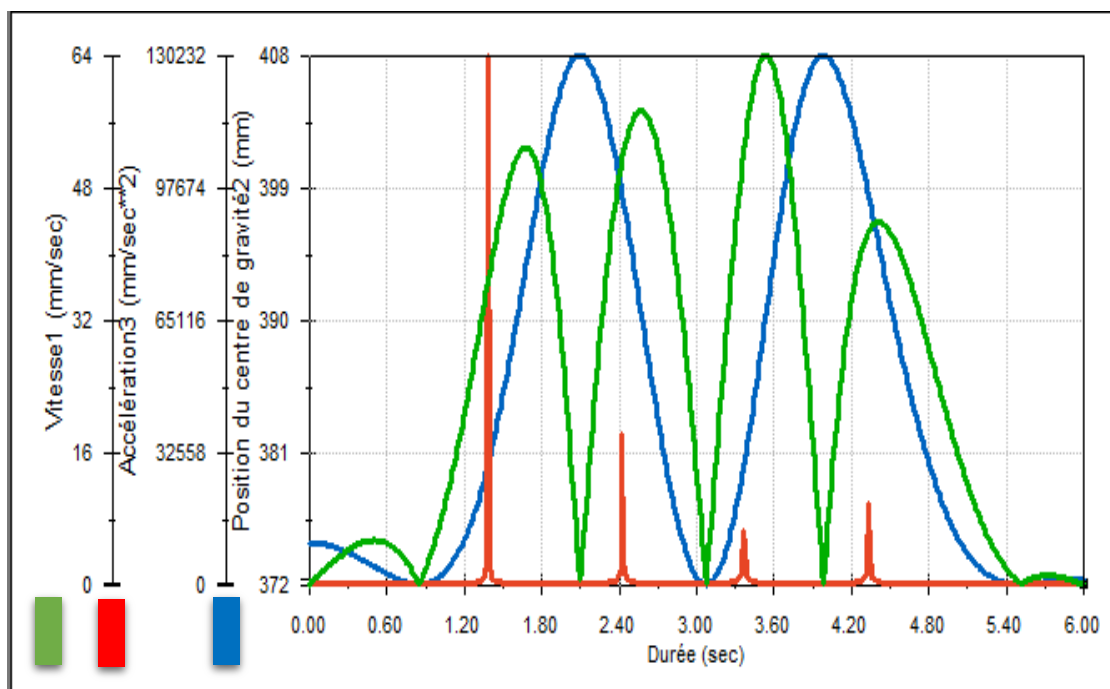


Figure 4.17. Graphe position/vitesse/ accélération d'un point de l'écrou 1 (q1)

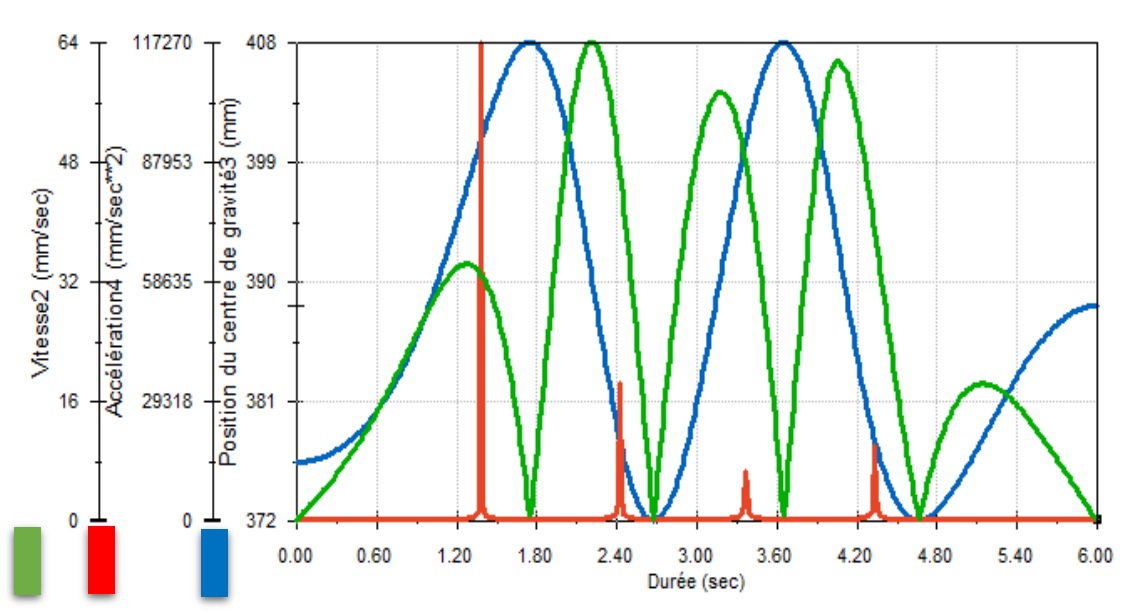


Figure 4.18. Graphe position/vitesse/ accélération d'un point de l'écrou 2 ( $q_2$ )

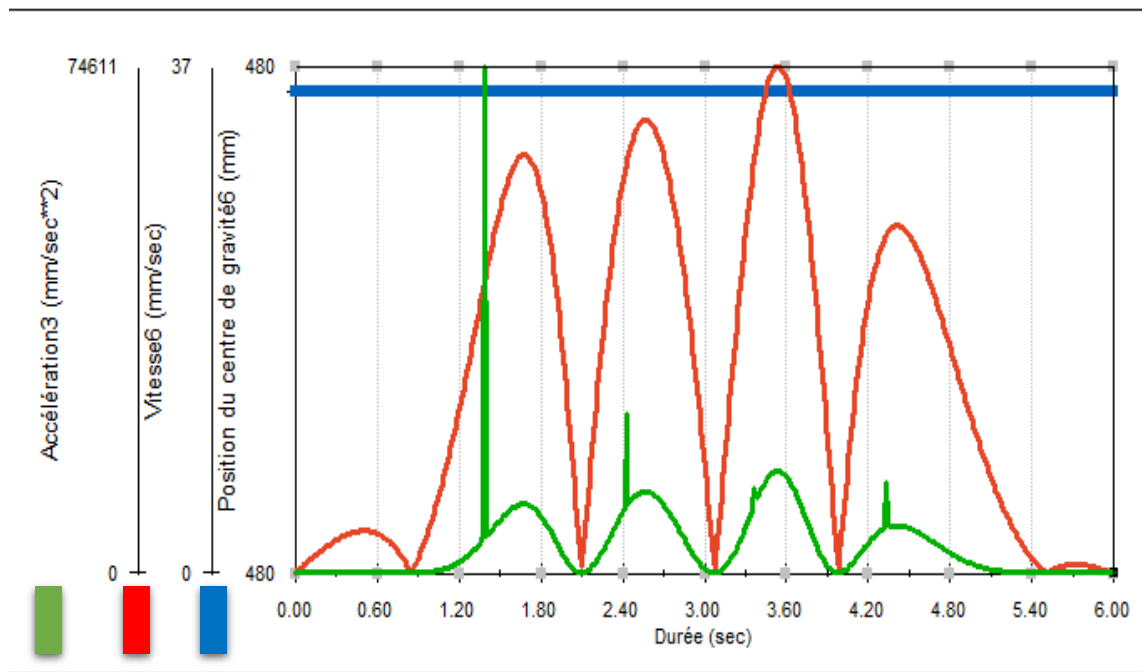


Figure 4.19. Graphe position/vitesse/ accélération d'un moteur

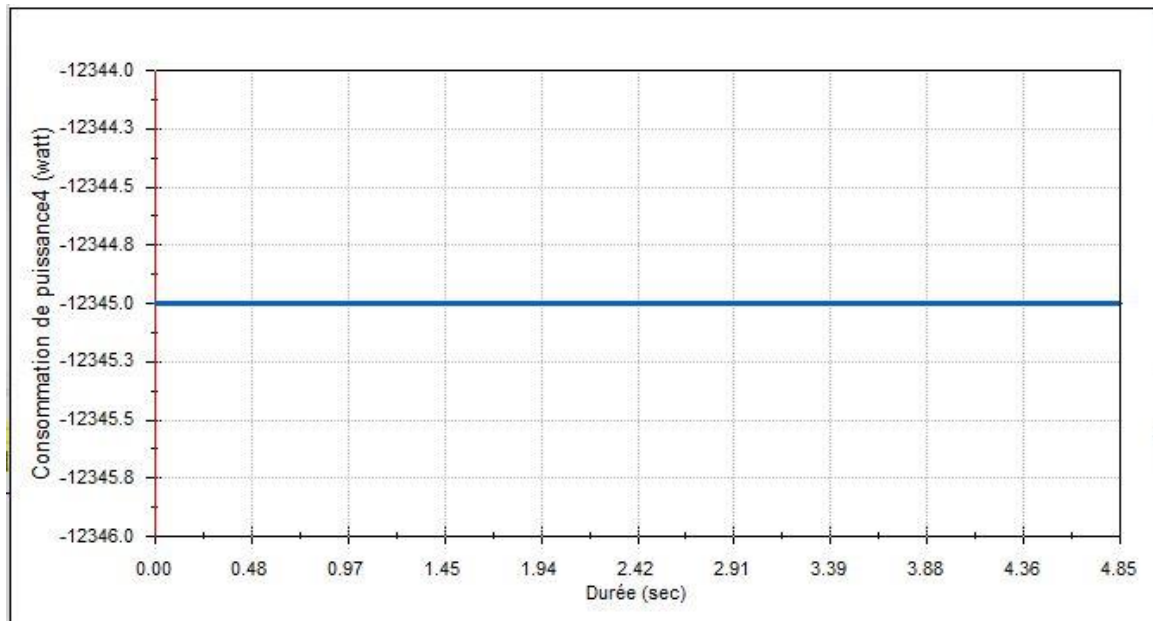


Figure 4.20 la puissance consommée pour la trajectoire spécifiée

#### 4.8) Comparaison des résultats de simulation Matlab avec celle SolidWorks motion

Nous allons prendre les résultats du programme Matlab que nous obtenons pour plusieurs trajectoires (cercle , ligne horizontale, ligne verticale ) et on les comparera avec les résultats du logiciel SolidWorks .

Le programme qu'on a réalisé sous Matlab est donné sur la figure 4.21. Les chemins obtenus sont illustrés sur la figure 4.22. Ces chemins sont ensuite exportés sous Excel où nous avons enregistré l'évolution des variables Y1 et Y2 pour les utiliser par la suite comme entrée de SolidWorks.

En effet, sous SolidWorks l'étude est menée d'une façon inverse. On a suivi les étapes suivantes (voir figures 4.23 et 4.24).

1. sélectionner moteur
2. clique sur moteur linéaire pour sélectionner le type
3. définir le mouvement :
  - 3.1.clique sur point donné
  - 3.2.importé le fichier Excel (SVG) se fichier Contient le déplacement Y1 (de même pour Y2)



```

Model2_PRRRPX.m x +
1 - clear all
2 - clc
3 - % Robot data
4 - H = 220; % Hauteur de la machine
5 - h = 30.5; % 1/2 Hauteur de la nacelle
6 - R = 146; % 1/2 largeur de la machine
7 - r = 41; % 1/2 largeur de la nacelle
8 - L = 185; % longueur des bras L>R
9 - N = 100; %Nbre de configuration à afficher
10
11 - px=[];
12 - py=[];
13 - for i=1:N;
14 -     % variables opérationnelles
15 -     % Mouvement vertical
16 -     % x = 0;
17 -     % y = -1*H*i/N
18
19 -     % Mouvement Horizontal
20 -     x = 50*i/N
21 -     y = -250.5
22
23 -     % Mouvement circulaire
24 -     % xc = 0;
25 -     % yc = -0.5*H;
26 -     % rc = 0.1*R;
34 -     % variables opérationnelles
35 -     % Mouvement vertical
36 -     % x = 0;
37 -     % y = -1*(70)*i/N-60;
38 -     % Mouvement Horizontal
39 -     x = 50*i/N
40 -     y = -250.5
41 -     % Mouvement circulaire
42 -     % xc = 0;
43 -     % yc = -0.5*H;
44 -     % rc = 0.1*R;
45 -     % x = xc+rc*cos(2*pi*i/N);
46 -     % y = yc+rc*sin(2*pi*i/N);
47 -     X1(i,1)=x ;
48 -     Y1(i,1)=y ;
49 -     fid=fopen('X1Y1.dat','r');
50 -     % variables articulaires MGI
51 -     sol= 1; % (on utilisera sol =1)
52 -     q = MGI_PRRRP(x,y,L,r,R,sol);
53 -     % y1 = q(1);
54 -     % y2 = q(2);
55 -     y1 = y+sqrt(L^2-((x+r)-R)^2);
56 -     y2 = y+sqrt(L^2-((x-r)+R)^2);
57 -     XX1(i,1)=y1;
58 -     YY1(i,1)=y2;
59 -     syms X Y;

```

Figure 4.21. Programme de dessin (cercle, linge H/V)

