

Rapport de projet :

La conception d'une machine CNC à trois axes

Réalisé par : LABIAD Anas

2019-2020

Sommaire

Remerciement	5
Introduction générale	6
Historique	7
Chapitre 1 : généralité sur les machines à commande numérique par ordinateur	
I.1. Introduction	8
I.2. Définition de la commande numérique	8
I.3. Définition d'une machine CNC	8
I.4. Domaine d'utilisation	10
I.5. Principe de fonctionnement	11
I.6. Les différents types des CNC	12
I.7. Classification des machines CNC	12
I.7.1. Classification des CNC selon le mode de fonctionnement	12
I.7.2. Classification des CNC selon le mode d'usinage	13
I.8. La structure d'une machine CNC	15
I.8.1. Le système électronique	15
I.8.2. Le système mécanique	15
I.8.3. Le système de programmation	15
I.9. Conclusion	15
Chapitre 2 : conception d'une machine CNC	
II.1. Introduction	17
II.2. Cahier de charge	17
II.3. Partie mécanique	17
II.3.1. Conception des pièces à usiner	17
II.3.2. Les pièces standards	21

II.3.3. Assemblage de la machine CNC	22
II.3.4. Dimensionnement des vis trapézoïdale	23
II.4. La partie électronique	29
II.4.1. Introduction	29
II.4.2. Moteur pas à pas	29
II.4.3. La carte de commande Arduino	33
II.4.4. Stepper motor driver A4988	35
II.4.5. La carte SHIELD CNC	37
II.5. Partie Informatique et Programmation	38
II.5.1. Introduction	38
II.5.2. Langue de contrôleurs numériques « G-code »	38
II.5.2.1. Description	38
II.5.2.2. Format d'une ligne	39
II.5.3. L'éditeur du texte ou image compilateur de langage CNC	40
II.5.3.1. Définition	40
II.5.3.2. Interface	40
II.5.4. Logiciel de pilotage de la machine CNC	42
II.5.4.1. Mode manuel	42
II.5.4.2. Mode automatique	43
II.5.5. Langage de programmation la machine CNC	46
II.5.5.1 Programme Grbl	45
II.5.5.1.1. La communication avec la carte Arduino	47
II.5.5.1.2. Code de l'opération	47
II.5.6. Conclusion	49

Chapitre 3 : Analyse fonctionnelle, Étude AMDEC et l'Analyse de défaillance (ADD)

III.1. Analyse fonctionnelle	50
III.1.1. Énoncé du besoin (Bête à cornes)	50
III.1.2. Analyse du besoin : Diagramme Pieuvre	51
III.1.3. Analyse technique : Diagramme Fast	52
III.2. Étude AMDEC	52
III.2.1. Définition AFNOR	52
III.2.2. Types de l'AMDEC	52
III.2.3. Buts de l'AMDEC	53
III.2.4. L'analyse des défaillances	53
III.2.5. L'évaluation	54
III.2.6. Les actions	54
III.2.7. Application de l'AMDEC sur la machine CNC	56
III.3. Analyse de défaillance (ADD)	58
III.3.1. Arbre de défaillance	58
III.3.2. Traitement qualitatif	58
III.3.3. Traitement quantitatif	58
Conclusion Générale	60

Remerciement

Il n'est sans doute pour moi que la plus agréable des tâches de ce travail est celle de vous exprimer mes plus sincères remerciements. Tout d'abord, je tiens à remercier Dieu de m'avoir donné beaucoup de volonté et de patience pour réaliser ce projet. Ensuite, Je tiens à adresser mes profonds remerciements à tous les professeurs qui m'ont transmis leur savoir, qui m'ont assuré un bon encadrement et qui ont toujours été là pour moi quand j'ai eu besoin d'aide. Sans bien sûr oublier de remercier le cadre administratif de l'ENSAM Casablanca pour cela. Finalement, je tiens à remercier le club A&M Mechatronics pour m'avoir fourni un certain bagage d'informations et qui m'a permis d'aboutir à un tel résultat.

Introduction Générale

L'automatisation consiste à rendre automatique les opérations qui exigeaient auparavant l'intervention humaine, elle est considérée comme l'étape d'un progrès technique ou apparaissent des dispositifs techniques susceptibles de seconder l'homme, non seulement dans ses efforts musculaires, mais également dans son travail intellectuel de surveillance et de contrôle.

Un automatisme est un sous-ensemble d'une machine destinée à remplacer l'action de l'être humain dans des tâches en générales simples et répétitives, réclamant précision et rigueur. On est passé d'un système dit manuel, à un système mécanisé, puis au système automatisé.

Dans l'industrie, les automatismes sont devenus indispensables : ils permettent d'effectuer quotidiennement les tâches les plus ingrates, répétitives et dangereuses. Parfois, ces automatismes sont d'une telle rapidité et d'une telle précision, qu'ils réalisent des actions impossibles pour un être humain. L'automatisme est donc synonyme de productivité et de sécurité.

Le savoir-faire de l'opérateur est transposé dans le système automatisé, il devient le PROCESSUS.

Un processus peut-être considéré comme un système organisé d'activités qui utilise des ressources (personnel, équipement, matériels et machines, matière première et informations) pour transformer des éléments entrants en éléments de sortie dont le résultat final attendu est un produit

Historique :

La fabrication de précision moderne exige une extrême précision dimensionnelle et la finition de surface. Une telle performance est très difficile à réaliser manuellement, voire impossible, même avec des opérateurs experts. Dans les cas où il est possible, il faut du temps beaucoup plus élevé en raison de la nécessité pour la mesure dimensionnelle fréquente pour prévenir la surexploitation. Il est donc évident que le contrôle de mouvement automatisé remplacerait le manuel de contrôle dans la fabrication moderne. Le développement de commande numérique par ordinateur (CNC) a également rendu possible l'automatisation des processus d'usinage avec souplesse pour gérer la production de petites et moyennes séries de pièces.

C'était en 1947, lorsque la commande numérique est née. Tout a commencé lorsque John C. Parsons de la Parsons Corporation, Traverse City, Michigan, un fabricant de pales de rotor d'hélicoptère, ne pouvait pas faire ses modèles assez vite. Donc, il a inventé une façon de coupler l'équipement informatique avec une pyrale du gabarit. M. Parsons a utilisé les cartes pour faire fonctionner son système de Ron chiffres de poing.

1949 a été l'année d'un autre « besoin urgent ». Le commandement américain Air Matériel s'est rendu compte que ses pièces pour avions et missiles ont été de plus en plus complexe. En outre, comme les dessins ont été constamment améliorés, les changements dans le dessin ont souvent été faites. Ainsi, dans leur recherche de méthodes de production plus rapide, un contrat d'étude de la Force aérienne a été attribué à la Parsons Corporation. Le laboratoire des mécanismes d'asservissement de l'Institut de Technologie du Massachusetts (MIT) était le sous-traitant.

En 1951, le MIT a repris le travail complet, et en 1952, le prototype de la machine NC d'aujourd'hui, une machine Cincinnati Hydrotel Milling modifiée, a été démontrée avec succès. La commande numérique terme a été l'origine au MIT.

Initialement, la technologie CNC a été appliqué sur les tours, fraiseuses, etc. qui pourrait effectuer un seul type d'opération de découpe de métal. Plus tard, on a tenté de traiter une variété de pièces qui peuvent nécessiter plusieurs types différentes opérations d'usinage et de les terminer en une seule mise en place. Ainsi centres d'usinage CNC capables d'exécuter plusieurs opérations ont été développés. Pour commencer, centres d'usinage CNC ont été développés pour les composants prismatiques usinage combinant des opérations comme fraisage, perçage, alésage et taraudage. Peu à peu, les machines pour la fabrication de composants cylindriques, appelés centres de tournage ont été développés.

Chapitre 1 : généralité sur les machines à commande numérique par ordinateur

I.1. Introduction

Dans ce chapitre on présente un aperçu général sur la technologie des CNC, citant la classification, l'architecture, la programmation des Machines CNC pour différentes commandes.

I.2. Définition de la commande numérique

Apparue il y a seulement quelques dizaines d'années, la commande numérique CN impose actuellement sa technologie dans le monde de l'usinage. conçue pour piloter le fonctionnement d'une machine à partir des instructions d'un programme sans intervention directe de l'opérateur pendant son exécution, elle a dans un premier temps, permis de franchir un pas important dans l'automatisation des machines-outils traditionnelles comme les fraiseuses, perceuses, aléseuses...etc. et elles sont ainsi devenue capable d'assurer en qualité et en quantité une production à peine imaginable quelques années auparavant.

Aujourd'hui, de plus en plus étroitement associé aux progrès de la microélectronique et de l'informatique, la CN voit ses performances augmenter régulièrement tandis que, son prix et son encombrement ne cessent de diminuer.

Elle pénètre, de ce fait dans les petites entreprises et devient accessible à tous les secteurs industriels faisant appel aux procédés de positionnement ou de suivi de trajectoire.

Les CN actuelles sont capables d'effectuer des mouvements en combinant simultanément les positions sur les axes X, Y et Z.

Elles s'appellent commande numérique par ordinateur. De plus, elles sont capables de gérer plusieurs axes simultanément.

I.3. Définition d'une machine CNC

C'est une machine totalement ou partiellement automatique à laquelle les ordres sont communiqués grâce à des codes qui sont portés sur un support matériel (disquette, USB, ...).

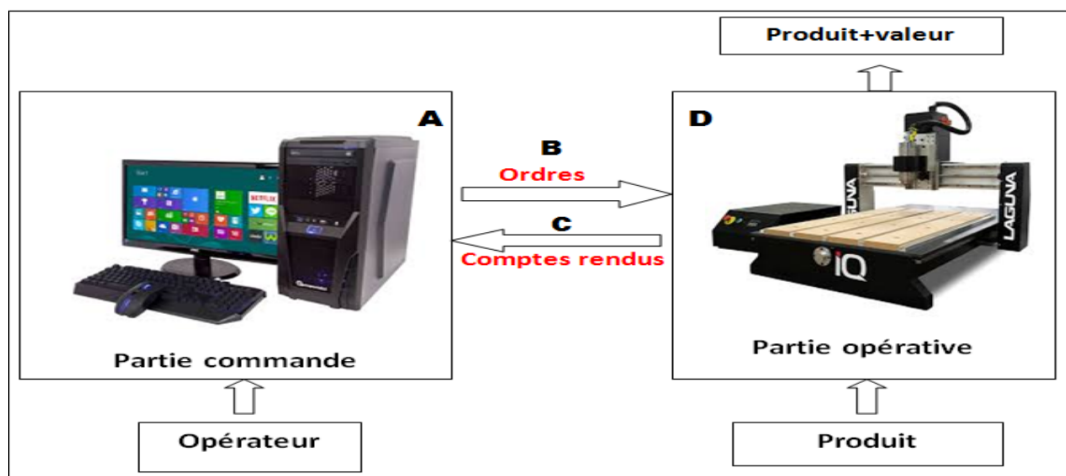
Lorsque la machine-outil est équipée d'une commande numérique capable de réaliser les calculs des coordonnées des points définissant une trajectoire (interpolation), on dit qu'elle est à ordinateur. Elle est appelée CNC (Commande Numérique par Ordinateur).