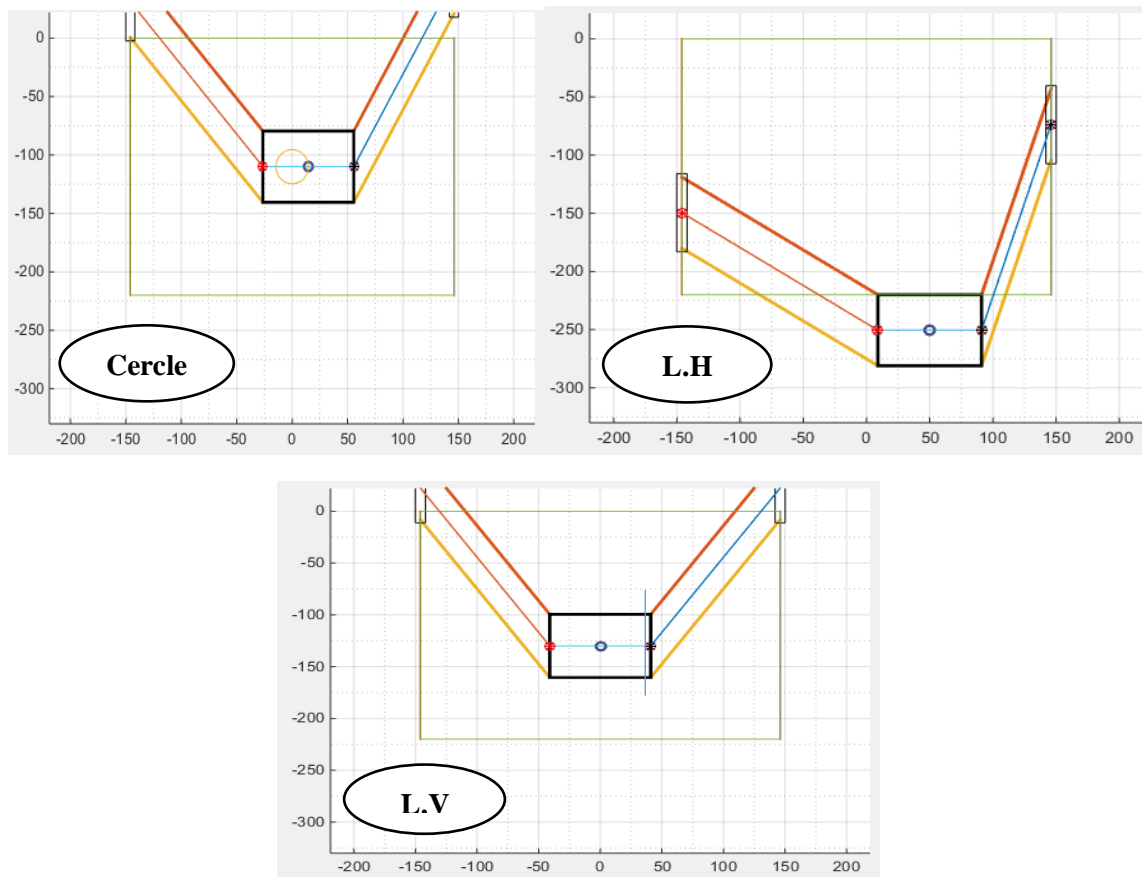


Figure 4.22. les chemins générés sous Matlab

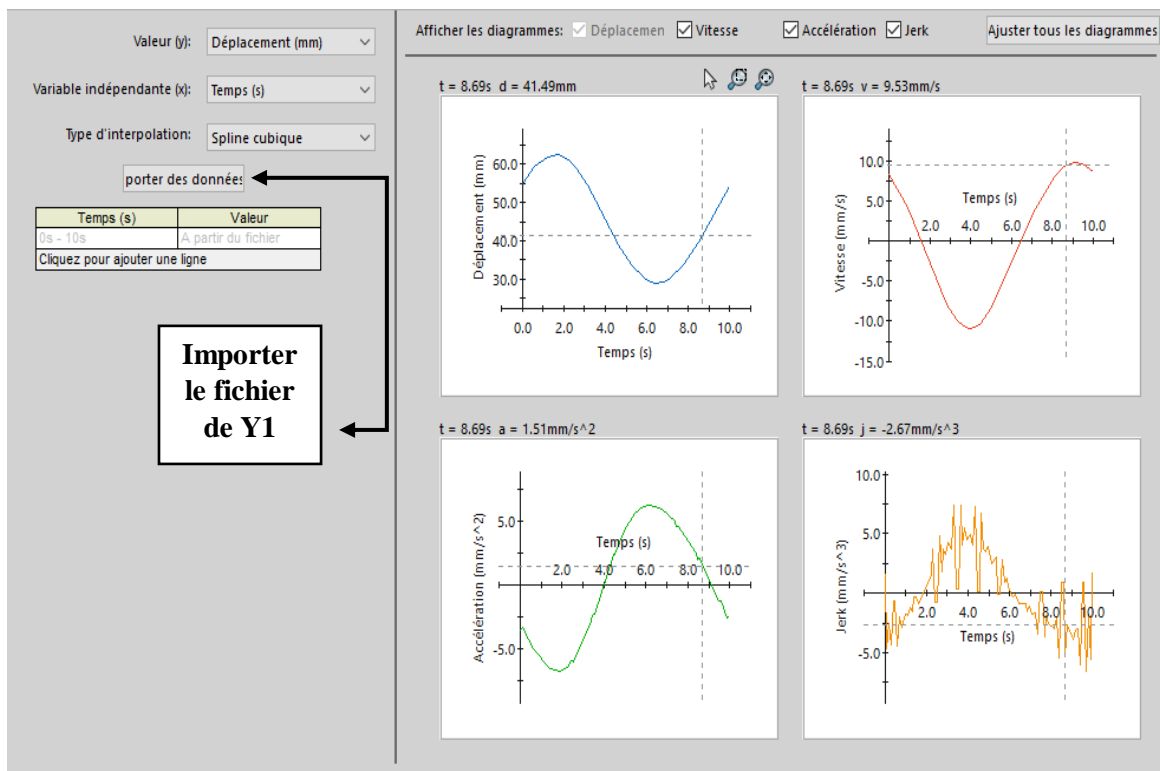


Figure 4.23. Résultat de Y1

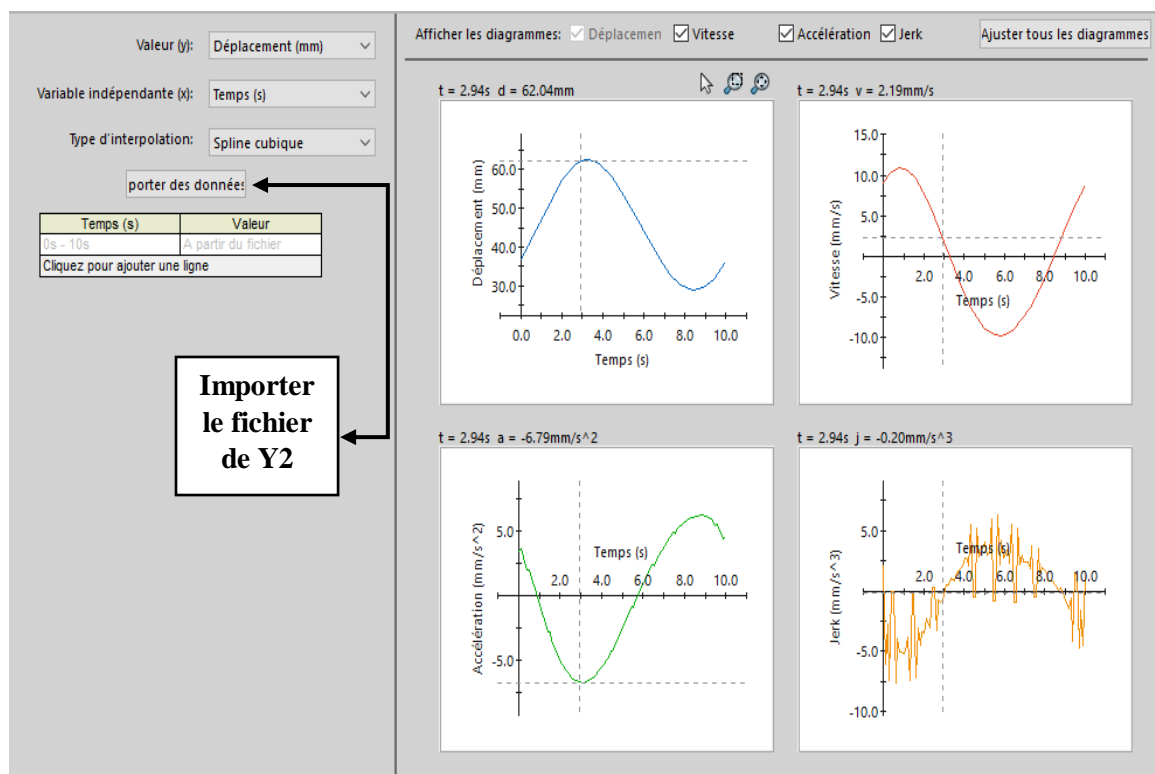
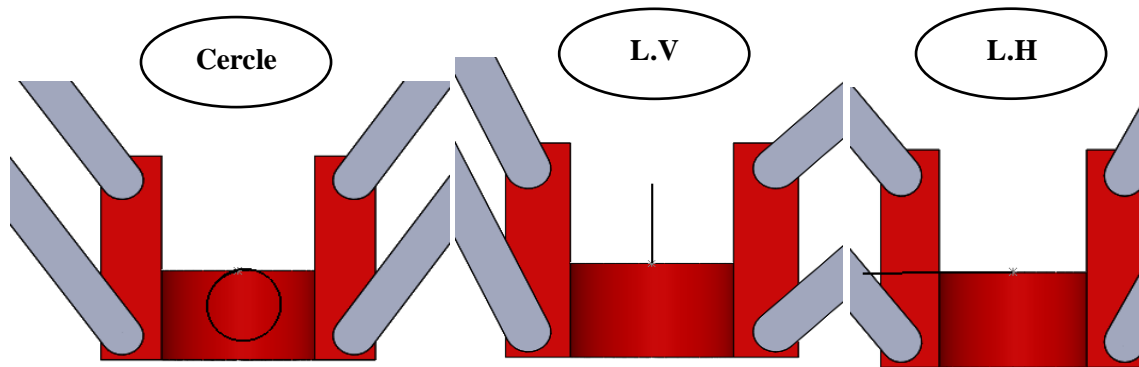


Figure 4.24. Résultat de Y2



**Figure 4.25.** Dessin sous SolidWorks des chemins importés

A travers les résultats obtenus dans SolidWorks, nous pouvons dire que la comparaison a révélé une concordance à **100%** car nous avons pu créer les mêmes formes avec les résultats du programme Matlab.

#### **4.9) Conclusion**

L'objectif de ce chapitre était de présenter l'analyse de mouvement de la structure de base de notre machine à l'aide de l'outil complémentaire de l'analyse des mouvements de SolidWorks. Nous avons dans un premier temps passé en revue les éléments de base de cet outil par la suite, nous avons présenté son application à notre cas. Par ailleurs, nous avons pu comparer les résultats obtenus par Matlab avec ceux délivrés par motion de SolidWorks, les résultats étaient concluants. Dans le prochain chapitre nous présenterons la partie électrique et informatique de notre machine.

## *Chapitre V :*

# *Présentation de la partie électronique et informatique*

## 5.1) Introduction

Ce chapitre sera consacré à la description du développement de la partie électrique et informatique de notre machine. Nos choix se sont naturellement tournés vers des solutions open-source largement utilisées par la communauté des *makers*. Nous utiliserons particulièrement GRBL, Arduino et CNC-shield. GRBL est un micrologiciel (appelé firmware) 3 axes open-source écrit pour une carte Arduino UNO, plus exactement pour l'ATMega 328, qui existe depuis de nombreuses années et fait un job remarquable pour sa petite taille (30 Ko) permettant le contrôle d'une CNC 3axe via une CNC-Shield avec précision en toute simplicité ! C'est la solution la plus utilisée par les *makers* pour tous les projets type CNC 3 axes. GRBL est d'ailleurs à la base de nombreux autres firmware plus élaborés.

## 5.2) Configurer GRBL et contrôler la machine CNC avec Arduino

Le pilotage d'une CNC se fait à l'aide de séquences d'instructions en G-code. GRBL est un outil pratique qui permet d'interpréter le G-code et de piloter très simplement une CNC avec Arduino. GRBL est un micrologiciel open source qui permet le contrôle de mouvement pour les machines CNC. Nous pouvons facilement installer GRBL sur une carte Arduino et nous obtenons ainsi instantanément un contrôleur CNC à faible coût et hautes performances. Le GRBL utilise le G-code comme entrée et produit le contrôle de mouvement via l'Arduino comme illustré sur la figure 5.1 [38]. D'après le diagramme, nous pouvons voir où se situe le GRBL dans la chaîne numérique d'une machine CNC. C'est le firmware que permet de contrôler les moteurs pas à pas de la machine CNC. En d'autres termes, la fonction du micrologiciel GRBL est de traduire le G-code en mouvements des différents moteurs de la machine.

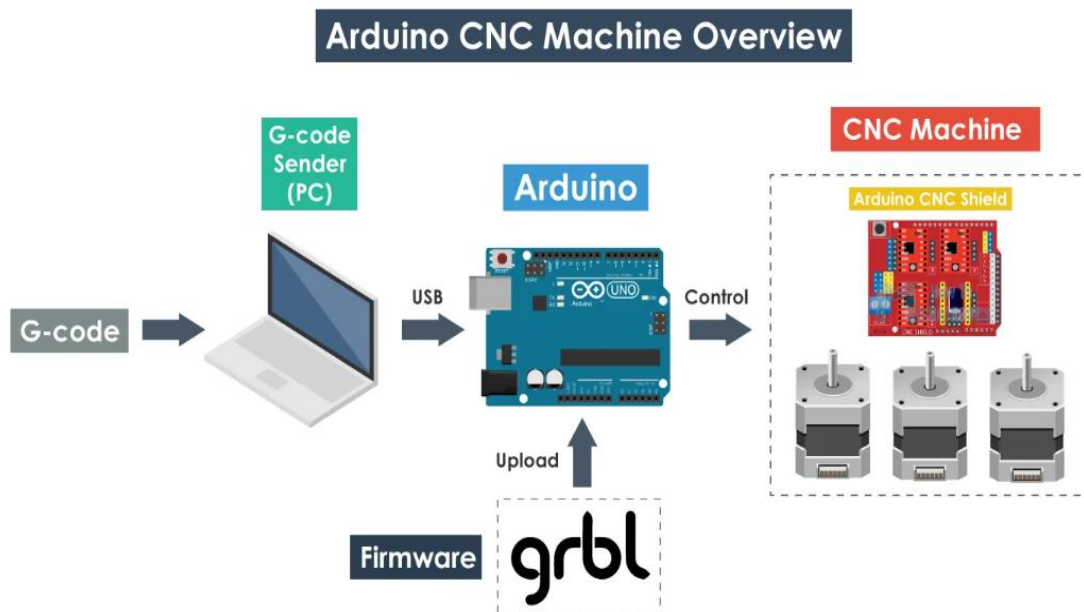


Figure 5.1. Communication du G-code vers la machine CNC

### 5.3) Matériel électronique

#### 5.3.1) Description des principaux éléments utilisés

Pour l'électronique, on va se baser sur une carte Arduino Uno qui est simple à prendre en main, elle est équipée d'une carte CNC Shield V3 (figure 5.2). Ce Shield porte un ensemble d'éléments de puissance (driver moteur) pour alimenter les actionneurs utilisés.

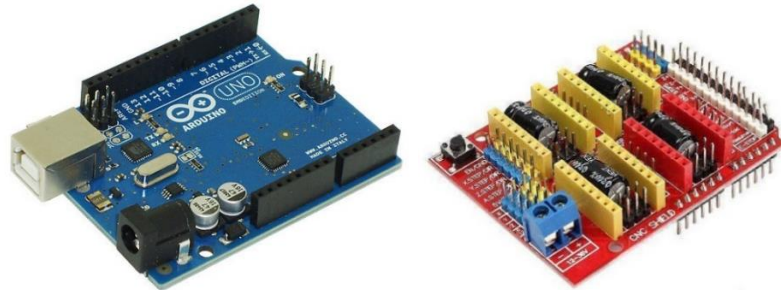


Figure 5.2. Arduino et le shield utilisés

En effet, le driver moteur (figure 5.3) va réceptionner les ordres de vitesse, de sens de rotation et de position provenant de l'électronique de commande, dans notre cas le Arduino, Puis transforme ces ordres en série d'impulsions électriques qu'il va envoyer au moteur. Nous avons utilisé dans notre montage le driver DRV8825 car il admet plus de courant et possède un micro-stepping bien plus fin.

Pour la motorisation nous allons utiliser des moteurs Nema17 (Fig. 5.4), qui sont des moteurs pas à pas, 200 pas par révolution. Ce sont des moteurs utilisés dans 99% des imprimantes 3D et CNC grand public qu'on peut trouver un peu partout .

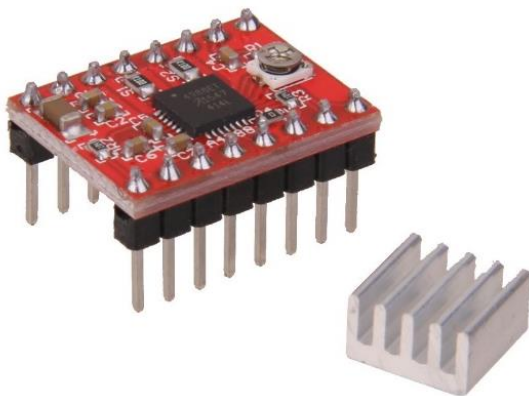


Figure 5.3. Driver DRV8825

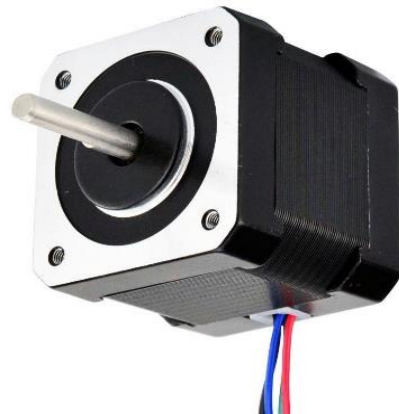


Figure 5.4. Le moteur pas a pas Nema 17

Pour la sécurité de fonctionnement de la machine nous allons utiliser des capteurs de fin de course comme celui de la figure 5.5. Ce sont des appareils électriques conçus pour ouvrir et fermer un circuit électrique. S'ils sont montés sur des mécanismes mobiles c'est pour limiter leur mouvement dans des limites prédéterminées. Les fonctions exécutées par ces appareils sont identiques à celles du commutateur standard. Le but de son utilisation

dans notre projet est de empêchent la machine de se déplacer au-delà de la zone de travail définie et ramener automatiquement la machine à la position de départ (Homing).

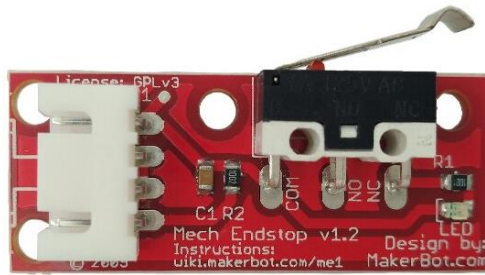


Figure 5.5. Capteur de fin de course

Enfin, pour l'alimentation, nous allons avoir besoin d'une alimentation assez puissante principalement pour supporter la charge des moteurs. Chaque moteur consomme environ 5W, ce qui nous donne 15W sur un montage à 3 moteurs. Nous avons utilisé une alimentation 12V 5A (environ 60W) qui est largement suffisante pour notre machine.



Figure 5.6. Alimentation 12V 5A

### 5.3.2) Assemblage des composants

Les différents composants cités précédemment seront connectés comme indiqué sur la figure 5.7.

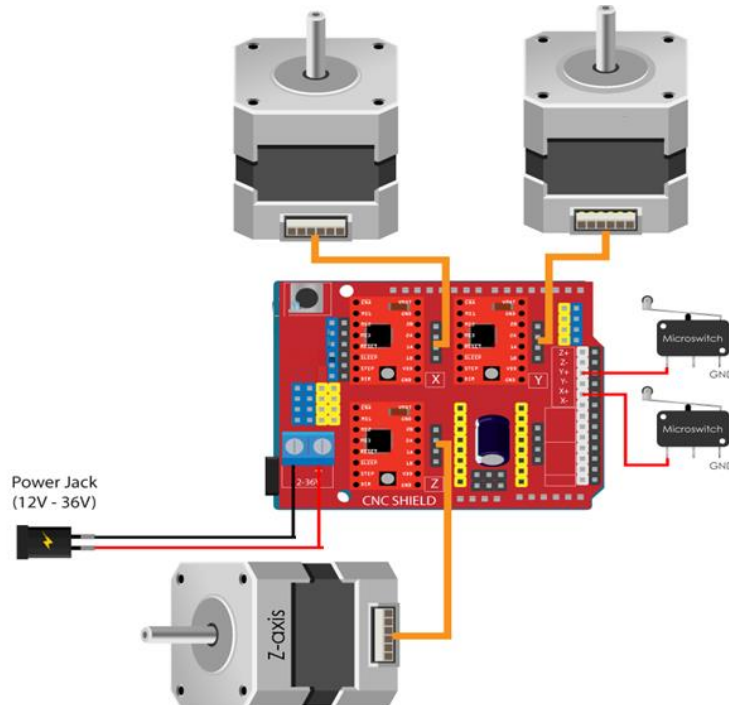


Figure 5.7. Schéma de câblage des différents éléments électriques

Pour commencer, on va monter la carte CNC Shield sur l'arduino comme indiqué sur la figure 5.8.

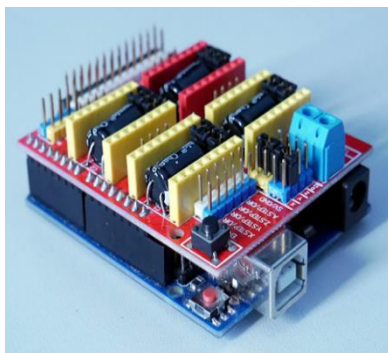


Figure 5.8. Position de montage du Shield

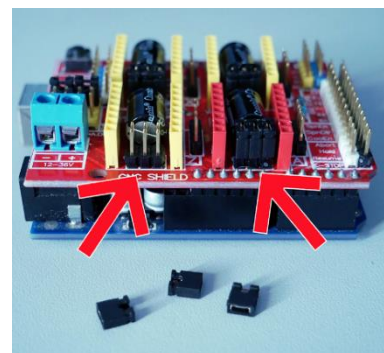


Figure 5.9. Placement des cavaliers de micro-pas

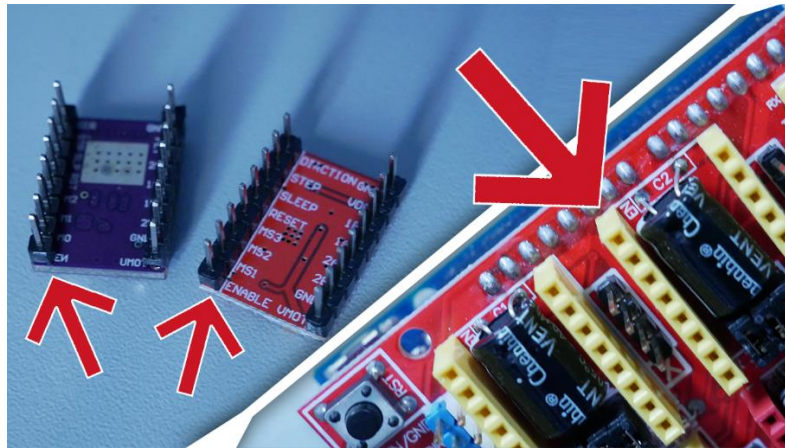
Ensuite, nous placerons les cavaliers de micro-pas, il y a 3 emplacements possibles sous chaque emplacement de pilote de moteur (Fig.5.9). En fonction du positionnement des cavaliers, on sélectionne le mode de division en micro-pas du driver (figure 5.10).



DRV8825			A4988			Mode de micro-stepping
J1	J2	J3	J1	J2	J3	
OFF	OFF	OFF	OFF	OFF	OFF	1 - aucun micro-stepping
ON	OFF	OFF	ON	OFF	OFF	1/2
OFF	ON	OFF	OFF	ON	OFF	1/4
ON	ON	OFF	ON	ON	OFF	1/8
OFF	OFF	ON	ON	ON	ON	1/16
ON	ON	ON				1/32

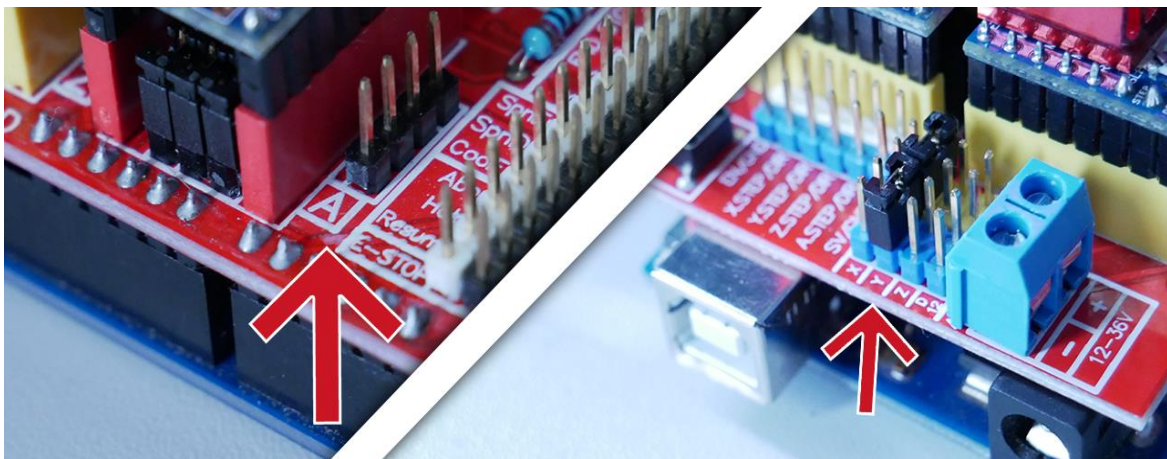
**Figure 5.10.** Placements possibles des cavaliers pour configurer les drivers

En plaçant les drivers sur le CNC-Shield, il faut surveiller et positionner le pin "Enable" (ou "En") du driver sur le pin "Enable" du CNC Shield.



**Figure 5.11.** comment placer correctement un driver sur un shield

Sur le CNC Shield, à côté de chaque emplacement de driver de moteur, il y a le nom de l'axe que ce driver pilotera (X, Y, Z ou autre). Enfin, Il suffit de brancher l'alimentation et les 3 moteurs selon le schéma de la **Figure 5.7**.



**Figure 5.12.** Placement potentiel des câbles moteurs

Pour l'alimentation c'est simple (Fig. 5.13), il y a un bornier sur la CNC Shield, on met le + sur le fil rouge, le - sur le fil noir. Pour connecter les moteurs, il y a un connecteur à 4 broches

à côté de chaque pilote. Il n'y a pas de sens particulier car en retournant le connecteur, on inverse simplement le sens de rotation du moteur.

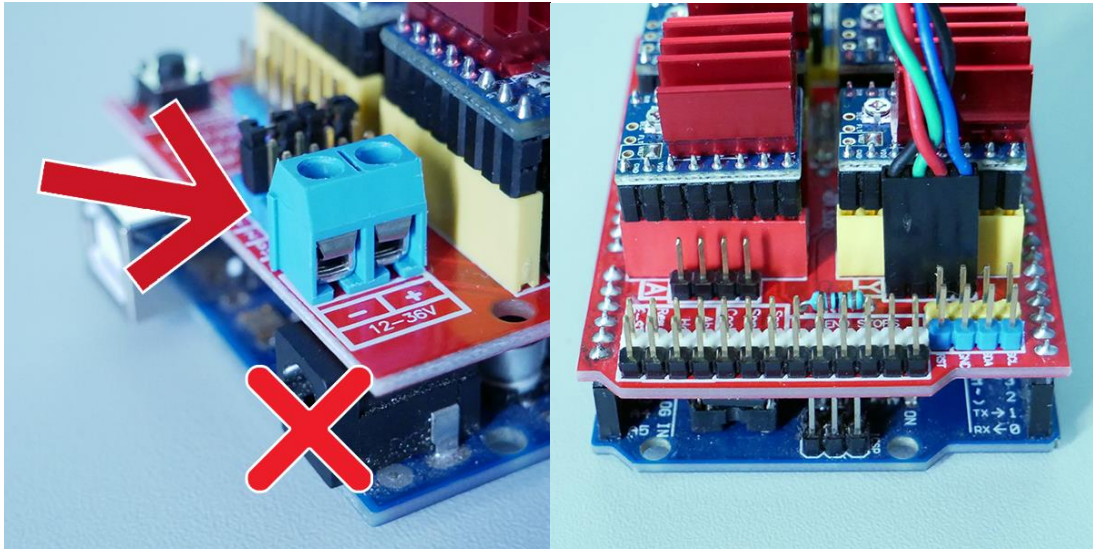


Figure 5.13. Placement de l'alimentation et des câbles moteurs

Enfin, nous allons connecter les fils des capteur de fin de course avec le Arduino , mais avant cela, il faut réaliser un montage précis de ces capteurs sur la machine aux endroits bien choisis comme illustré sur la figure 5.14 . Nous connectons le capteur de l'axe Z à l'entrée (SpnEn) et le capteur de l'axe X et Y à l'entrée X+ et Y+ (figure 5.15).