Le premier rôle d'une machine CNC (Computerized Numerical Control), est de générer des mouvements. Elle recevra des valeurs de positionnement, de vitesse et d'accélération et générera, suite à un traitement, des consignes numériques en sortie. Elle dispose d'une grande puissance de calcul et d'une plus grande souplesse d'utilisation qu'un automate programmable. Ce dernier est néanmoins qu'un complément de la commande CNC pour ce qui est de la gestion des entrées telles que : interrupteur, bouton d'arrêt d'urgence, ...etc..

En résumé, la commande CNC va générer des mouvements selon des consignes numériques.

La machine-outil à commande numérique forme un ensemble comprenant :

- La partie opérative.
- La partie commande



Partie A : L'unité centrale, le clavier, la souris, l'écran et un logiciel de CFAO constituent la partie commande.

Partie B : La partie commande ordonne (ordres) des actions sur le système et son environnement

Partie C : La partie opérative renvoie (comptes-rendus) les états du système et son environnement.

Partie D : Le bâti, l'outil, le montage d'usinage et la pièce à usiner constituent la partie opérative.
 Pour but de faire des usinages.

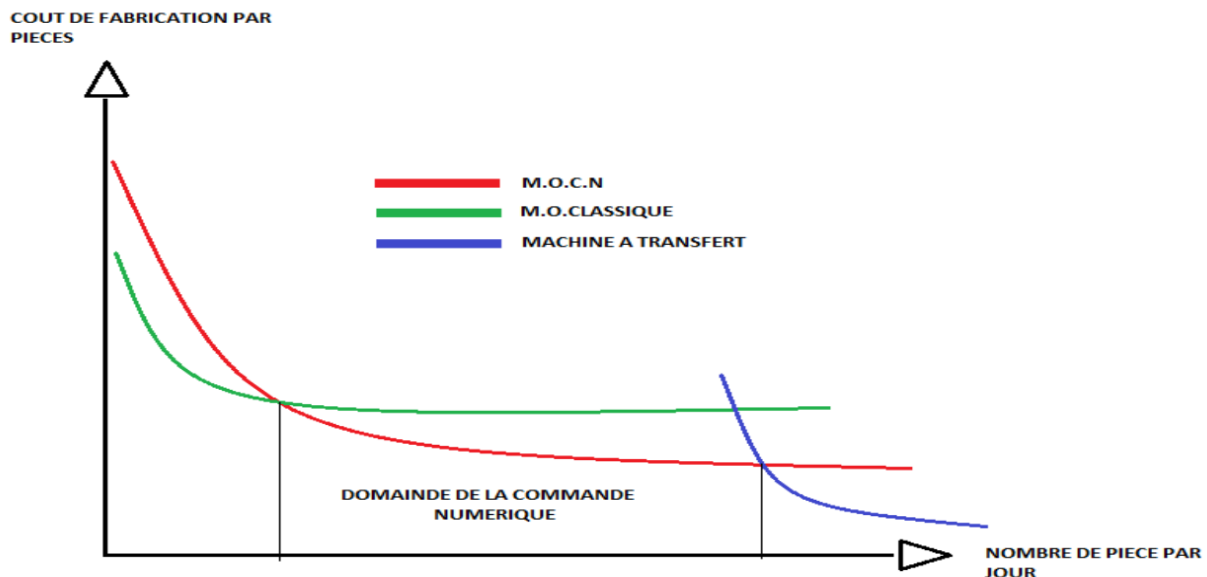
La démarche est la suivante : lorsqu'on a une idée précise de la forme de la pièce que l'on souhaite fabriquer, il faut :

- programmer, à l'aide d'un ordinateur relié à la machine-outil à commande numérique, la forme de la pièce (avec les différents usinages à réaliser : perçages, découpes...).
- définir les paramètres d'usinage : diamètre des outils, vitesse de coupe, origine de la pièce, ...
- transférer ces informations sur la machine.
- fixer la pièce sur la table de la machine.
- vérifier le respect de toutes les consignes de sécurité et lancer l'usinage de la pièce.

I.4. Domaine d'utilisation

Les Machines CNC conviennent à la fabrication en petites et moyennes séries renouvelables. Elles permettent la réalisation, sans démontage, de pièces complexes comportant beaucoup d'opérations d'usinage.

Elles se situent entre les machines conventionnelles très "flexibles" réservées aux travaux unitaires (prototypes, maintenance) et les machines transferts, très productives, réservées aux grandes séries.



Domaine d'utilisation des M.O.C.N

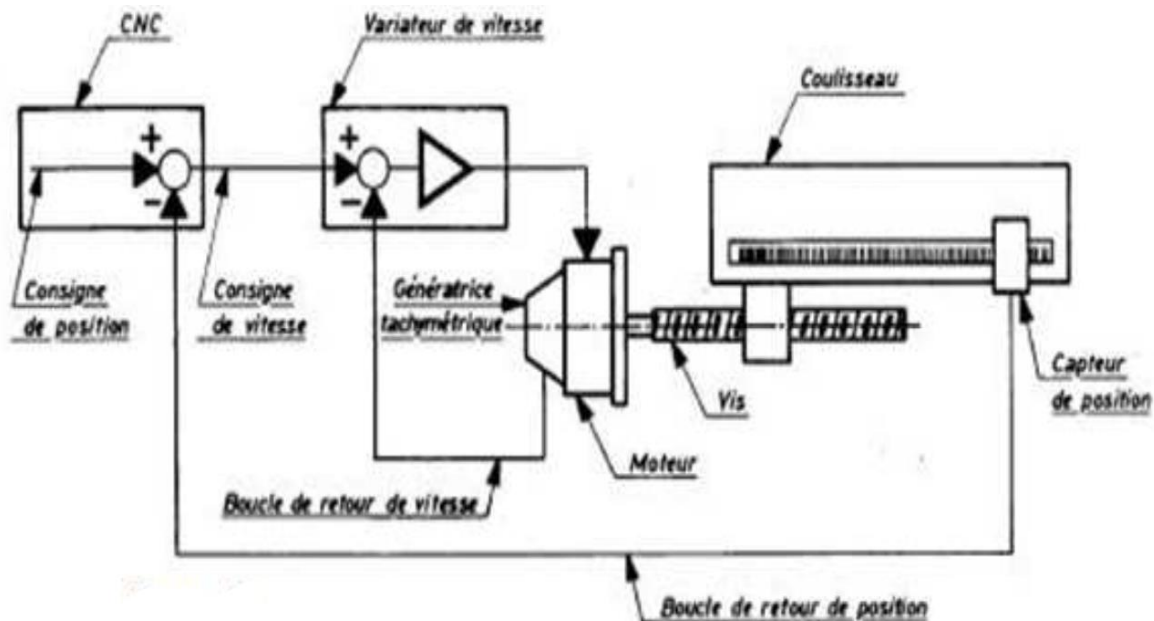
I.5. Principe de fonctionnement

La machine CNC travaille avec des systèmes de contrôles en boucle fermée.

Des ordres vont être générés vers la commande par le biais d'un programme pièce ou par action manuelle de l'opérateur.

La commande va traiter ces informations et générer des consignes afin d'obtenir les déplacements voulus par le biais des moteurs d'axes. Des contrôles de vitesse et de position seront alors effectués de manière continue par la machine.

La position sera réglée par la commande numérique alors que la vitesse sera le plus souvent réglée par le système d'asservissement moteur. On se trouve donc en face d'un système à deux boucles et l'on parle de système asservi.



Asservissement de la machine

I.6. Les différents types de Machines CNC

On distingue plusieurs types de machines :

les machines à enlèvement de copeaux : les perceuses, les tours, les centres de tournages, les fraiseuse, les centres d'usinage, les rectifieusesetc.

les électroérosions : les machines à enfonçages, les machines à fil.

les machines de découpes : oxycoupage, laser, jet d'eau...

les presses : métal, injection plastique.

les machines spéciales : à panneaux, à têtes multiples, de conditionnement (pour l'agroalimentaire)...

I.7. Classification des machines CNC

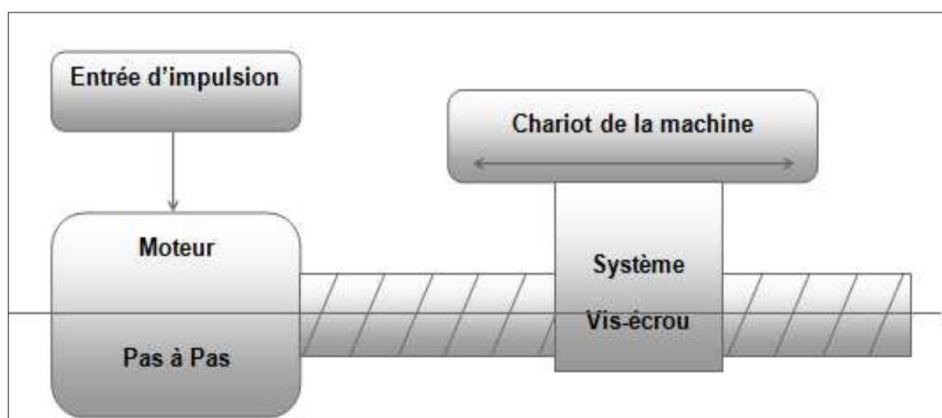
Les machines-outils à commande numérique par calculateurs (CNC) sont classées suivant :

- Le mode de fonctionnement de la machine.
- Le mode d'usinage.