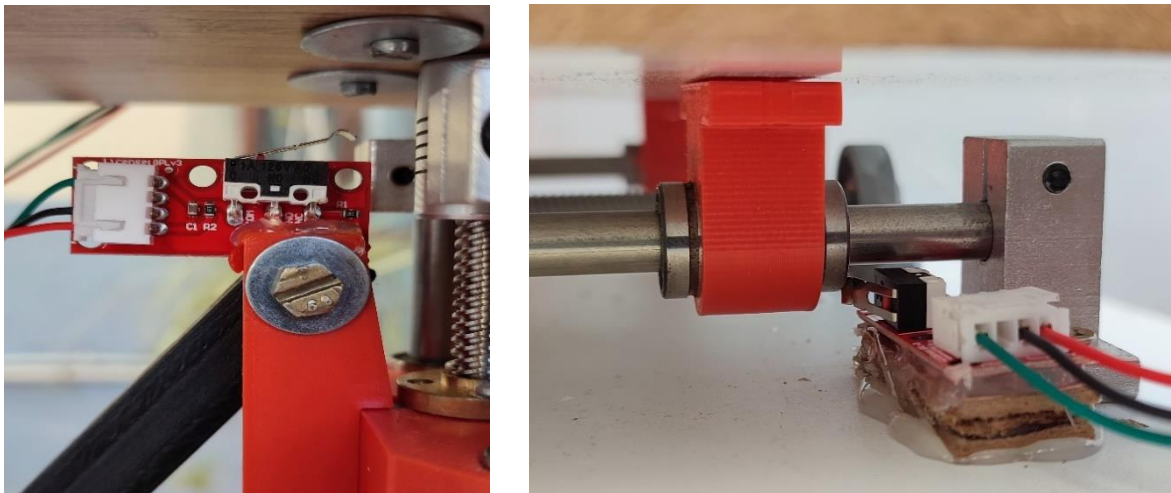


Figure 5.14. Montage des capteurs fin de course sur la machine

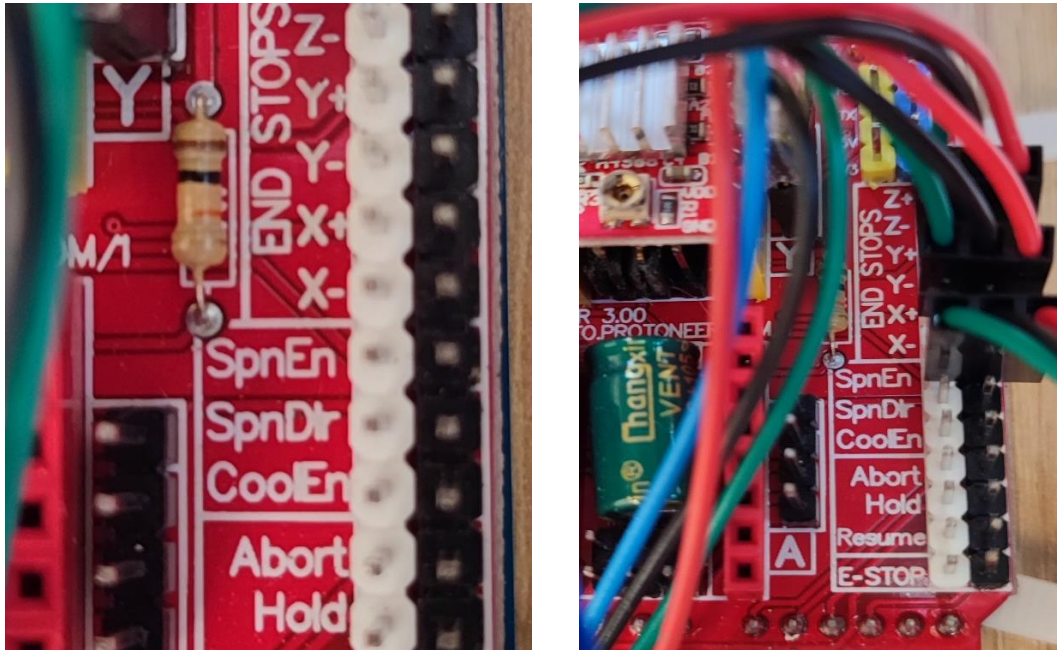


Figure 5.15. Montage de capteur fin-course en Arduino

L'assemblage est terminé, Nous pouvons maintenant passer à la mise en place de GRBL

#### 5.4) Comment installer le GRBL

Tout d'abord, pour pouvoir installer ou télécharger le GRBL sur l'Arduino, nous avons besoin de l'IDE Arduino. Ensuite, nous pouvons télécharger le firmware GRBL depuis github.com.

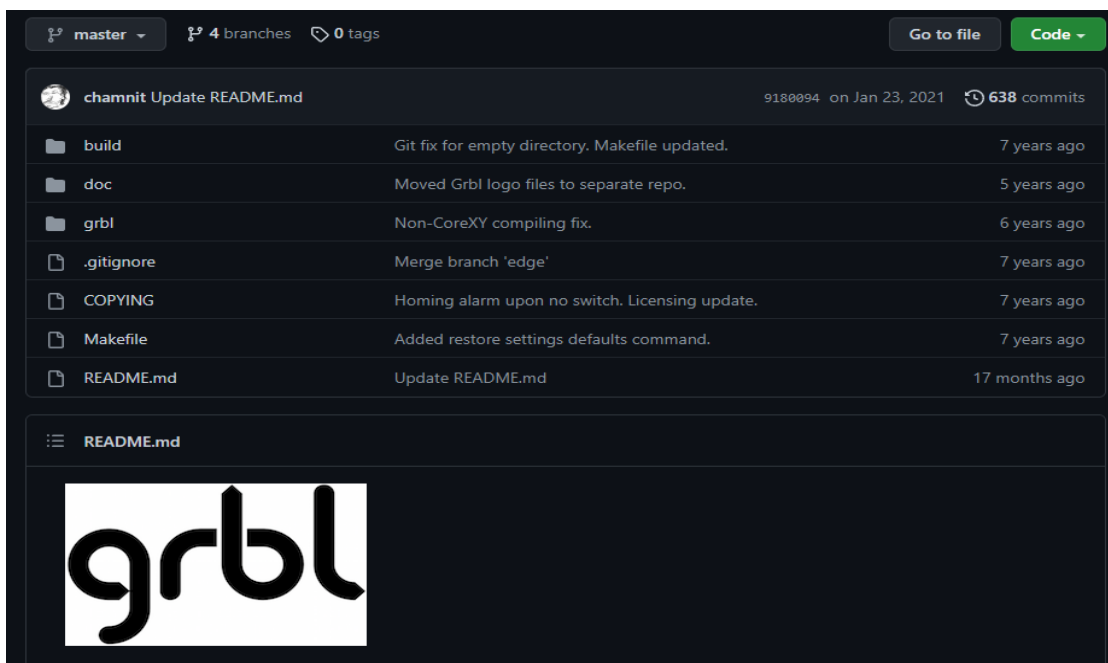


Figure 5.16. Téléchargement du GRBL

Téléchargez-le sous forme de fichier .ZIP, puis suivez ces étapes :

- Ouvrir le fichier grbl-master.zip et extraire les fichiers
- Ouvrir l'IDE Arduino, accéder à Sketch > Include library > add .ZIP...

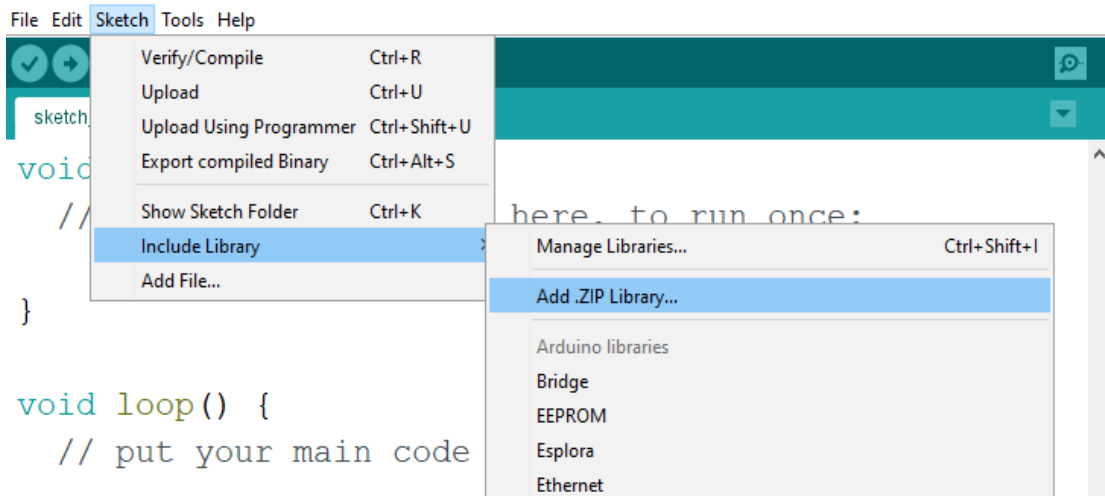


Figure 5.17. IDE Arduino

- Accéder au dossier extrait "grbl-master", sélectionner le dossier "grbl" et cliquer sur le fichier ouvert. Nous trouverons maintenant GRBL en tant que bibliothèque Arduino.

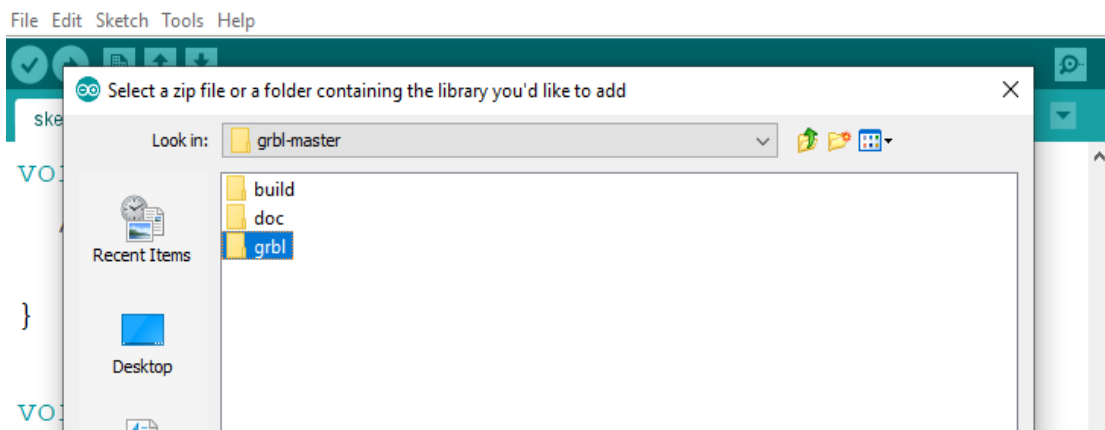


Figure 5.18. Sélection de GRBL

- Ensuite, accéder à Fichier > Exemples > grbl > grblUpload. Une nouvelle esquisse s'ouvrira et nous devons la télécharger sur la carte Arduino. Le code peut sembler bizarre car il ne s'agit que d'une seule ligne, mais pas de soucis, tout se passe en arrière-plan dans la bibliothèque. Il nous suffit donc de sélectionner la carte Arduino, le port COM et d'appuyer sur ce bouton de téléchargement et nous avons terminé.

## 5.5) Comment communiquer avec GRBL

Une fois que nous avons installé le micrologiciel GRBL, notre Arduino sait maintenant comment lire le code G et comment contrôler la machine CNC en fonction de celui-ci. Cependant, pour envoyer le code G à l'Arduino, nous avons besoin d'une sorte d'interface ou d'un logiciel de contrôleur qui dira à l'Arduino quoi faire. En fait, il existe de nombreux programmes open source et commerciaux à cette fin. Nous nous en tiendrons à l'open source, donc à titre d'exemple, nous utiliserons CNCjs (Figure 5.19) . En fait le CNCjs nous permettra de communiquer avec GRBL. Nous pourrons le configurer mais aussi et surtout, contrôler notre CNC !

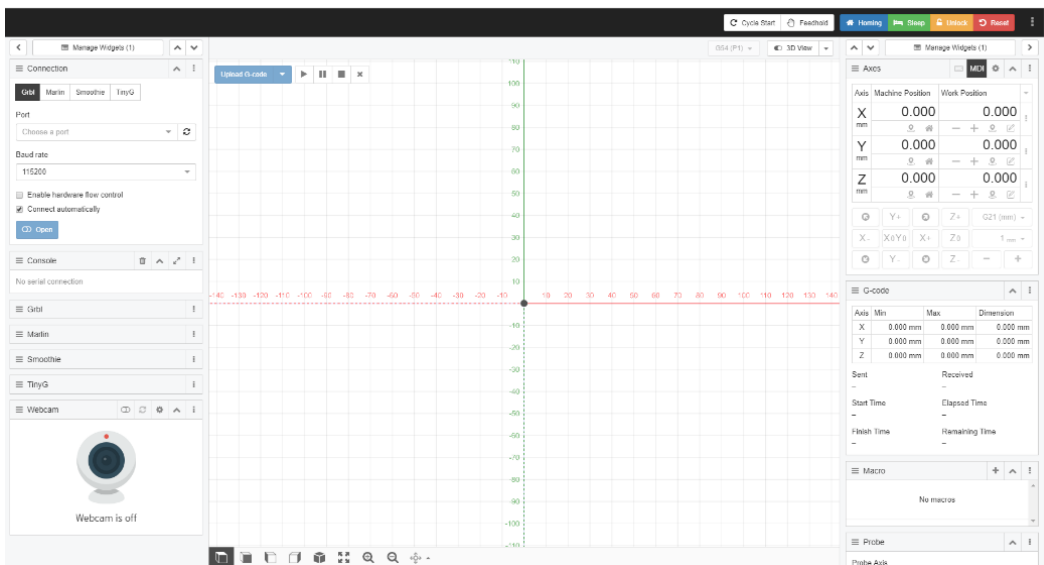


Figure 5.19. Interface CNC JS

Vous pouvez voir sur la gauche une section intitulée "Connection". Sélectionnez le port de communication de votre Arduino, réglez la vitesse de transmission sur 115200 puis cliquez sur "Open".