I.7.1. Classification des CNC selon le mode de fonctionnement

a. Fonctionnement en boucle ouvert

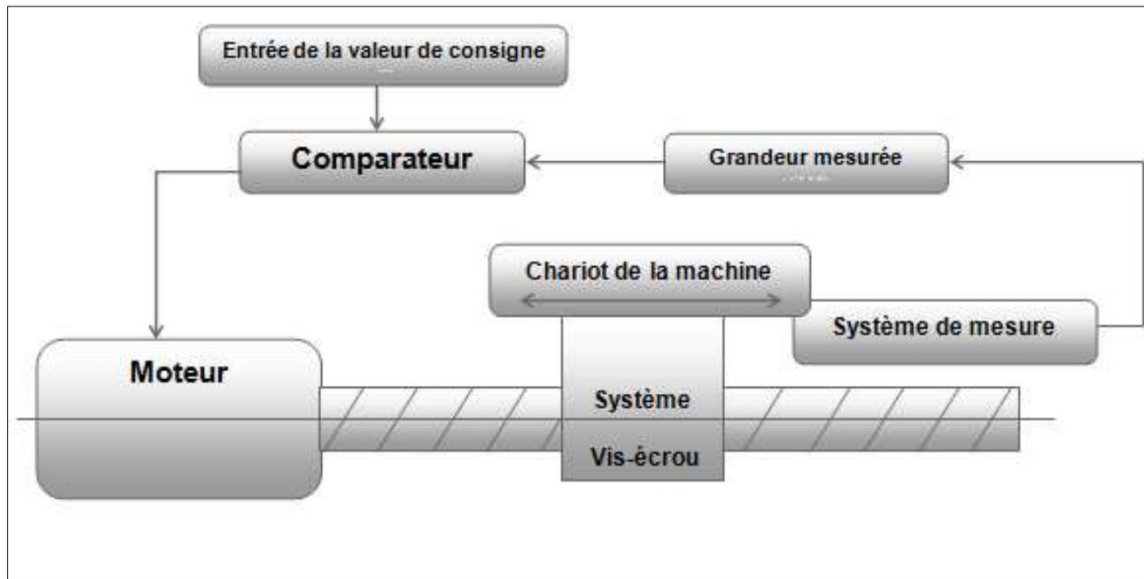
En boucle ouverte, comme l'illustre la figure. I.4, le système assure le déplacement du chariot mais ne le contrôle pas.



Fonctionnement en boucle ouvert

b. Fonctionnement en boucle fermé

En boucle fermée le système contrôle le déplacement ou la position jusqu'à égalité des grandeurs entrée (E) dans le programme et celui mesuré .



Commande en boucle fermée

I.7.2. Classification des CNC selon le mode d'usinage

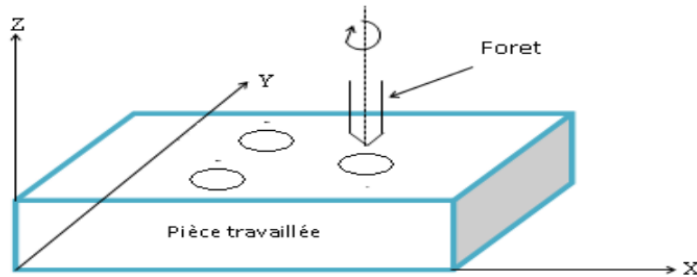
Selon le mode d'usinage on peut classer les CNC en trois catégories :

- Commande numérique point à point
- Commande numérique de contournage
- Commande numérique par axiale

1) Commande numérique point à point :

C'est la mise en position de l'outil ou de la pièce Par déplacements non synchronisés. Le mouvement de coupe (usinage) n'est possible que lorsque le mouvement de positionnement est effectué.

- Exemples d'opération d'usinage : perçage, alésage, lamage taraudage, petit fraisage.

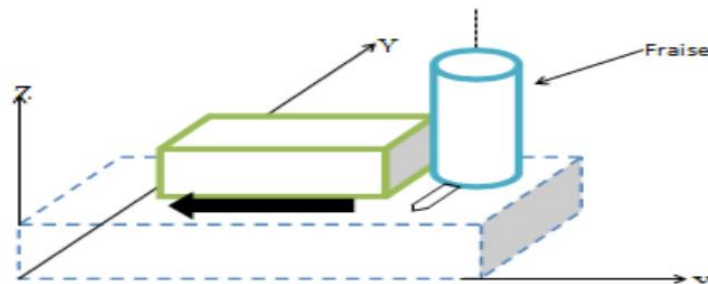


Commande Numérique Point À Point

2) Commande numérique par axiale :

Ce sont des déplacements parallèles aux axes avec les vitesses d'avance programmée. Le mouvement de coupe et de positionnement sont synchronisés de façon à avoir un usinage selon des trajectoires parallèles aux axes de déplacement.

- Exemples d'opération d'usinage : tournage, fraisage, alésage.

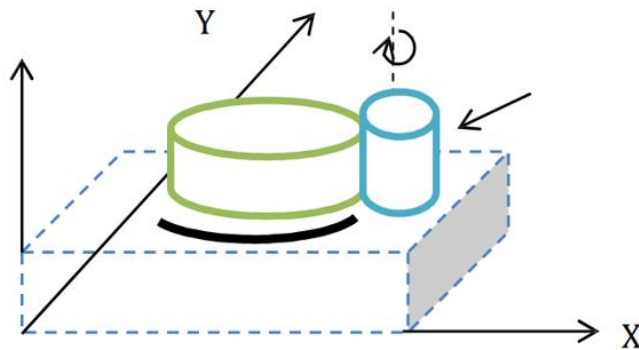


Commande Paraxiale

3) Commande numérique de contournage :

Ce sont des déplacements qui synchronise des divers axes avec la vitesse d'avancement programmée. Les trajectoires sont décomposées en éléments de droites ou de cercles dans un ou plusieurs plans.

-exemple d'opération d'usinage : toute opération possible sur un centre de tournage ou centre d'usinage.



Commande numérique de contournage

I.8. La structure d'une machine CNC

Ce système peut être divisé en trois parties. Le système mécanique qui reçoit des signaux de commandes nécessaires du système électronique qui aboutit finalement à actionnement souhaité des moteurs. Le système électronique obtient une commande ou un ensemble de commandes du système logiciel et génère des commandes pour le système mécanique.

I.8.1. Le système électronique

Le système électronique est responsable de la génération du signal de commande pour les moteurs qui guide le mouvement de la trajectoire de l'outil.

le système électronique est composé de :

- 1) L'alimentation
- 2) La carte microcontrôleur
- 3) La carte de commande des moteurs pas à pas

I.8.2. Le système mécanique

Le système mécanique est assemblé de telle manière que le mouvement des 3 axes soit réalisé en utilisant des tiges filetées avec des écrous, les moteurs sont montés chacun à l'axe qui est source de mouvement agi en fonction du signal de commande généré à partir du circuit électronique.

Chaque moteur est relié à une vis sans fin de chaque axe qui est chargé de transformer le mouvement du moteur de rotation en mouvement linéaire.

Le mouvement contrôlé de chaque axe est obtenu directement par la commande de la rotation du moteur. La vitesse du mouvement de chaque axe peut également être contrôlée par le contrôle direct de la vitesse du moteur, en donnant des signaux de commande nécessaires. Ainsi, la trajectoire de l'outil fixée à l'organe terminal est contrôlée dans chaque axe pour une action sur la pièce à usiner.

I.8.3. Le système de programmation

La pièce à usiner sera conçue dans une conception assistée par ordinateur (CAO), dont la sortie est un dessin dans un des nombreux formats acceptables, le format le plus préférable est le format .stl. Ce dessin est ensuite introduit dans le logiciel de fabrication assistée par ordinateur (FAO), dont la sortie est le code lisible par une machine utilisée pour la commande numérique de la machine.

Depuis la mise en œuvre du Gcode, il est nécessaire de tester de différents choix pour un interprète open source Gcode pour l'Arduino, de sorte que les mouvements corrects seront obtenus par les axes de la machine par l'intermédiaire du pilote de moteur.

I.9. Conclusion

Dans ce chapitre on a décortiqué d'une manière générale une machine à commande numérique par ordinateur et l'intérêt de chaque partie qui constitue cette dernière, on a cité les commandes les plus utilisées et parler sur les classifications des machines CNC, Nous pouvons ainsi entamer la prochaine étape qui consiste à présenter la phase de conception.

Chapitre 2 : conception d'une machine CNC

II.1. Introduction

Le développement majeur dans la technologie informatique et la disponibilité du matériels sources vont nous permettre de fabriquer une machine CNC à faible coût.

Le travail est reparti en trois parties comme suit :

La partie électronique, la partie mécanique et la partie logiciel et programmation.

II.2. Cahier de charge

Notre projet consiste à la conception d'une machine à commande numérique (mini fraiseuse 3 axes).

Cette machine a été réalisée pour travailler sur des matériaux tendres comme le bois, les matières plastiques, en aucun cas, elle ne peut usiner des matériaux durs comme l'acier.

Les spécifications de machine :

spécification technique

- Tension d'alimentation 220V /50H
- Déplacement maximal :
Suivant l'axe x (450mm)
Suivant l'axe y(250mm)
Suivant l'axe z(120mm)

spécification économique

- Cout : réduit au maximum, utilisation des pièces normalisées si c'est possible.

spécification opérationnel

- Montage et démarrage faciles
- Manipulation facile

II.3 Partie mécanique

Conception de la machine avec logiciel CATIA V5

La conception du support a été réalisée par une conception CAO sous le logiciel CATIA V5.

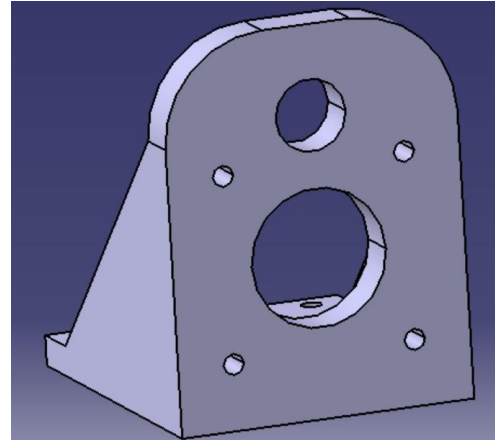
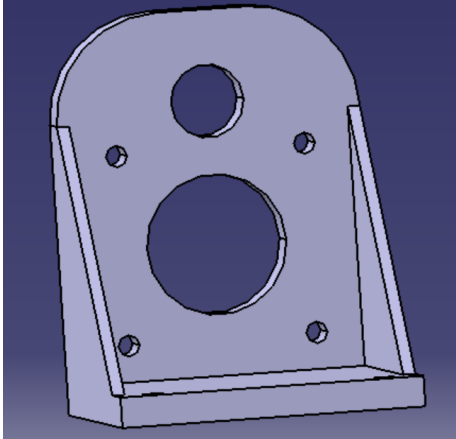
Dans ce paragraphe on va traiter la méthode de la conception en 3D du machine cnc en donnant tous les détails ainsi que les différentes étapes.

Notre machine cnc est conçu selon des pièces standard et des pièces à usiner.