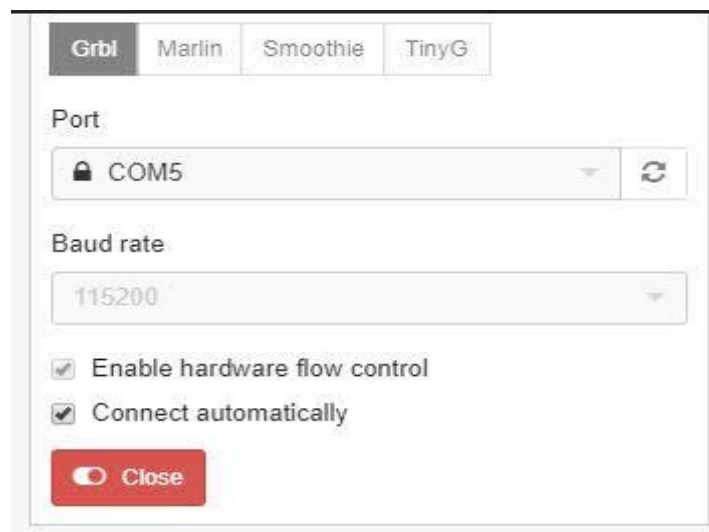


Figure 5.20. Sélection l'arduino

La section "Console" juste en dessous devrait s'ouvrir et afficher un tas de paramètres blancs sur fond noir. Ces paramètres sont envoyés par GRBL, ce qui signifie que GRBL est correctement exécuté. Nous avons fini d'installer le logiciel et de programmer l'arduino. Nous avons aussi vérifié que GRBL est bien installé et qu'il répond.

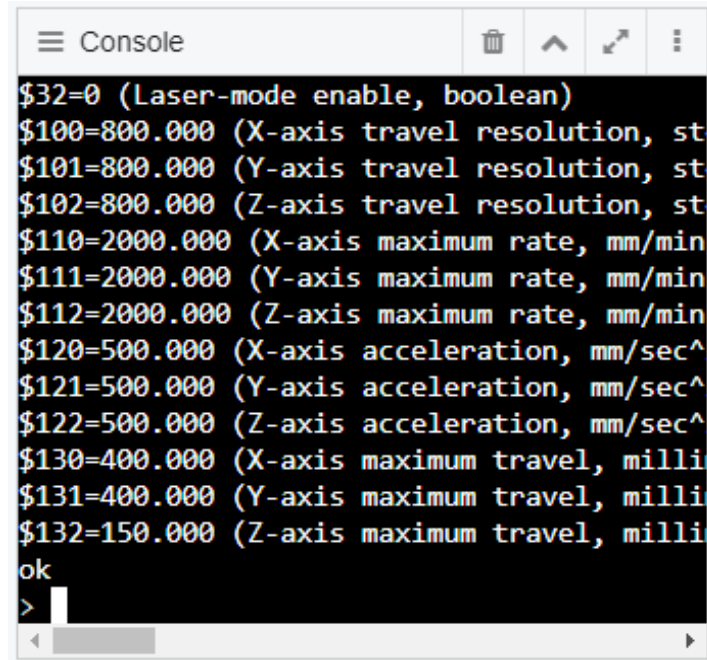
A screenshot of a console window titled "Console". The window has a dark background with white text. The text displays various GRBL parameters and their values, such as \$32=0 (Laser-mode enable, boolean), \$100=800.000 (X-axis travel resolution, steps/mm), \$101=800.000 (Y-axis travel resolution, steps/mm), \$102=800.000 (Z-axis travel resolution, steps/mm), \$110=2000.000 (X-axis maximum rate, mm/min), \$111=2000.000 (Y-axis maximum rate, mm/min), \$112=2000.000 (Z-axis maximum rate, mm/min), \$120=500.000 (X-axis acceleration, mm/sec^2), \$121=500.000 (Y-axis acceleration, mm/sec^2), \$122=500.000 (Z-axis acceleration, mm/sec^2), \$130=400.000 (X-axis maximum travel, mm), \$131=400.000 (Y-axis maximum travel, mm), and \$132=150.000 (Z-axis maximum travel, mm). The text ends with "ok" and a prompt character ">". The window has a standard toolbar with icons for back, forward, and search.

Figure 5.21. La section "Console"

## 5.6) Configuration du GRBL

La première étape ici consiste à sélectionner le débit en bauds, qui devrait être de 115200, et le port auquel notre Arduino est connecté. Une fois que nous avons connecté le CNCjs à l'Arduino,

Dans l'étape suivante, nous pouvons ajuster le paramètre pas/mm . Ici, il est beaucoup plus facile de comprendre comment l'ajuster car l'assistant de configuration calculera et nous dira à quelle valeur nous devons mettre à jour le paramètre.

La valeur par défaut est de 250 pas/mm. Cela signifie que si nous cliquons sur le bouton de déplacement "X-", le moteur fera 250 pas. Désormais, en fonction du nombre de pas physiques du moteur, de la résolution de pas sélectionnée et du type de transmission, la machine se déplacera sur une certaine distance. À l'aide d'une règle, nous pouvons mesurer le mouvement réel effectué par la machine et entrer cette valeur dans le « Mouvement réel » (Figure 5.22).



**Figure 5.22.** Réglage du paramètre pas/mm

Dans mon cas, la machine a bougé de 3 mm. Donc, dans La section "Console" on doit introduire les commandes suivantes :

\$100=200.000 (x, pas/mm)  
\$101=200.000 (y, pas/mm)  
\$102=200.000 (z, pas/mm)  
\$110=400.000 (taux max x, mm/min)  
\$111=400.000 (taux max y, mm/min)  
\$112=400.000 (taux max z, mm/min)  
\$120=10.000 (x accélération, mm/sec<sup>2</sup>)  
\$121=10.000 (y accel, mm/sec<sup>2</sup>)  
\$122=10.000 (z accel, mm/sec<sup>2</sup>)

En plus \$22=1 pour activer la fonction Homing.

maintenant nous pouvons voir les commandes qui sont exécutées. On peut remarquer qu'on a mis à jour les pas/mm .

```

Console
$100=200.000 (X-axis travel resolution, step
$101=200.000 (Y-axis travel resolution, step
$102=200.000 (Z-axis travel resolution, step
$110=500.000 (X-axis maximum rate, mm/min)
$111=500.000 (Y-axis maximum rate, mm/min)
$112=500.000 (Z-axis maximum rate, mm/min)
$120=10.000 (X-axis acceleration, mm/sec^2)
$121=10.000 (Y-axis acceleration, mm/sec^2)
$122=10.000 (Z-axis acceleration, mm/sec^2)
$130=200.000 (X-axis maximum travel, millime
$131=200.000 (Y-axis maximum travel, millime
$132=200.000 (Z-axis maximum travel, millime
ok
> $22=1
ok

```

Figure 5.23. Fenêtre de la Concole de CNCjs

Pour rendre le GRBL compatible avec la cinématique de notre machine, il est nécessaire de modifier le **GRBL** afin qu'il puisse utiliser les coordonnées X et Y du G-code pour calculer directement les coordonnées  $y_1$  et  $y_2$  en utilisant les équations du modèle géométrique inverse (MGI) et faire fonctionner correctement les deux moteurs. Cette modification peut faire l'objet d'une future étude.

Pour surpasser cette difficulté on a utilisé le logiciel MATLAB pour discrétiser les mouvements dans le plan vertical du porte outil. Pour chaque chemin désiré, on définit une série de points de coordonnées  $(x, y)$ , dans l'espace de travail et calculer ensuite  $y_1$  et  $y_2$  correspondant en utilisant le MGI. Ces derniers seront utilisés comme X, Y à transmettre à la machine en utilisant la fonction MACRO du logiciel CNCJS.

En ce qui concerne l'axe Z, aucun changement pour les coordonnées sur l'axe Z vu que ce dernier est indépendant des mouvement sur les deux axes X et Y.

### 5.7) Exemples d'application

Nous avons réalisé des programmes de Matlab pour préparer des opérations de perçage ou usinage 3 axe. Les figures suivantes montrent des captures d'écrans pour ces programmes.



```

Editor - C:\Users\Aymen\Desktop\edit program by younes 3\persage.m
afficher_PRRRPX.m x Model2_PRRRPX.m x persage.m x +
1 - clear all
2 - clc
3 - % Robot data
4 - H = 220; % Hauteur de la machine
5 - h = 30.5; % 1/2 Hauteur de la nacelle
6 - R = 146; % 1/2 largeur de la machine
7 - r = 41; % 1/2 largeur de la nacelle
8 - L = 185; % longueur des bras L>R
9 - N = 100; %Nbre de configuration à afficher
10 - a = 54 ; %origine
11 - b = 115 ; %distance entr centr de la nacelle et loutil
12
13 - for i=1:N;
14 -     % variables opérationnelles
15 -     % Mouvement vertical horizontal suur Z :
16 -     %MVM1
17 -     x1(i,1) = 35 ;
18 -     y1(i,1) = -20*i/N -330 + b;
19 -     z1(i,1) = 60 ;
20
21 -     %MVM2
22 -     x2(N+i,1) = 35 ;
23 -     y2(N+i,1) = 20*i/N -350 + b ;
24 -     z2(N+i,1) = 60 ;
25
26 -     %MVM3

```

<pre> 25 26 - %MVM3 27 -     x3(2*N+i,1) = -35*i/N + 35 ; 28 -     y3(2*N+i,1) = -330 + b ; 29 -     z3(2*N+i,1) = 60 ; 30 - %MVM4 31 -     x4(3*N+i,1) = 0; 32 -     y4(3*N+i,1) = -20*i/N -330 + b ; 33 -     z4(3*N+i,1) = 60 ; 34 - %MVM5 35 -     x5(4*N+i,1) = 0; 36 -     y5(4*N+i,1) = 20*i/N -350 + b ; 37 -     z5(4*N+i,1) = 60 ; 38 - %MVM6 39 -     x6(5*N+i,1) = -35*i/N + 0 ; 40 -     y6(5*N+i,1) = -330 + b ; 41 -     z6(5*N+i,1) = 60 ; 42 - %MVM7 43 -     x7(6*N+i,1) = -35 ; 44 -     y7(6*N+i,1) = -20*i/N -330 + b ; 45 -     z7(6*N+i,1) = 60 ; 46 - %MVM8 47 -     x8(7*N+i,1) = -35 ; 48 -     y8(7*N+i,1) = 20*i/N -350 + b ; 49 -     z8(7*N+i,1) = 60 ; 50 - %MVM9 </pre>	<pre> 50 - %MVM9 51 -     x9(8*N+i,1) = -35 ; 52 -     y9(8*N+i,1) = -330 + b ; 53 -     z9(8*N+i,1) = 35*i/N + 60 ; 54 - %MVM10 55 -     x10(9*N+i,1) = -35 ; 56 -     y10(9*N+i,1) = -20*i/N -330 + b ; 57 -     z10(9*N+i,1) = 95 ; 58 - %MVM11 59 -     x11(10*N+i,1) = -35 ; 60 -     y11(10*N+i,1) = 20*i/N -350 + b ; 61 -     z11(10*N+i,1) = 95 ; 62 - %MVM12 63 -     x12(11*N+i,1) = 35*i/N - 35 ; 64 -     y12(11*N+i,1) = -330 + b ; 65 -     z12(11*N+i,1) = 95 ; 66 - %MVM13 67 -     x13(12*N+i,1) = 0; 68 -     y13(12*N+i,1) = 20*i/N -350 + b ; 69 -     z13(12*N+i,1) = 95 ; 70 - %MVM14 71 -     x14(13*N+i,1) = 0; 72 -     y14(13*N+i,1) = - 20*i/N -350 + b ; 73 -     z14(13*N+i,1) = 95 ; </pre>
---	--

```

afficher_PRRRPX.m x Model2_PRRRPX.m x persage.m x +
74 %MVM15
75 x15(14*N+i,1) = 35*i/N ;
76 y15(14*N+i,1) = -330 + b ;
77 z15(14*N+i,1) = 95 ;
78 %MVM16
79 x16(15*N+i,1) = 35;
80 y16(15*N+i,1) = - 20*i/N -350 + b ;
81 z16(15*N+i,1) = 95 ;
82 %MVM17
83 x17(16*N+i,1) = 35;
84 y17(16*N+i,1) = 20*i/N -350 + b ;
85 z17(16*N+i,1) = 95 ;
86
87 end
88
89 X1=[x1; x2(101:200,1); x3(201:300,1); x4(301:400,1); x5(401:500,1);
90 Y1=[y1; y2(101:200,1); y3(201:300,1); y4(301:400,1); y5(401:500,1);
91 Z1=[z1; z2(101:200,1); z3(201:300,1); z4(301:400,1); z5(401:500,1);
92
93 sol= 1; % (on utilisera sol =1)
94 % q = MGI_PRRRP(x,y,L,r,R,sol);
95 % y1 = q(1);
96 % y2 = q(2);
97 y1 = (Y1+sqrt(L.^2-((X1+r)-R).^2))+a;
98 y2 = (Y1+sqrt(L.^2-((X1-r)+R).^2))+a;
99

```

```

usage.m x +
1 clear all
2 clc
3 % Robot data
4 H = 220; % Hauteur de la machine
5 h = 30.5; % 1/2 Hauteur de la nacelle
6 R = 140.5; % 1/2 largeur de la machine
7 r = 41; % 1/2 largeur de la nacelle
8 L = 185; % longueur des bras L>R
9 N = 100; %Nbre de configuration à afficher
10 a = 40 ; %origine
11 b = 115 ; %distance entr centr de la nacelle et loutil
12 px=[];
13 py=[];
14 for i=1:N;|
15 % variables opérationnelles
16 % Mouvement circulaire
17 xc = 0;
18 yc = -345 + b;
19 zc = 60;
20 rc = 40;
21 x = xc+rc*cos(2*pi*i/N) ;
22 y = -345 + b ;
23 z = zc+rc*sin(2*pi*i/N) ;
24 X1(i,1)=x ;
25 Y1(i,1)=y ;
26 Z1(i,1)=z ;
27 % fid=fopen('X1Y1.dat','r');