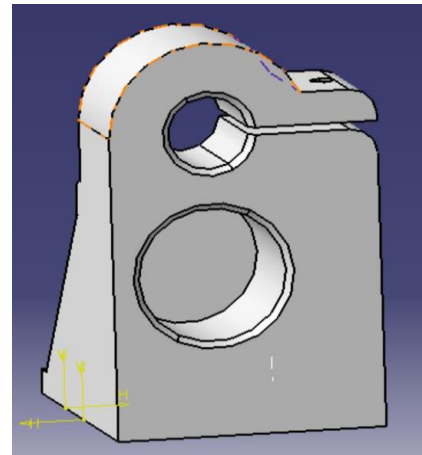
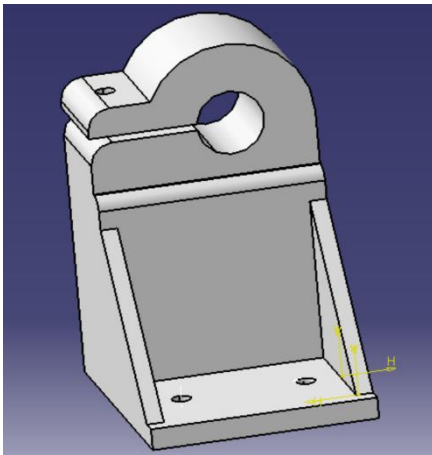
II.3.1. Conception des pièces à usiner

a) Deux support moteur (axe X)

Il assure la fixation du moteur

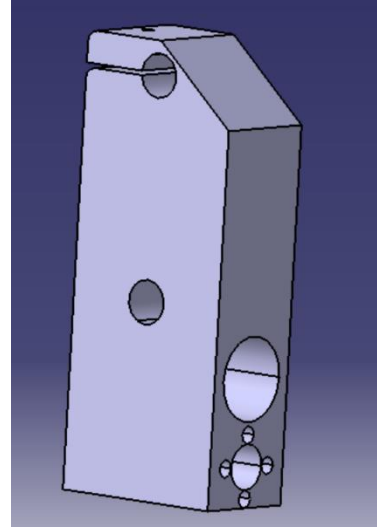
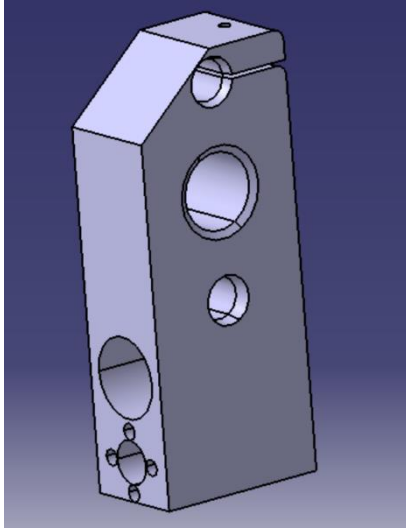


b)

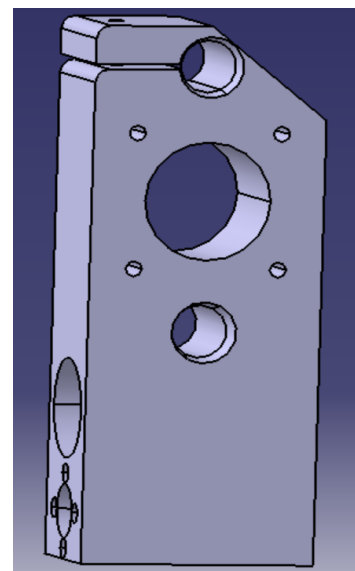
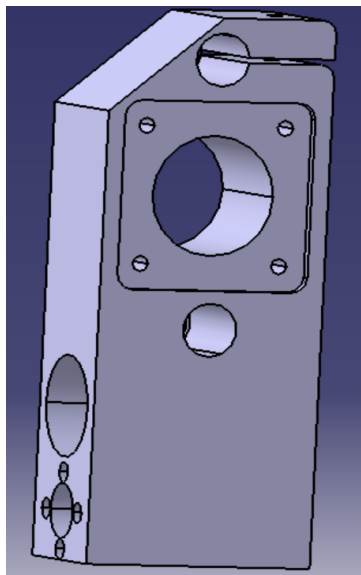


c) Un support axe X

Cette pièce va porter le support Y et fait la translation suivant X .

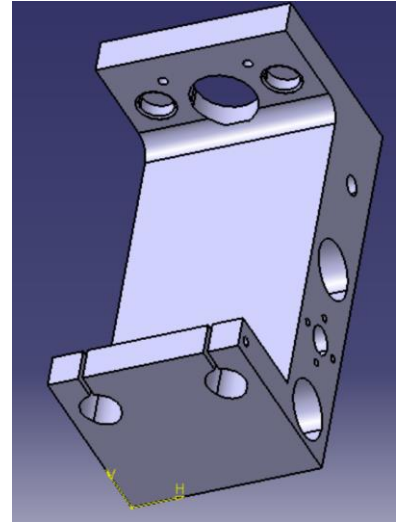
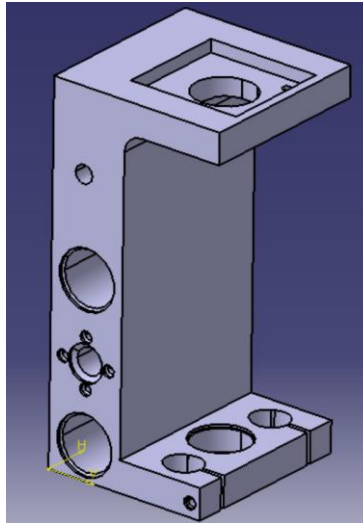
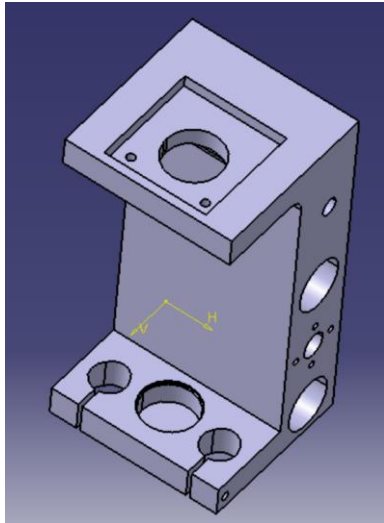


d) Un support axe X (il porte le moteur qui assure la translation suivant Y)



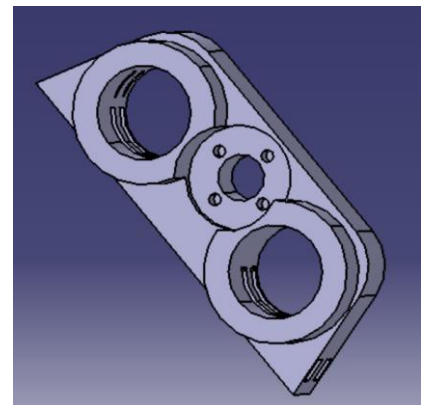
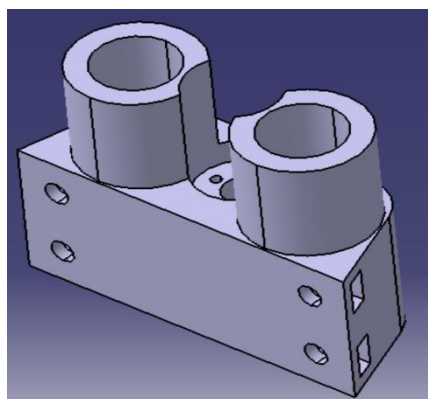
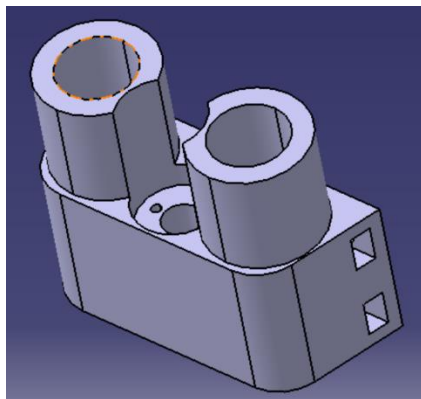
e) Un support axe Y

Cette pièce porte le support Z et fait la translation suivant Y



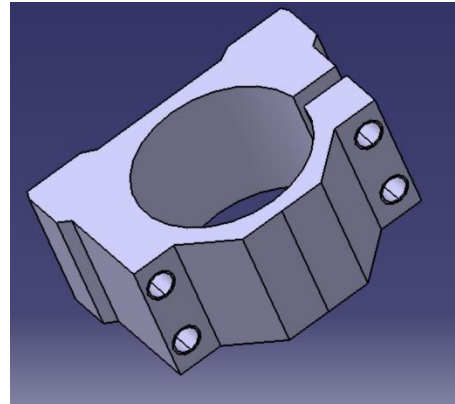
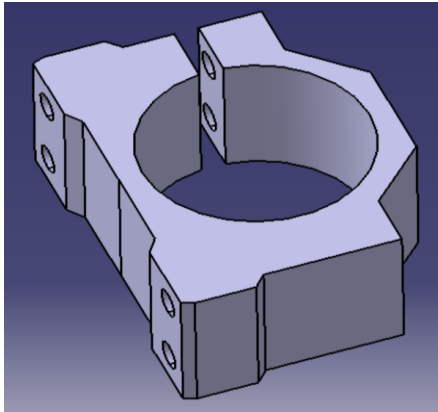
f) Un support porte outil

Cette pièce porte le support de l'outil (mini Perseus) et fait la translation suivant Z

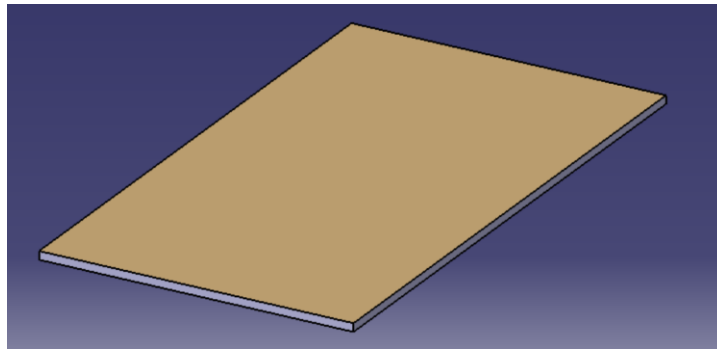


g) Porte outil



Il porte la Perseus







h) Table de fixation de pièce



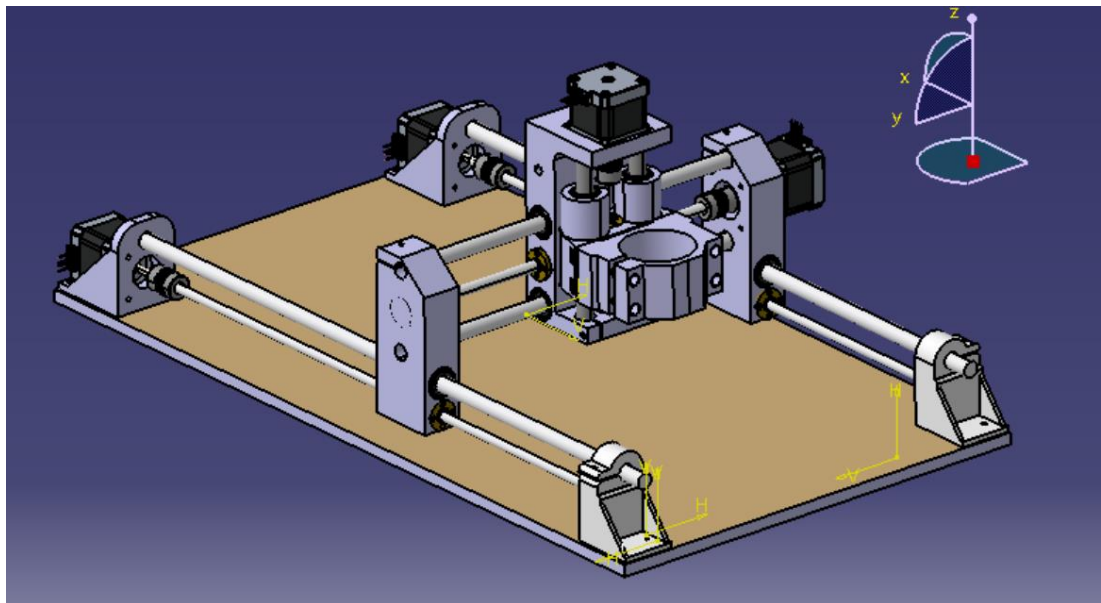
II.3.2. Les pièces standards :