```

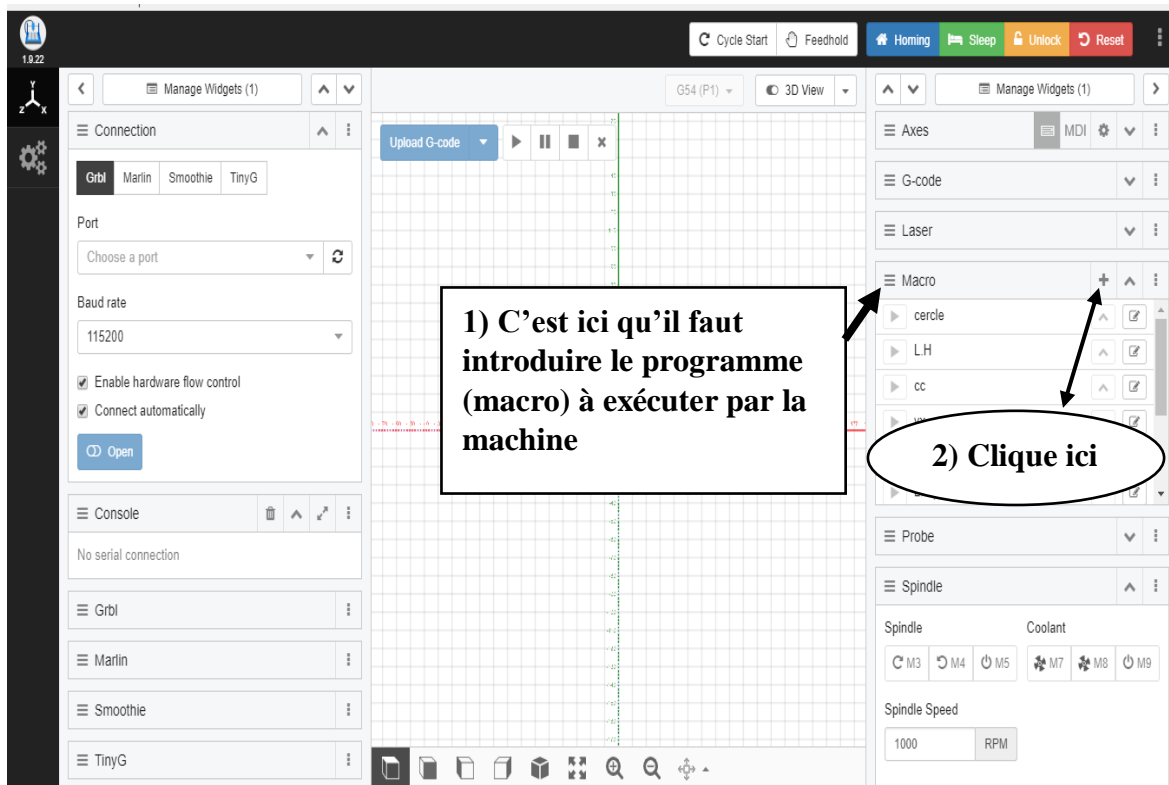
```

27      %   fid=fopen('X1Y1.dat','r');
28      % variables articulaires MGI
29 -    sol= 1; %(on utilisera sol =1)
30      %   q  = MGI_PRRRP(x,y,L,r,R,sol);
31      %   y1 = q(1);
32      %   y2 = q(2);
33 -    y1 = y+sqrt(L^2-((x+r)-R)^2)+a;
34 -    y2 = y+sqrt(L^2-((x-r)+R)^2)+a;
35 -    X(i,1)=y1;
36 -    Y(i,1)=y2;
37 -    Z(i,1)=z;
38 -    end
39 -    RESULTS=[X Y Z];
40      % save XYZpersage.dat RESULTS -ascii
41      %save XYZpersageXYZpersage(mvm10,mvm11).dat RESULTS -ascii
42 -    cercel.m
43 -    Affichage de cercel.m en cours...

```

Figure 5.24. Programme Matlab pour perçage et usinage

Les résultats obtenus à partir de ces programmes sont injectés dans l’interface CNCjs en utilisant les macro, pour cela nous suivons les étapes montrées dans la figure5.25.





**Figure 5.25.** Introduction du programme à exécuter (macro) pour suivre un chemin précis

### 5.8) Conclusion

Dans ce chapitre, nous avons présenté les principaux composants électriques utilisés dans notre projet ainsi que leur câblage. En outre nous avons décrit la partie informatique de la machine. Par ailleurs, Nous avons réussi à contrôler les moteurs pas à pas en utilisons GRBL téléversé sur arduino, sur la base des calculs effectués par le logiciel Matlab via le logiciel CNCjs.

# *Conclusion générale*

Le travail présenté dans ce mémoire visait la mise au point d'une petite MOCN à architecture parallèle pouvant réaliser diverses tâches d'usinage classiques de type gravure, perçage ou fraisage 3 axes. La partie mécanique responsable du positionnement de l'outil dans le plan est basée sur la structure d'un robot parallèle de type biglide à deux degrés de liberté qui assure aussi bien la rigidité que la précision de positionnement.

En résumé, notre travail a permis d'aboutir aux résultats suivants :

- ✚ La réalisation matérielle d'un premier prototype opérationnel de la machine en se basant sur des solutions technologiques disponibles sur le marché algérien dont le prix reste abordable.
- ✚ La mise au point de différents modèles analytiques du robot qui ont servis de base pour générer les trajectoires d'usinage du robot.
- ✚ La simulation cinématique et dynamique du comportement de notre machine après assemblage sous SolidWorks pour apprécier son comportement, dimensionner ses actionneurs et évaluer éventuellement ses performances
- ✚ L'adaptation de solutions informatiques d'usage commun dans le domaine des amateurs (Makers) des mini-CNC ou imprimantes 3D pour proposer une chaîne numérique de CFAO utilisable avec notre machine.

A travers notre étude, nous avons pu présenter une méthodologie générale pour l'intégration et l'adaptation des robots aux opérations d'usinage. Il faut souligner que la machine n'est qu'à sa première version, et que sa conception modulaire offre la possibilité de son développement afin d'inclure de nouveaux modules dédiés à des opérations d'usinage plus complexe.

Comme complément à ce travail, nous proposons les points suivants :

- ✚ Régler les paramètres de commande visant à assurer le comportement correct du système en termes de vitesse, d'accélération et de couple, pour permettre le respect des conditions de coupe préalablement établies.
- ✚ Réaliser d'une structure mécanique plus rigide afin de supporter des efforts de coupe plus importants générés lors des opérations d'usinage, en tenant compte des aspects ergonomiques et économiques. Il faut garder en tête le fait que l'architecture robot de base lui permet d'effectuer des opérations à Grande Vitesse si on utilise les bons actionneurs.
- ✚ S'attaquer au problème de la calibration, de manière à minimiser les sources d'erreurs susceptibles de diminuer la précision de la machine.
- ✚ Intégrer un axe d'orientation de pièces, pour augmenter la gamme de pièces réalisables (usinage en quatre axes).
- ✚ Exploiter les hautes performances de la structure du robot, en termes de précision et de répétabilité, pour l'utiliser comme une machine à mesure tridimensionnelle,

en partant des informations données par des codeurs qui seront placés à différents endroits de la machine et de la position d'un palpeur placé sur la nacelle du robot. Ce même concept peut être adopté pour la reconstitution (palper) des surfaces complexes (reverse engineering).

Enfin, nous espérons que ce modeste travail pourrait servir de référence aux Projets futurs des prochaines promotions et les inciter ainsi à s'intéresser d'avantage au côté pratique de mécanique en l'occurrence l'application de la robotique à la fabrication.

*Référence bibliographique*



- [1] : international federation of robotics <https://ifr.org/>
- [2] : Khalil W., Dombre E.: Modélisation identification et commande des robots, Hermès, Paris, 1999.
- [3] : MILUTINOVIC D., GLAVONJIK M., SLAVKOVIC N., DIMIC Z., ZINANOVIC S., KOKOTOVIC B., TANOVIC L. Reconfigurable robotic machining system controlled and programmed in a machine tool manner. International Journal Advanced In Manufacturing Technologie, London, 2010
- [4] : DENKENA B., LITWINSKI K., SCHÖNHERR M., Innovative drive concept for machining robots, 2ème CIRP Global Web Conference, pp. 67-72, 2013.
- [5] : Jose Gutierrez Tapia. Modélisation et identification géométrique de robots utilisés pour des opérations d'usinage. Médecine humaine et pathologie. Université Blaise Pascal - Clermont-Ferrand II, 2016.
- [6] : Jose Gutierrez Tapia. Modélisation et identification géométrique de robots utilisés pour des opérations d'usinage. Médecine humaine et pathologie. Université Blaise Pascal - Clermont-Ferrand II, 2016. Français. ffNNT : 2016CLF22728ff. fftel-01469691f .
- [7] : Merlet J.-P., Les robots parallèles, Hermès, Paris, 1997.
- [8] : Merlet J-P. Managing the redundancy of N-1 wire-driven parallel robots. In ARK, Juin 2012.
- [9] : Stewart D., A platform with six degrees of freedom. Proc. Inst. Mech. Engr, Vol.180, Pt 1(No 15): 371-386, 1965.
- [10] : Clavel R., "DELTA, fast robot with parallel geometry", PRO. Int. Symposium on industrial robots, avril 1988, p. 91-100.
- [11] : Gough V. E., "Contribution to discussion of papers on research in automotive stability, control and tyre performance", in Proc. Auto. Div., Institute of mechanical engineering, 1957.
- [12] : Sheldon, P. C., "Six Axis Machine Tool." US patent 5388935. Giddings & Lewis, Inc., fon du Lac, WI, 1995.
- [13] : TERRIER M., Optimisation du processus de fabrication en usinage à grande vitesse sur machines-outils à structure parallèle, Thèse de doctorat, Université de Nantes / Ecole Centrale de Nantes, Septembre 2005.
- [14] : COMPANY O., Machines-outils rapides à structure parallèle. Méthodologie de conception, application et nouveaux concept, Thèse de doctorat, Université de Montpellier II, décembre 2000.
- [15] : Sébastien Krut. A new parallel mechanism for Scara motions 2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422) .
- [16] : M. Terrier, M. Gymenez, J.Y. Hascoët, Verne – A 5 axis parallel kinematics milling machine, IMECH International Journal of Engineering Manufacture 219 (2004) 1343–1353.

- [17] : TLUSTY J., ZIEGERT J., RIDGEWAY S., Fundamental comparison of the use of serial and parallel kinematics for machine tools, *Annals of CIRP*, Volume 48/1, pp 351-356, 1999.
- [18] : Martin Y. S., Gimenez M., Rauch M., Hascoet J. Y., VERNE – A New 5- Axes Hybrid Architecture Machining Centre, 5th Chemnitzer Parallelkinematik Seminar, Chemnitz (Germany), April 25-26 2006, pp. 657-676 .
- [19] : Neumann, K. E.: Exechon concept. *Parallel Kinematic Mach. Res.*, (2006).
- [20] : T Puchtler - 5th Chemnitz Parallel Kinematics Seminar, 2006.
- [21] : Neumann K. E. Robot. U. S. Patent 4 732 525, Mars 22, 1988.
- [22] : Gough, V. E.: 1956, Contribution to discussion of papers on research in automobile stability, control and tyre performance, in: *Proc. Auto Div. Inst. Mech. Eng.*, pp. 392–395.
- [23] : Merlet, J.-P.: 2000, *Parallel Robots*, Kluwer Academic Publishers, London. Salisbury, J. K. and Graig, J. J.: 1982, Articulated hands: Force control and kinematic issues, *The Internat. J. Robotics Research* 1, 4–17.
- [24] : Liu, X.-J., Wang, J., Gao, F., and Wang, L.-P.: 2001, On the analysis of a new spatial three degrees of freedom parallel manipulator, *IEEE Trans. Robotics Automat.* 17, 959–968.
- [25] : Siciliano, B.: 1999, The Tricept robot: Inverse kinematics, manipulability analysis and closed-loop direct kinematics algorithm, *Robotica* 17, 437–445.
- [26] : Kong, X. and Gosselin, C. M.: 2002, Kinematics and singularity analysis of a novel type of 3-CRR 3-DOF translational parallel manipulator, *Internat. J. Robotics Research* 21, 791–798.
- [27] : Asada, H. and Kanade, T.: 1983, Design of direct-drive mechanical arms, *ASME Journal of Vibration, Acoustics, Stress, and Reliability in Design* 105, 312–316.
- [28] : Gao, F., Liu, X.-J., and Gruver, W. A.: 1998, Performance evaluation of two degrees of freedom planar parallel robots, *Mechanisms and Machine Theory* 33, 661–668.
- [29] : McCloy, D.: 1990, Some comparisons of serial-driven and parallel driven mechanisms, *Robotica* 8, 355–362.
- [30] : XIN-JUN LIU., QI-MING WANG., JINSONG WANG., Kinematics, Dynamics and Dimensional Synthesis of a Novel 2-DoF Translational Manipulator , *Journal of Intelligent and Robotic Systems* 41, 205–224, 2004.
- [31] : Gosselin, C. and Angeles, J.: 1991, A global performance index for the kinematic optimization of robotic manipulators, *Transaction of the ASME, Journal of Mechanical Design* 113, 220–226.
- [32] : Liu, X.-J., Jin, Z.-L., and Gao, F.: 2000, Optimum design of 3-DOF spherical parallel manipulators with respect to the conditioning and stiffness indices, *Mechanism and Machine Theory* 35, 1257– 1267.

[33] : Daniali, H. R. M., Zsombor-Murray, P. J., and Angeles, J.: 1995, Singularity analysis of a general class of planar parallel manipulators, in: Proceedings of IEEE Internat. Conference on Robotics and Automation, Nagoya, Japan, pp. 1547–1552.

[34] : Catalogue vis trapézoïdale et écrou .CORETEC.