Nom du pièce	Quantités	Le rôle	Figure
Coupleurs	4	Accoupler l'arbre moteur avec les tiges filetées	
Écrou en bronze (T8 pitch 2 mm)	4	Transformer le mouvement de rotation en mouvement de translation	

Douille linéaire	6	Assurer le guidage en translation	
Roulement	4	Permettre à une pièce de tourner par rapport à une autre selon un axe de rotation défini avec une très faible résistance au pivotement	
Tige fileté	4	Assurer la transmission de translation le l'écrou	
Axe de guidage	6	Assurer le mouvement de translation sur les 3 axes	

II.3.3. Assemblage de la machine CNC :

Après avoir conçu nos pièces on passe à la partie d'assemblage pour assurer le bien fonctionnement de la machine et d'éviter l'erreur de la fabrication.



II.3.4. Dimensionnement des vis trapézoïdale

Calcule du rendement

nous avons choisis une vis trapézoïdale :

- De pas=2 mm
- De diamètre = 8mm

Le rendement en général est : 25%

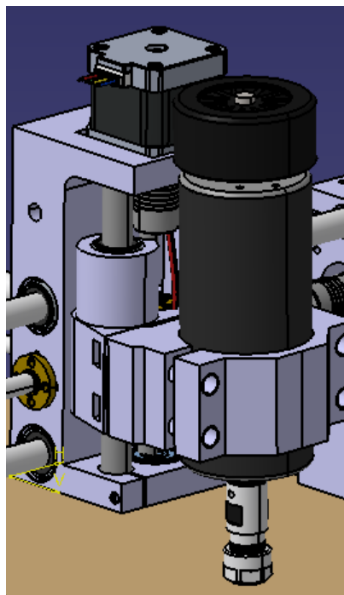
Détermination des masses

Dans cette partie on va calculer la masse de chaque élément de notre machine. Nous avons utilisé deux types de pièces, normalisées et des pièces qui seront usinées. Les masses des pièces normalisées sont tirées d'après les catalogues des fournisseurs. Les masses des pièces a usinées sont calculées approximativement par le logiciel de conception CatiaV5 . Le tableau suivant représente les masses des pièces de notre dispositif :

Déplacements	Nom de pièce	mi	Masse (Kg)
Direction Z	Moteur	m1	0.35
	Axes de guidages	m2	0.168
	Vis trapézoïdale+ écrou	m3	0.13
	Porte outil	m4	0.114
	Support porte outil	m5	0.11
	Mini Perseus	m6	0.4
	Douilles linéaires	m7	0.06
	Accouplement	m8	0.013
	Roulement	m9	0.0124
Direction Y	Moteur	m10	0.35
	Axes de guidages	m11	0.533
	Vis trapézoïdale + écrou	m12	0.175
	Douilles linéaires	m13	0.063
	Support axe Y	m14	1.2

	Accouplement	m15	0.013
Direction X	Moteurs	m16	0.7
	Axes de guidages	m17	0.9324
	Vis trapézoïdale + écrou	m18	0.246
	Douilles linéaires	m19	0.063
	Supports axe X	m20	0.294
	Accouplements	m21	0.026
	Roulements	m21	0.0248

II.3.4.1. Dimensionnement de la vis trapézoïdale Z



Système de mouvement Z

a) La charge appliquée sur la vis

F la charge appliquée sur la vis trapézoïdale de l'axe Z est calculé par la formule suivante :

$$F_z = (m \times g)$$

Ou g : la gravité

m : l'ensemble des masses sur le vice

on a : $m = m_4 + m_5 + m_6 + m_7 + m_9 = 0.696 \text{ Kg}$

Donc : $F_z = 6.96 \text{ N}$

b. Le couple nécessaire

Il est donné par la formule suivante :

$$C_z = (F_z \cdot p) / (2 \cdot \eta \cdot \pi \cdot 10^3) \quad \text{Avec : } p : \text{ pas de la vis} = 2 \text{ mm}$$

$$C_z = (6.96 \cdot 2) / (2 \cdot 3,14 \cdot 0.25 \cdot 10^3)$$

$$C_z = 8.86 \cdot 10^{-3} \text{ N.m}$$

c. La puissance nécessaire

$$P_z = C \cdot \omega$$

$$\omega = (2 \cdot \pi \cdot v / 60)$$

Supposons que $v = 1000 \text{ tr/min}$

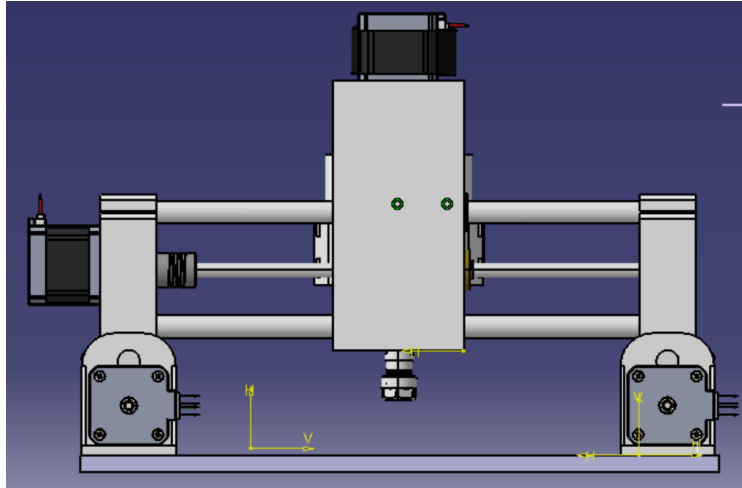
$$\omega = (2 \cdot 3,14 \cdot 1000 / 60)$$

$$\omega = 104.72 \text{ rd / s}$$

$$P_z = 8.86 \cdot 10^{-3} \cdot 104.72$$

$$= 0.927 \text{ watt}$$

II.3.4.2. Dimensionnement de la vis trapézoïdale Y



Système de mouvement Y

a) La charge appliquée sur la vis

F_Y la charge appliquée sur la vis trapézoïdale de l'axe Y est calculé par la formule suivante :

$$F_Y = (m \times g)$$

Où g : la gravité

m : l'ensemble des masses sur le vice

$$\text{on a : } m = m_1 + m_2 + m_3 + m_4 + m_5 + m_6 + m_7 + m_8 + m_9 + m_{13} + m_{14} = 2.62 \text{ Kg}$$

$$\text{Donc : } F_Y = 26.2 \text{ N}$$

b. Le couple nécessaire

Il est donné par la formule suivante :

$$C_Y = (F_Y \cdot p) / (2 \cdot \eta \cdot \pi \cdot 10^3)$$

Avec : p : pas de la vis = 2 mm

$$C_Y = (26.2 \cdot 2) / (2 \cdot 3,14 \cdot 0.25 \cdot 10^3)$$

$$C_Y = 0.033 \text{ N.m}$$

c. La puissance nécessaire

$$P_Y = C * \omega$$

$$\omega = (2 * \pi * v / 60)$$

Supposons que $v = 1000$ tr/min

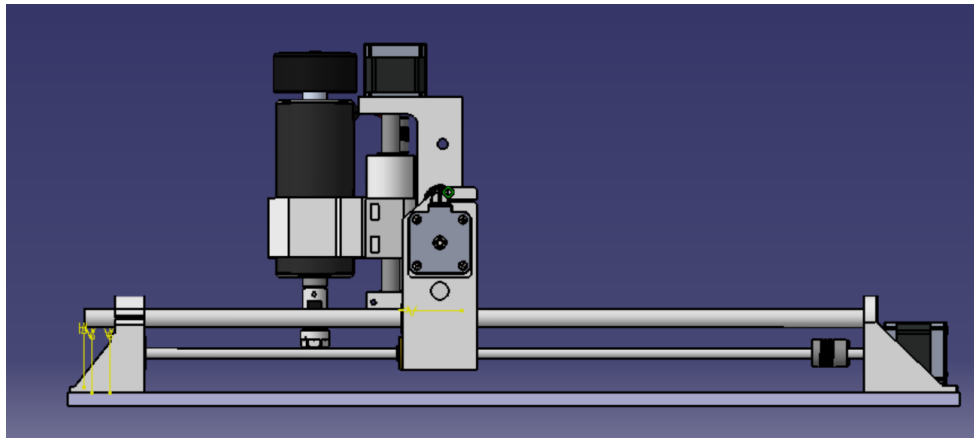
$$\omega = (2 * 3,14 * 1000 / 60)$$

$$\omega = 104.72 \text{ rd / s}$$

$$P_Y = 0.033 * 104.72$$

$$= 3.456 \text{ watt}$$

II.3.4.3. Dimensionnement de la vis trapézoïdale X



Système de mouvement X

a) La charge appliquée sur la vis

F_X' la charge appliquée sur les deux vis trapézoïdales de l'axe X est calculé par la formule suivante :

$$F_X' = (m \times g)$$

Ou g : la gravité

m : l'ensemble des masses sur le vice

on a : $m = m_1 + m_2 + m_3 + m_4 + m_5 + m_6 + m_7 + m_8 + m_9 + m_{10} + m_{11} + m_{12} + m_{13} + m_{14} + m_{15} + m_{19} + m_{20} = 4.048 \text{ Kg}$

Donc : $F_x' = 40.48 \text{ N}$

Donc la charge appliquée sur un vis est :

$$F_x = F_x' / 2 = 20.24 \text{ N}$$

b. Le couple nécessaire

Il est donné par la formule suivante :

$$C_x = (F_x \cdot p) / (2 \cdot \eta \cdot \pi \cdot 10^3) \quad \text{Avec : } p : \text{pas de la vis} = 2 \text{ mm}$$

$$C_x = (20.24 \cdot 2) / (2 \cdot 3,14 \cdot 0.25 \cdot 10^3)$$

$$C_x = 0.025 \text{ N.m}$$

c. La puissance nécessaire

$$P_x = C \cdot \omega$$

$$\omega = (2 \cdot \pi \cdot v / 60)$$

Supposons que $v = 1000 \text{ tr/min}$

$$\omega = (2 \cdot 3,14 \cdot 1000 / 60)$$

$$\omega = 104.72 \text{ rd / s}$$

$$P_x = 0.025 \cdot 104.72$$

$$= 2.7 \text{ watt}$$

II.4. La partie électronique

II.4.1. Introduction

Après avoir décrit la partie mécanique, nous allons maintenant entamer la partie électronique qui va commander tout le système. Commençons tout d'abord par une description théorique des différents composants qu'on va utiliser pour pouvoir réaliser notre commande.

II.4.2. Moteur pas à pas

II.4.2.1. Généralité sur les moteurs pas à pas

Les moteurs pas à pas sont très utilisés dans toutes les applications mécaniques où l'on doit contrôler simplement la position ou la vitesse d'un système. Ces moteurs sont par exemple utilisés dans les imprimantes jet d'encre ou laser, pour positionner les têtes d'impression ou pour l'avancée du papier.

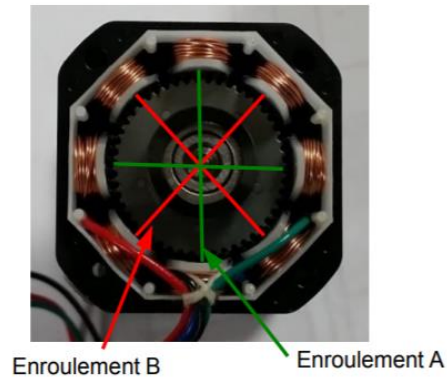
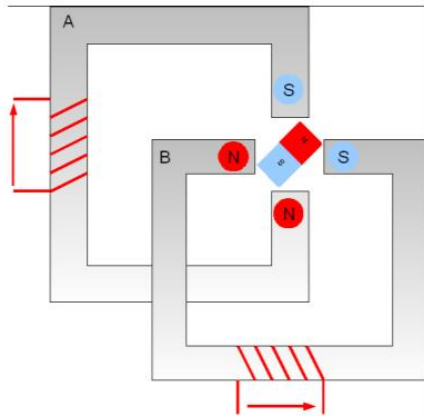
La commande des moteurs pas à pas est relativement simple car on n'a pas besoin d'accessoires tels que des codeurs pour connaître la position, chaque impulsion du système de commande les fait avancer d'un pas. Il existe différents types de moteurs pas à pas (aimants permanents, à reluctance variable ou hybrides), pour notre application on va travailler avec moteur à aimant permanent bipolaire.

II.4.4.2. Moteur à aimant permanent bipolaire

i) Définition

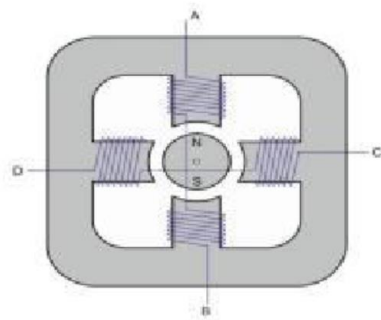
Dans la famille des moteurs pas à pas à aimants permanents, on distingue les moteurs pas à pas bipolaires qui contiennent deux enroulements (bobines). Chaque enroulement est commandé en courant positivement ou négativement. Le rotor aimant possède plusieurs pôles Nord-Sud. Ces moteurs possèdent 4 fils (2 par bobine).

La photo de la structure interne du moteur pas à pas Nema 17 montre deux enroulements repartis en croix ainsi qu'un rotor constitué de nombreux pôles.



Représentation schématique simplifier d'un moteur pas à pas Bipolaire

ii) fonctionnement du pas complet



Dans le mode pas complet (monophasé), le couple n'est pas important puisqu'un seul enroulement est alimenté pour effectuer un pas. La première consiste à alimenter les enroulements en suivant la séquence A vers B/C vers D/B vers A/D vers C (BA est les mêmes enroulements que AB mais alimenté par un courant de polarité inverse). Par la suite nous simplifierons la notation pour une meilleure correspondance avec les chronogrammes des phases en indiquant uniquement la phase qui est alimentée par un courant « positif » Soit A B C D. Cette séquence est connue sous le nom de "one phase on full step" (traduisez phase par phase ou une phase à la fois en pas entier).

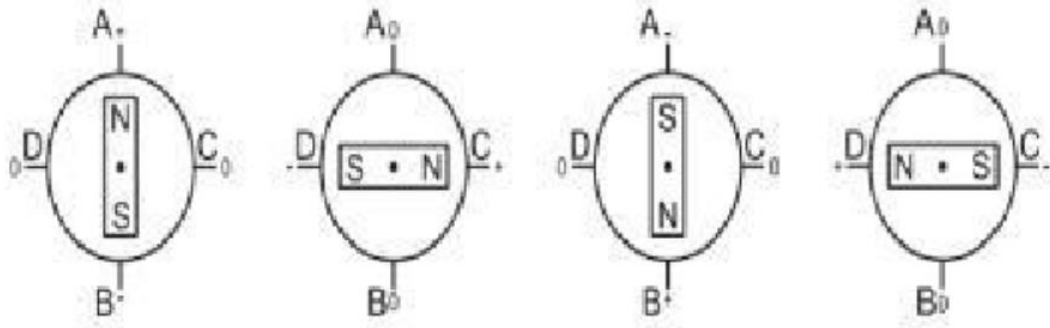


Figure : fonction a pas complet

iii) Fonctionnement avec couple maximal

C'est dans le mode couple maximal (biphase) que le moteur développera la plus grande puissance (couple élevé) car les deux phases seront alimentées en même temps de façon à ce que le rotor se positionne entre deux pôles. La séquence sera donc AC/CB/BD/DA.

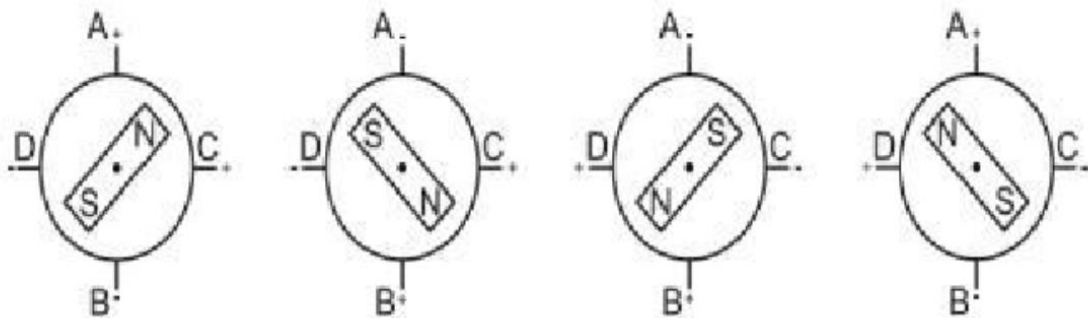


Figure : fonctionnement avec couple maximale

iv) fonctionnement a demi pas

Le fonctionnement à demi-pas est un mélange des deux premières puisque l'on alimente tour à tour le moteur sur une phase puis deux puis une ... cette séquence connue sous le nom de mode demi-pas procure effectivement une division par 2 de l'angle d'avance d'un pas, mais aussi un couple moins régulier.

La séquence qui en découle est la suivante : A/AC/C/CB/B/BD/D/DA.

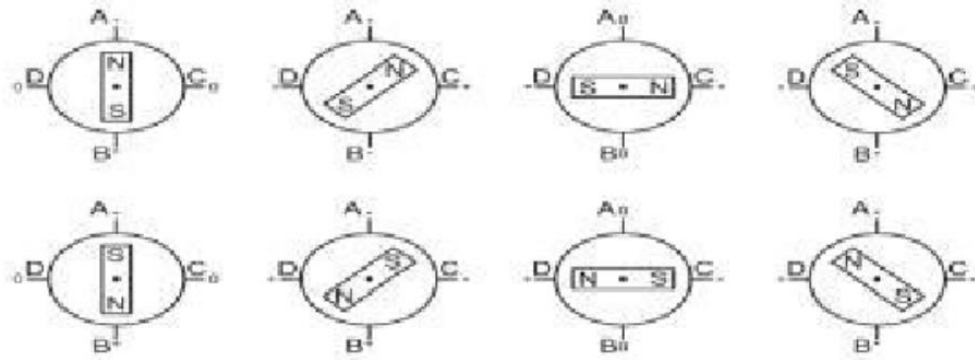


Figure : Fonctionnement a demi-pas

Pour obtenir une rotation dans la direction opposée les mêmes séquences sont utilisées en inversant l'ordre de défilement.

II.4.3. La carte de commande Arduino

a) Mise en œuvre de la carte Arduino Uno

La carte Arduino est une carte électronique basée autour d'un microcontrôleur et de composants minimum pour réaliser des fonctions plus ou moins évoluées à bas cout. Elle possède une interface USB pour la programmer. C'est une plateforme open -source qui est basée sur une simple carte a microcontrôleur (de la famille AVR), et un logiciel, véritable environnement de développement intégré, pour écrire, compiler et transférer le programme vers la carte a microcontrôleur.

Arduino est peut-être utilise pour développer des applications matérielles industrielles légères ou des objets interactifs, et il peut recevoir en entrées une très grande variété de capteurs.

Les projets Arduino peuvent être autonomes, ou communiquer avec des logiciels sur un ordinateur (Flash, Processing ou MaxMSP, Labview). Les cartes électroniques peuvent être fabriquées manuellement ou bien être achetées préassemblées, le logiciel de développement open-source est téléchargeable gratuitement.

La programmation de la carte Arduino présente les principales fonctionnalités de l'interface

de l'application Arduino. L'application Arduino vous permet de créer et éditer un programme (appelé sketch) qui sera compilé puis téléversé sur la carte Arduino. Ainsi, lorsque vous apportez des changements sur le code, ces changements ne seront effectifs qu'une fois le programme téléversé sur la carte.

b) présentation et caractéristiques du module Arduino Uno

La carte Arduino Uno, est une carte a microcontrôleur basée sur un Atmega328p.

Cette carte dispose :

- ✓ Broches numériques d'entrées/sorties
- ✓ Entrées analogiques
- ✓ Quatre UART (port série matériel)
- ✓ Quartz 16Mhz
- ✓ Connexion USB
- ✓ Connecteur d'alimentation jack
- ✓ Connecteur ICSP
- ✓ Bouton de réinitialisation

Elle contient tout ce qui est nécessaire pour le fonctionnement du microcontrôleur. Pour pouvoir l'utiliser et se lancer, il suffit simplement de la connecter a un ordinateur à l'aide d'un câble USB (ou de l'alimenter avec un adaptateur secteur ou une pile, mais ceci n'est pas indispensable, l'alimentation est fournie par le port USB).