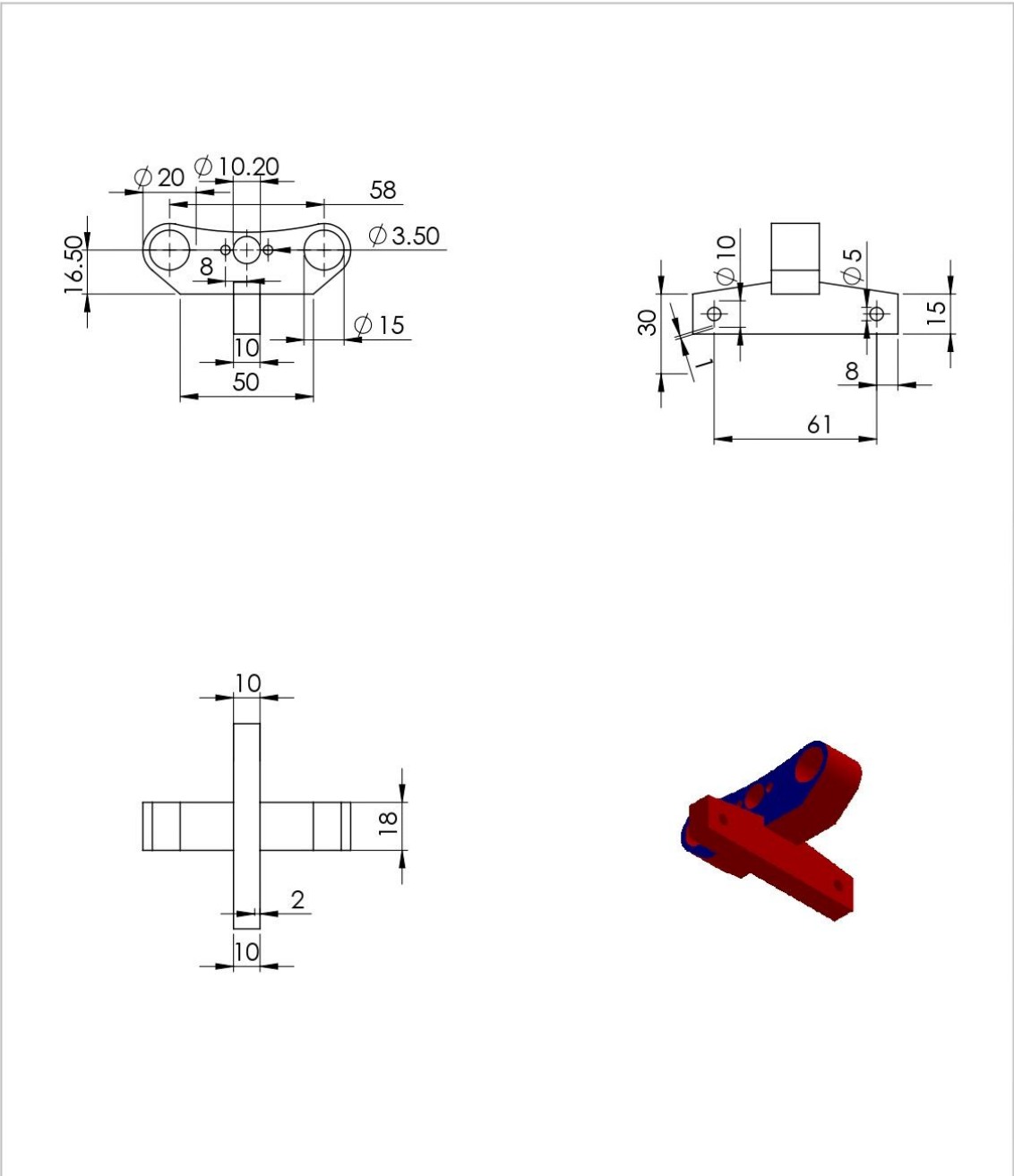
[35] : Catalogue Général Industry Entrez dans l'Univers SNR.

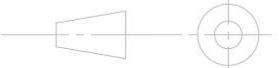
[36] : Conception et étude d'une fraiseuse à commande numérique MOCN OULMI ,Tassadit Mlle,MERZOUKI Zin, Université de Mouloud Mammeri de Tizi Ouzou,algeria, 2014/2015.

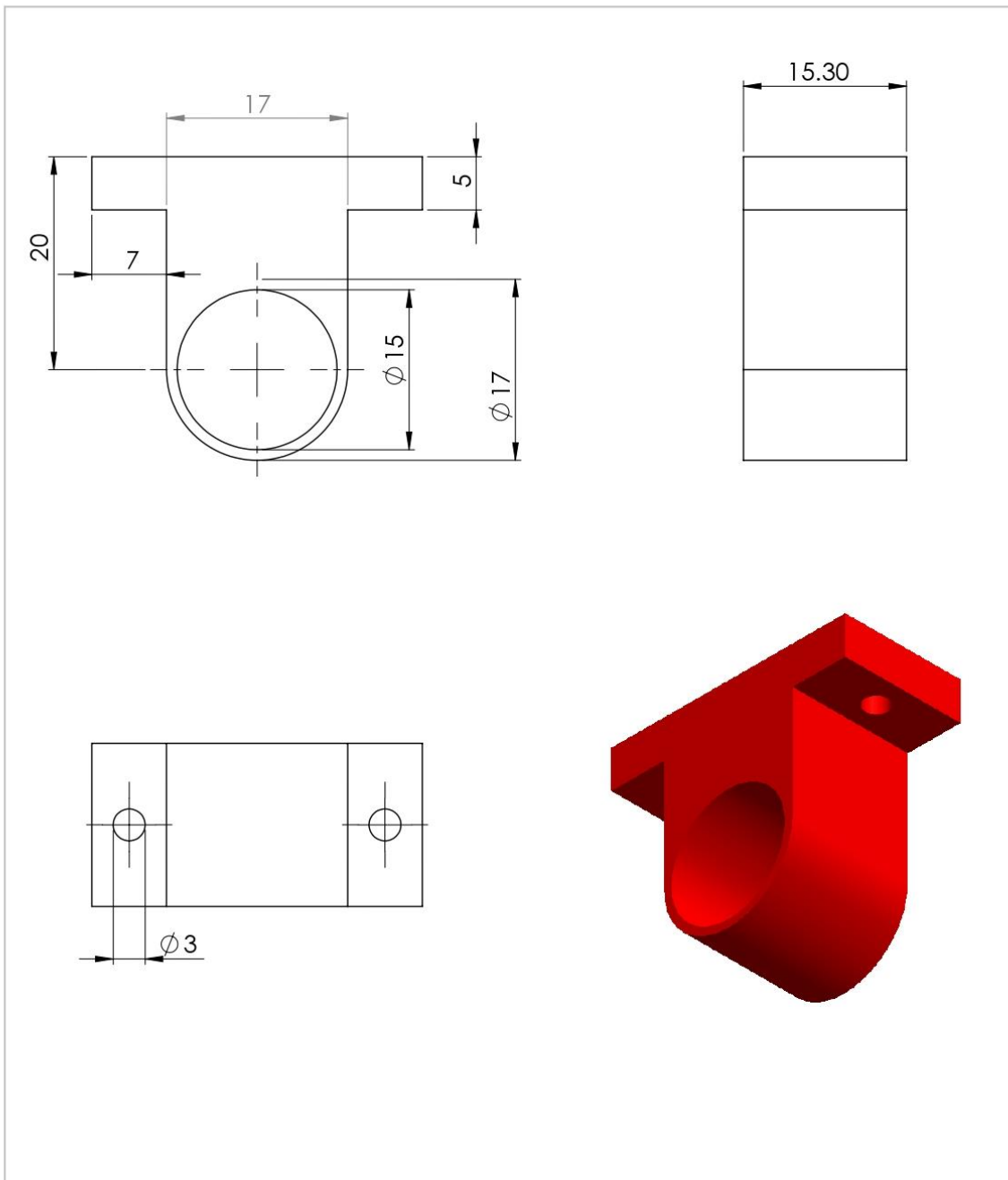
[37] : Catalogue Roulements à billes et à rouleaux NTN.

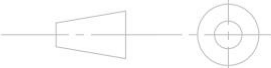
[38] : <https://howtomechatronics.com/tutorials/how-to-setup-grbl-control-cnc-machine-with-arduino/> accès le 26/05/2022.

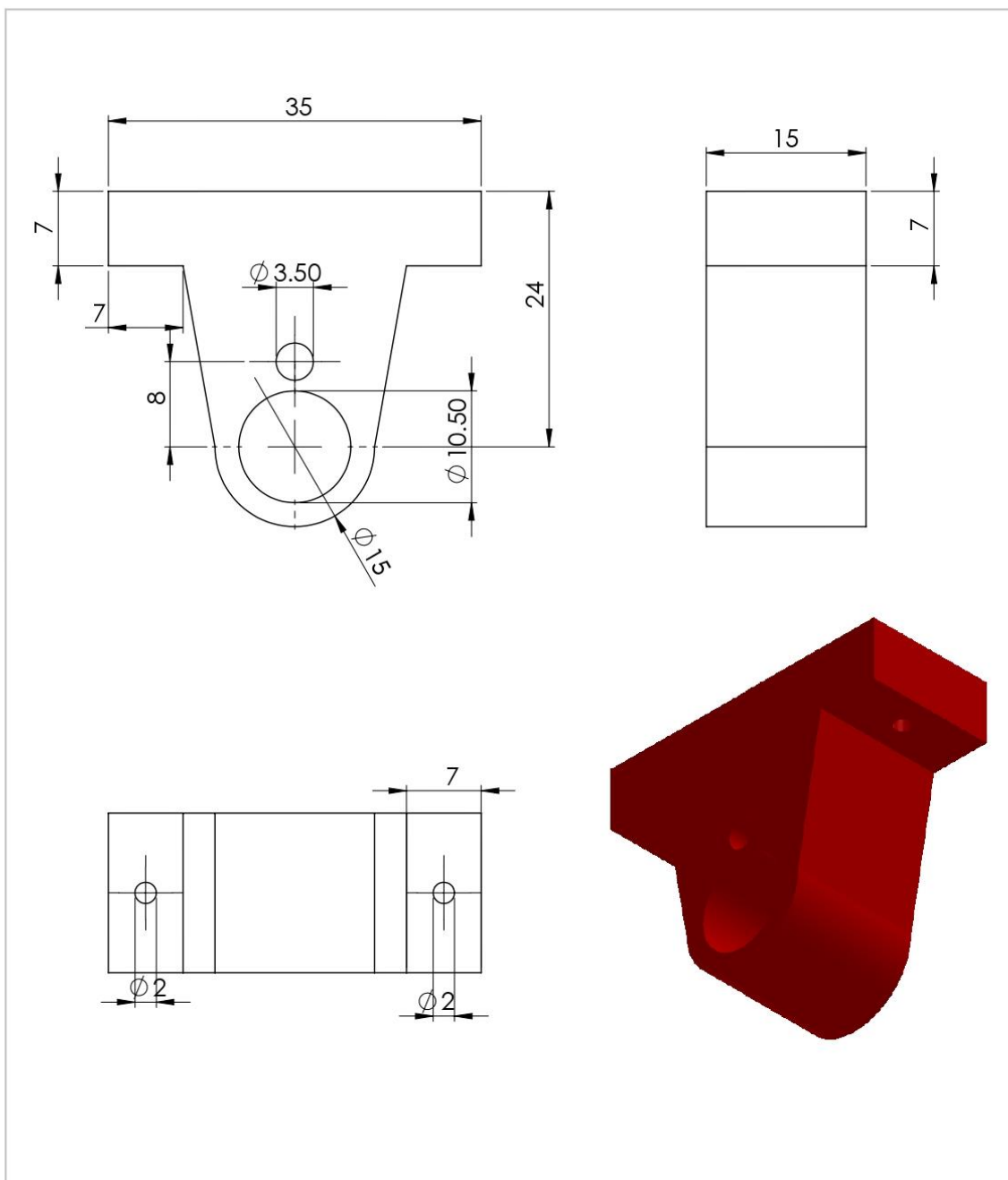
# *Annexes*

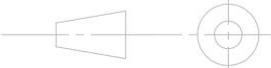


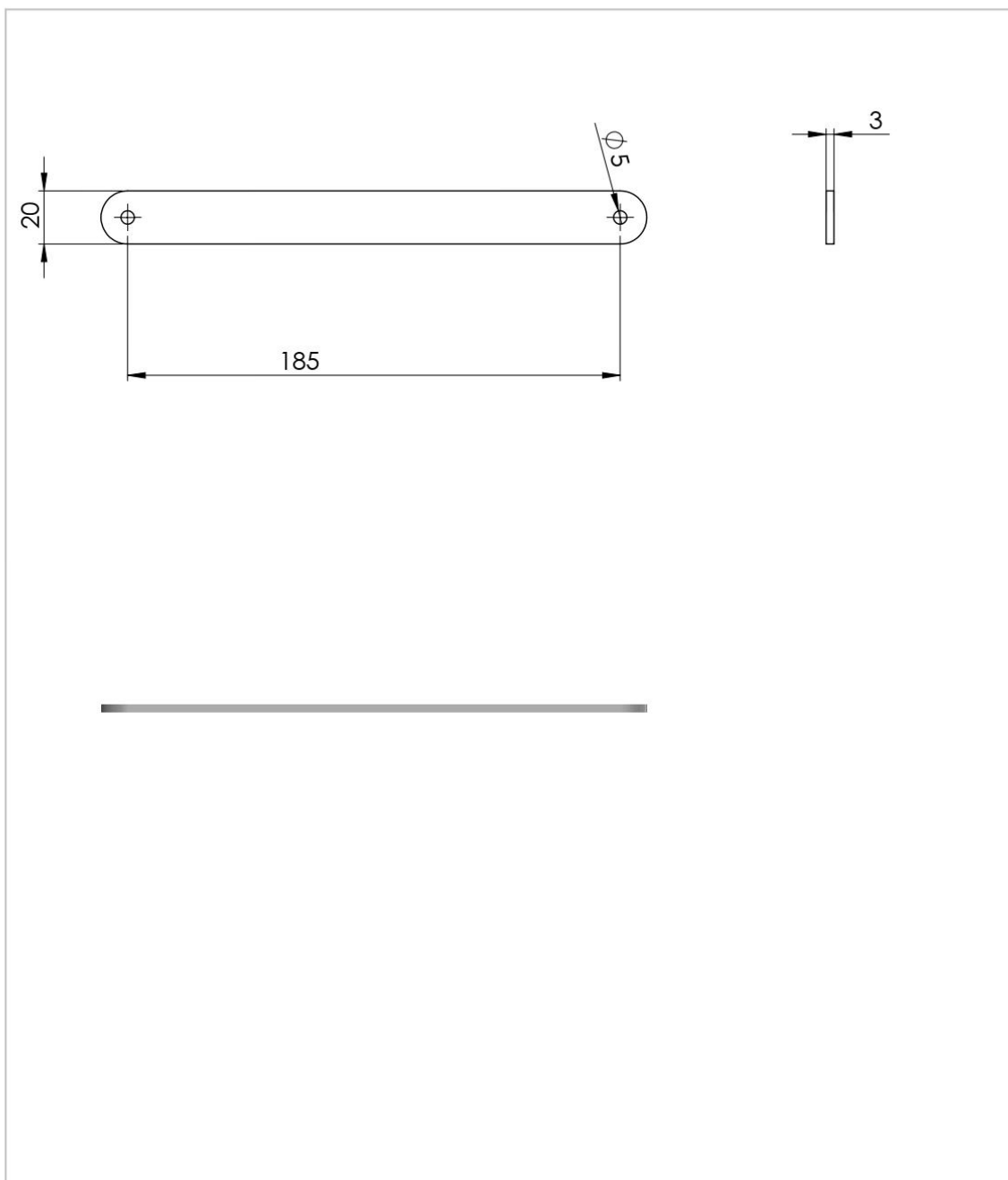
	<b>2</b>	<b>Piece.01</b>	<b>Plastique</b>	
Rep	Nbr	Designation	Matière	Observation
<b>Echelle : 1.2</b>		<b>Mini-CNC 3-axes à architecture parallèle et à commande numérique</b>	<b>Réalisé par: Lakas Aymen Smadi Younes</b>	
			Projet de fin d'études <b>2021/2022</b>	
A4		Université Saad Dahlab.BLIDA.01	<b>FMP_Master 2</b>	