La carte Arduino Uno dispose de toute une série de facilites pour communiquer avec un ordinateur, une autre carte Arduino, ou avec d'autres microcontrôleurs. L'Arduino Uno se dispose de quatre UARTs (Universal Asynchrones Receiver Transmitter ou emetteur-recepteur asynchrone universel en français) pour une communication en série de niveau TTL(5V) et qui est disponible sur les broches 0 (RX) et 1 (TX). Un circuit intégré Atmega16U2 sur la carte assure la connexion entre cette communication série de l'un des ports série de l'Atmega Uno vers le port USB de l'ordinateur qui apparait comme un port COM virtuel pour les logiciels de l'ordinateur. Le code utilise pour programmer l'Atmega8U2 utilise le driver standard USB COM, et aucun autre driver externe n'est pas nécessaire.

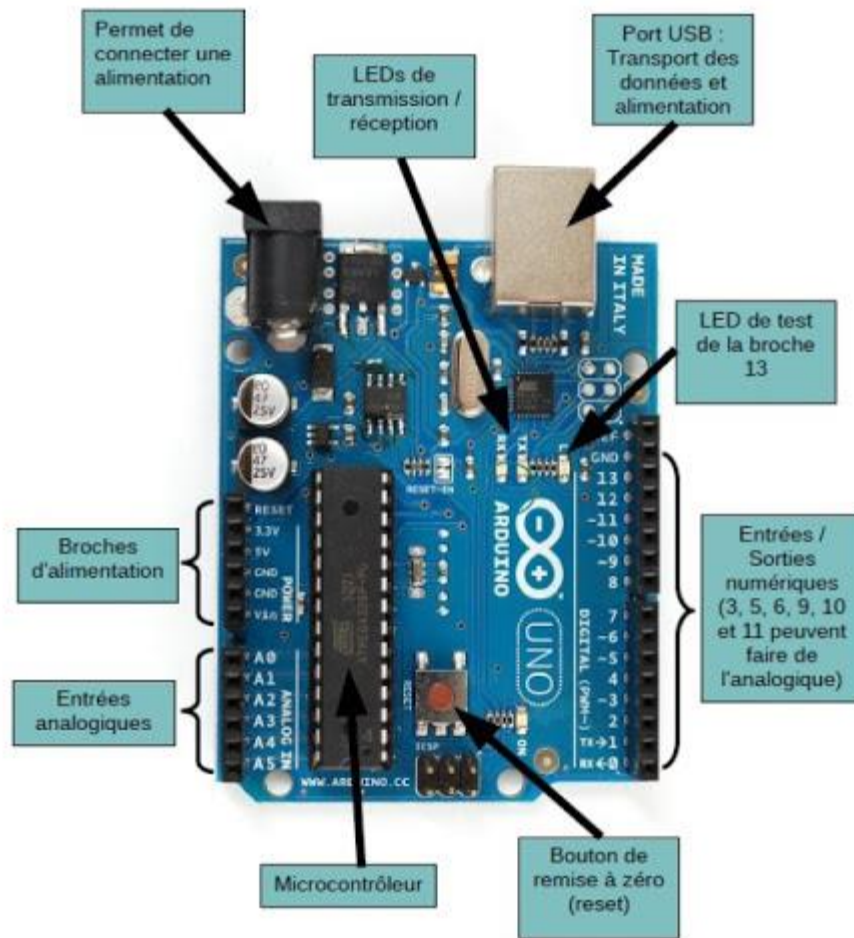


Figure : Module ARDUINO UNO

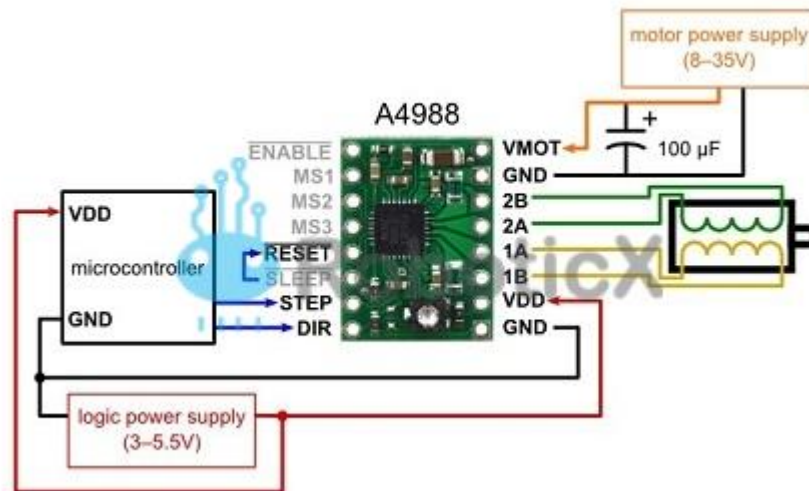
II.4.4. steppers motor driver A4988

a) Définition :

Le A4988 est un circuit électronique permettant de contrôler un moteur pas à pas avec différents modes. Son interface de puissance est assurée par des pentes en H réalisé par des Mosfets N-channel.

b) Fonctionnement :

Le driver A4988 permet non seulement de fournir la puissance nécessaire à la commande des bobines (hacheur intégré) mais également de simplifier le pilotage et choisir facilement le mode de pas à pas. Il offre ainsi non seulement la possibilité de travailler en pas complet et demi-pas mais également deux autres modes : 1/4 de pas et 1 /8e de pas en autorisant des positions intermédiaires dans un pas. Cela est rendu possible en modulant intelligemment la quantité de courant dans les bobines du moteur pas-a-pas. Par exemple, piloter un moteur en mode « 1/4 de pas » permet d'obtenir 800 micro-steps (micro-pas) sur un moteur prévu pour 200 pas par révolution et cela en imposant 4 niveaux de courants différents pour chacun des micro-pas. Le driver est donc équipé d'un asservissement de courant pour gérer correctement celui-ci dans les bobines. Un potentiomètre permet de définir la limite de courant (ou consigne de courant maximale) qui doit être choisie en fonction de la charge entraînée par le moteur et de la vitesse souhaitée.



Circuit de contrôle d'un moteur à l'aide du A4988

La configuration du driver se fait à l'aide des broches MS1, MS2 et MS3.

Le câblage du driver est fourni ci-après. Outre l'alimentation de la carte en 5V, le raccordement des 4 fils du moteur pas à pas et l'alimentation de puissance, on trouve plusieurs broches à raccorder au microcontrôleur :

- **ENABLE** : entrée de validation, active à l'état logique haut permettant de commander l'alimentation de la carte (sinon le moteur reste maintenu dans un pas donné par les bobines et consomme du courant).

- **DIR** : permet de spécifier le sens de rotation du moteur

- **STEP** : permet de spécifier le nombre de pas ou micro-pas à effectuer (le déplacement effectif du moteur dépendra du mode choisi pas entier, demi-pas etc.)

- **MS1** a **MS3** permettent de choisir le mode de fonctionnement du moteur. Ces entrées disposent de résistance de tirage à l'état bas (pull-down) internes, ce qui signifie que si l'on ne place pas le potentiel de ces broches au niveau logique haut = High = VDD(avec VDD la tension choisie pour la logique de commande) alors elles seront automatiquement ramenée au niveau logique bas = Low. Le tableau suivant permet de comprendre le mode de pas choisi en fonction des broches MS1,MS2,et MS3 .

MS1	MS2	MS3	Résolutions de pas
BAS	BAS	BAS	Pas entiers
HAUT	BAS	BAS	Demi-pas
BAS	HAUT	BAS	Quart de pas
HAUT	HAUT	BAS	Huitième de pas
HAUT	HAUT	HAUT	Seizième de pas

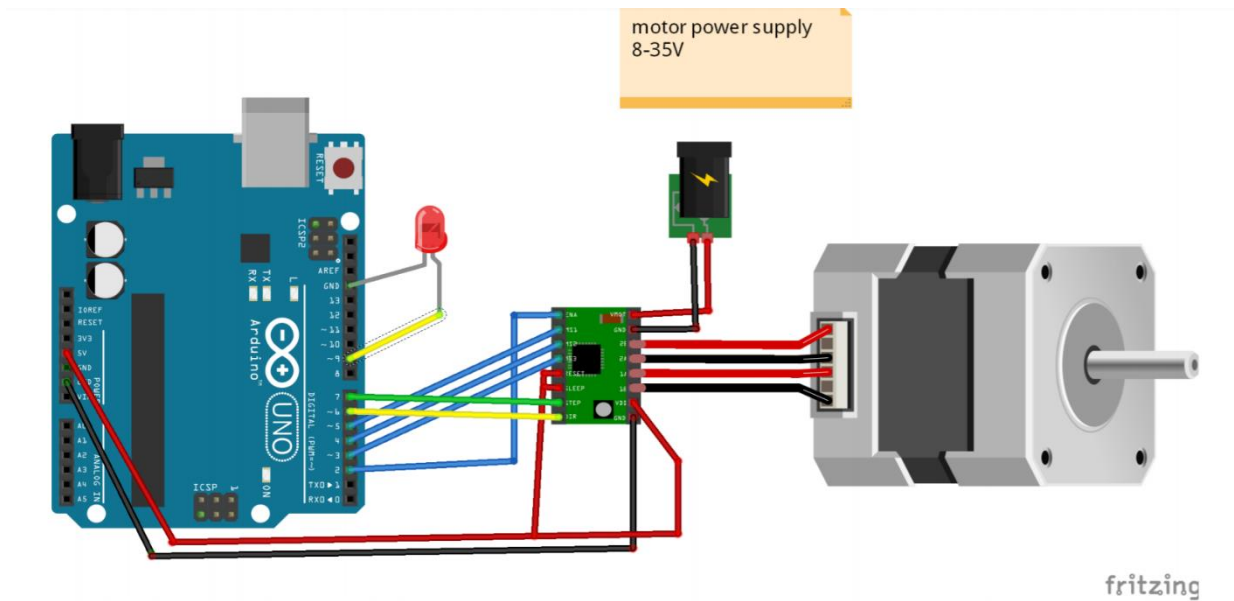


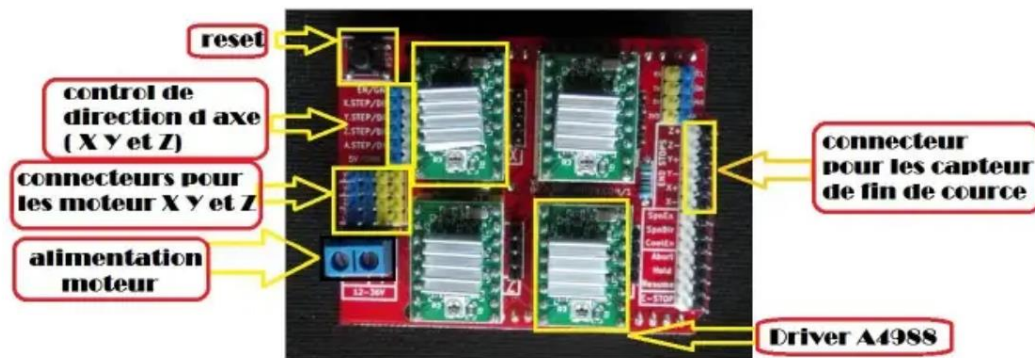
Figure : Câblage d'un moteur pas à pas avec le driver et l'Arduino

II.4.5. La carte SHIELD CNC

Le CNC-Shield est une carte électronique de "connectique" qui s'imbrique directement sur l'Arduino Uno. Il y a un emplacement prévu pour 4 contrôleurs moteurs du même type que le driver Motors stepper A4988.