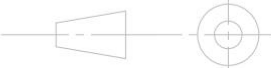


	<b>4</b>	<b>Piece.02</b>	<b>Plastique</b>	
Rep	Nbr	Designation	Matière	Observation
Echelle : 1.2		<b>Mini-CNC 3-axes à architecture parallèle et à commande numérique</b>	<b>Réalisé par: Lakas Aymen Smadi Younes</b>	
			Projet de fin d'études <b>2021/2022</b>	
A4		Université Saad Dahlab.BLIDA.01	<b>FMP_Master 2</b>	

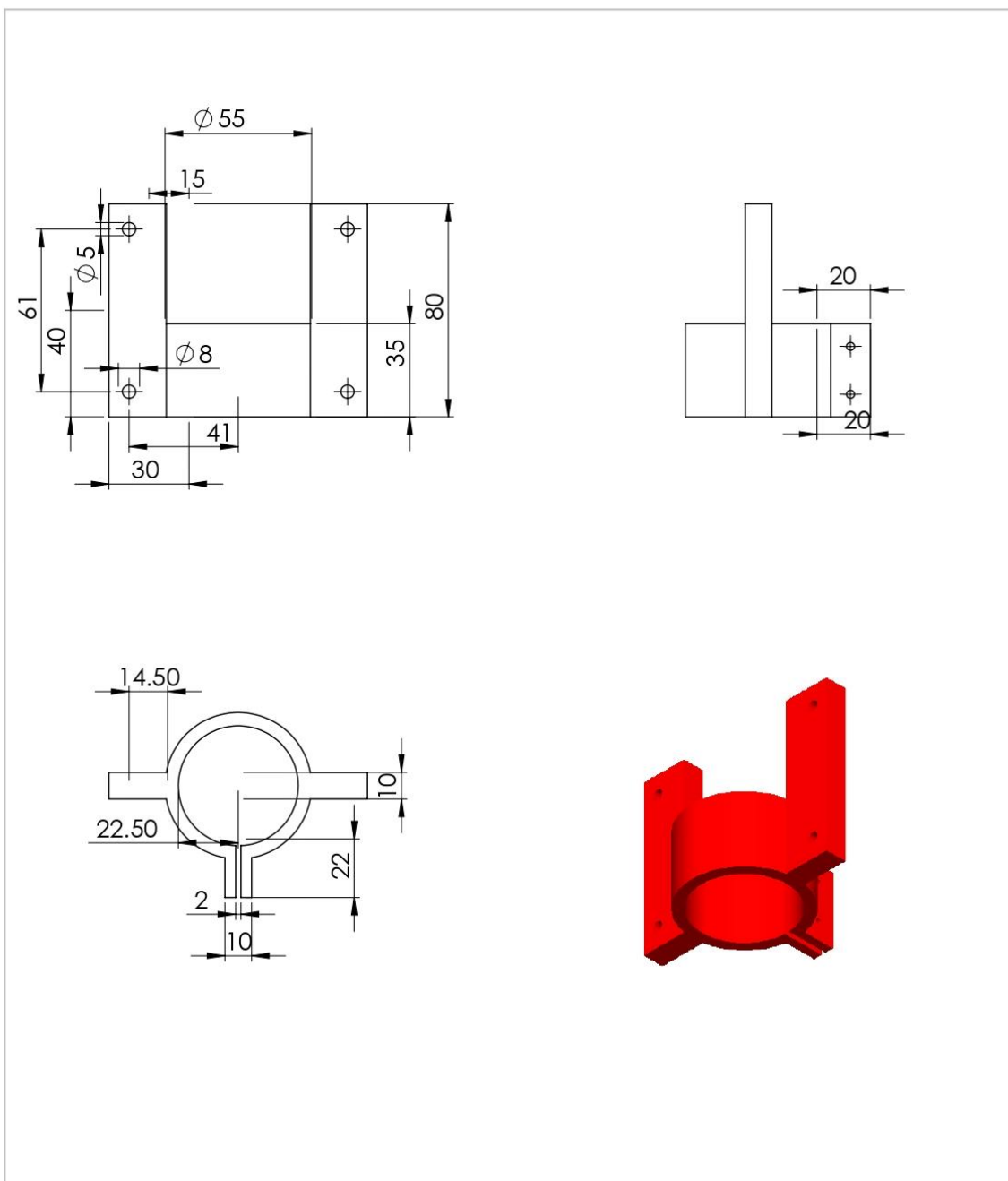


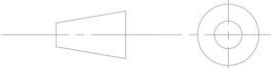
	1	Piece.03	Plastique	
Rep	Nbr	Designation	Matière	Observation
Echelle : 1.2		Mini-CNC 3-axes à architecture parallèle et à commande numérique	Réalisé par: Lakas Aymen Smadi Younes	
			Projet de fin d'études 2021/2022	
A4		Université Saad Dahlab.BLIDA.01	FMP_Master 2	

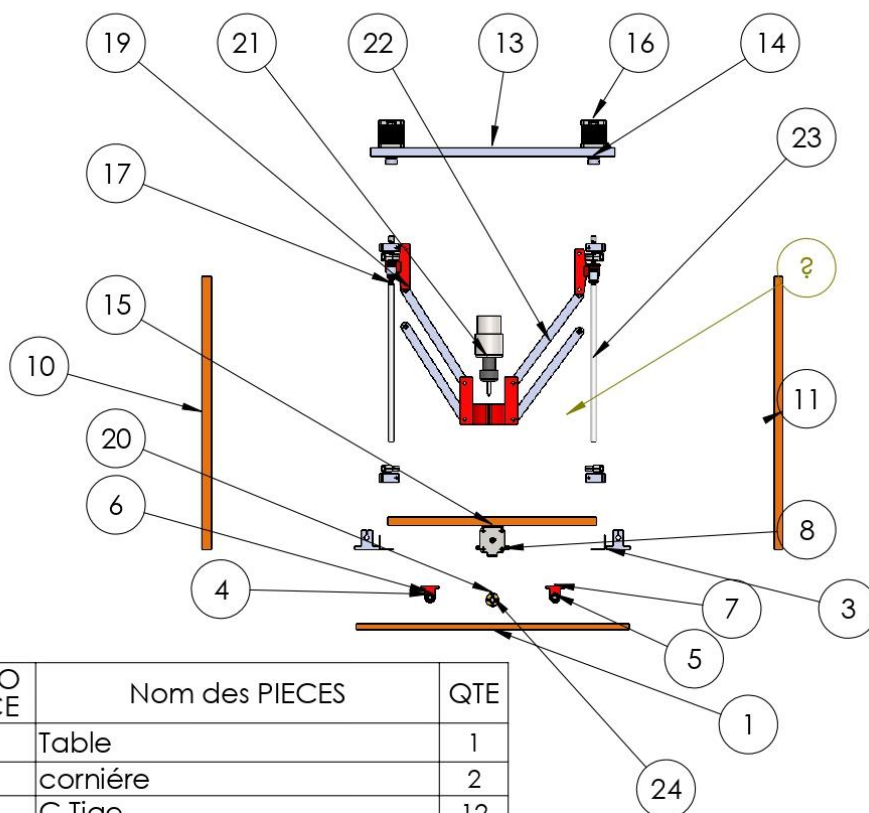


Rep	Nbr	Designation	Matière	Observation
	4	Bras	Acie	
Echelle : 1.2		Mini-CNC 3-axes à architecture parallèle et à commande numérique		Réalisé par: Lakas Aymen Smadi Younes
				Projet de fin d'études <b>2021/2022</b>
A4		Université Saad Dahlab.BLIDA.01	FMP_Master 2	





Rep	Nbr	Designation	Matière	Observation
	1	La Nacelle	Plastique	
Echelle : 1.2		Mini-CNC 3-axes à architecture parallèle et à commande numérique		Réalisé par: Lakas Aymen Smadi Younes
				Projet de fin d'études 2021/2022
A4		Université Saad Dahlab.BLIDA.01	FMP_Master 2	



NUMERO DE PIECE	Nom des PIECES	QTE
1	Table	1
2	cornière	2
3	C.Tige	12
4	tige AXE.X(410mm)1	1
5	tige AXE.X(410mm)2	1
6	Roulement linéaire	8
7	c.roulement	4
8	palier a bille	6
9	ISO 15 ABB - 198 - 10,SI,NC,10_68	2
10	Plac .AXE.Z	1
11	Plac .AXE.Z	1
12	structure ronforcéé	2
13	Plac .AXE.Y	1
14	coupling	3
15	Plac.axe.x	1
16	moteur pas apas . Nema 17	3
17	tige AXE .Z (264mm)	4
18	C.BROCHE	1
19	C.bielle	2
20	Ecrou de vis 8mm x P2mm	3
21	BROCHE	1
22	bielle	4
23	vis a chonfen (YZ)(264mm)	2
24	vis a chonfen (X)(410mm)	1
25	C.vis.X	1

**Mini-CNC  
3-axes  
à architecture  
parallèle  
et à  
commande  
numérique**

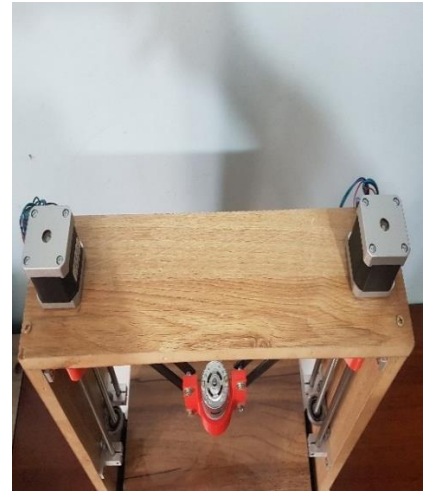
**Réalisé par:**  
**Lakas Aymen**  
**Smadi Younes**  
Projet de fin d'études  
2021/2022

**FMP\_Master 2**

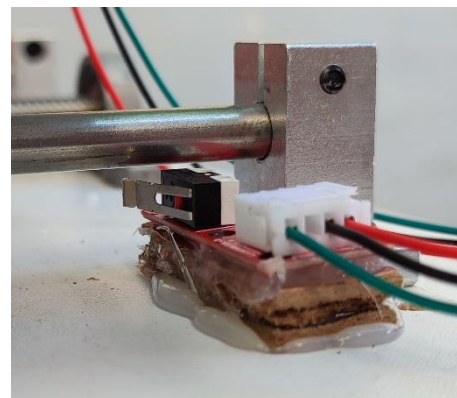
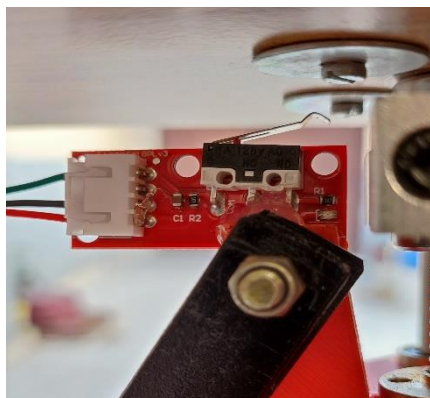
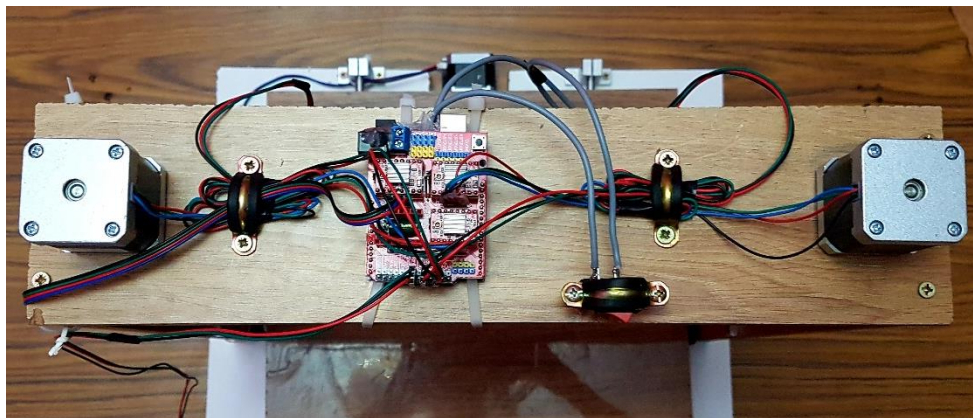
➤ Les différentes éléments de la machine



➤ Différentes étapes de réalisation



➤ Les éléments électroniques



➤ **Forme finale de la machine**