Cette carte dispose :

- ✓ Emplacement pour 3 étages moteurs pour axes XYZ
- ✓ 1 emplacement pour un étage pour axe supplémentaire (dédoublage d'un des XYZ ou 4ème axe)
- ✓ 1 broche PWM pour la Perseus ou Laser
- ✓ Entrées pour 3 capteurs de fin course



Il suffit donc de clipser le contrôleur moteur et de connecter le moteur grâce aux pins situés à côté du contrôleur pour simplifier le câblage est d'utiliser un Shield CNC . Il est très pratique car il se fixe directement sur l'Arduino.

II.5. La partie Informatique et Programmation

II.5.1. Introduction

Les machines-outils à commande numérique sont complètement pilotées par ordinateur.

Pour que la machine comprenne les instructions données par l'ordinateur de contrôle, il faut que ces instructions soient transmises dans un langage de programmation spécial, qui est le G-code, ou code-G. Les outils informatiques actuels de la chaîne soft CAO permettent de se passer de la connaissance pratique du langage, mais pour ceux qui voudraient éventuellement faire des programmes simples directement avec un éditeur pour les envoyer dans le contrôleur ARDUINO, il est important de connaître quelques bases de ce langage. **Inkscape** est un logiciel de dessin vectoriel qui offre une interface graphique de gestion des images vectorielles

III.5.2. Langue de contrôleurs numériques « G-code »

III.5.2.1. Description

Le G-code est le langage de programmation des machines numériques est basé sur des lignes de code. Chaque ligne (également appelée un *bloc*) peut inclure des commandes pour faire produire diverses actions à la machine. Plusieurs lignes de code peuvent être regroupées dans un fichier

pour créer un programme G-code. Une ligne de code typique commence par un numéro de ligne optionnel suivi par un ou plusieurs *mots*. Un mot commence par une lettre suivie d'un nombre (ou quelque chose qui permet d'évaluer un nombre). Un mot peut, soit donner une commande, soit fournir un argument à une commande. Par exemple, G1 X3 est une ligne de code valide avec deux mots. G1 est une commande qui signifie déplace toi en ligne droite à la vitesse programmée et X3 fournit la valeur d'argument (la valeur de X doit être 3 à la fin du mouvement). La plupart des commandes G-code commencent avec une lettre G ou M. Les termes pour ces commandes sont G-codes ou M-codes.

Le langage G-code n'a pas d'indicateur de début et de fin de programme. L'interpréteur cependant traite les fichiers. Un programme simple peut être en un seul fichier, mais il peut aussi être partagé sur plusieurs fichiers. Un fichier peut être délimité par le signe pour-cent de la manière suivante. La première ligne non vide d'un fichier peut contenir un signe % seul, éventuellement encadré d'espaces blancs, ensuite, à la fin du fichier on doit trouver une ligne similaire. Délimiter un fichier avec des % est facultatif si le fichier comporte un *M2* ou un *M30*, mais est requis sinon. Une erreur sera signalée si un fichier a une ligne pour-cent au début, mais pas à la fin. Le contenu utile d'un fichier délimité par pour -cent s'arrête après la seconde ligne pour-cent. Tout le reste est ignoré.

Le langage G-code prévoit les deux commandes (*M2* ou *M30*) pour finir un programme. Le programme peut se terminer avant la fin du fichier. Les lignes placées après la fin d'un programme ne seront pas exécutées. L'interpréteur ne les lit pas.

II.5.2.2. Format d'une ligne

Une ligne de G-code typique est construite de la façon suivante, dans l'ordre avec la restriction à un maximum de 256 caractères sur la même ligne.

- 1.Un caractère optionnel d'effacement de bloc, qui est la barre oblique.
- 2.Un numéro de ligne optionnel.
- 3.Un nombre quelconque de mots, valeurs de paramètres et commentaires.
- 4.Un caractère de fin de ligne (retour chariot ou saut de ligne ou les deux)

II.5.3. L'éditeur de texte ou image compilateur de langage CNC

L'éditeur de texte est une interface graphique sur l'ordinateur où l'opérateur peut traiter le G-code à exécuter s'appelé « Inkscape ». Ensuite, ce texte doit être compilé, s'il y a des erreurs ; l'opérateur est prévu afin de les corriger. Une fois compilé, le code est prêt à envoyer à la carte Arduino Uno ou à simuler sur l'ordinateur.

II.5.3.1. Définition

Inkscape est un logiciel libre de dessin vectoriel, multiplateforme : il s'utilise aussi bien sur Windows, Mac OS X et Gnu/Linux. Inkscape est intuitif et ses utilisateurs s'amuse rapidement dès la première prise en main. Logiciel vectoriel performant, entièrement dédié à la création et aux modifications des chemins, il ravira toutes les personnes ayant un besoin d'illustration sur divers supports et de taille variable. En effet, Inkscape s'utilise aussi bien pour créer des flyers, logos, affiches, cartes de visites, illustrations, interfaces de logiciels, d'applications ou de site web. Il est également utile au dessin technique ou à la création de fonte (en SVG), même si cet usage n'est pas son objectif premier.

Son format d'enregistrement est le SVG, format préconisé par le W3C et s'allie aisément avec le HTML5, CSS3 et JavaScript. Ce qui favorisera, peut-être, l'augmentation d'illustrations au format SVG dans les pages webs.

II.5.3.2. Interface

Inkscape se présente en une seule fenêtre. La part belle est laissée au canevas, l'espace blanc où se déroule la création de l'image.

1. Le premier menu déroulant horizontal contient les commandes essentielles du logiciel communes à de nombreux autres (copier, enregistrer...) et spécifiques à Inkscape (Chemin, Texte., Extensions...).
2. La deuxième barre horizontale affichent les options de l'outil sélectionné. Ce menu est donc contextuel : il s'adapte aux spécificités de l'outil.
3. La barre verticale située à gauche présente tous les outils principaux du logiciel.
4. Le grand espace blanc est le canevas, là où l'image se créer. Un cadre noir représente, par défaut, une page au format A4.

5. La barre à droite du canevas contient des boutons vers des commandes utiles, autrement accessible via les menus déroulants du premier menu horizontal.
6. La dernière barre de droite concerne le magnétisme.
7. En haut et à gauche du canevas, des règles, pour rendre compte de l'échelle de l'image et pour placer des guides précisément.
8. À droite et en bas du canevas, des ascenseurs sont visibles pour se déplacer sur le canevas.
9. L'avant dernière barre horizontale située au bas de la fenêtre présente des raccourcis vers la palette des couleurs.
10. La dernière barre horizontale contient des informations diverses, comme les couleurs de l'objet sélectionné, les calques, une barre d'état contenant des informations sur les opérations en cours, les coordonnées du pointeur, le zoom. D'autres commandes sont disponibles au travers de dialogues qui s'attachent à droite du canevas pour ne pas gêner la visibilité.

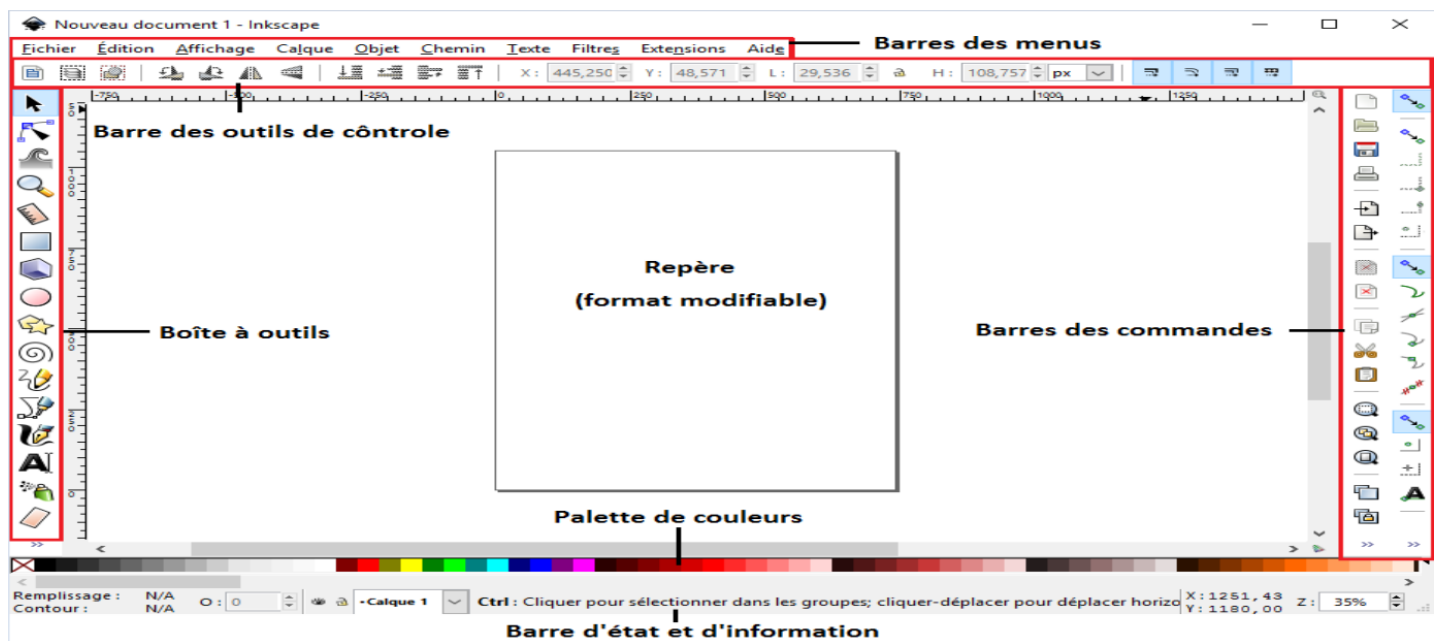


Figure : L'interface d'Inkscape

"G-code tools" est une extension d'Inkscape permettant de générer du G-code à partir des chemins d'une image vectorielle.

II.5.4. Logiciel de pilotage de la machine CNC

La solution la plus fonctionnelle que j'ai trouvée est une évolution de **Universal Gcode Sender** (développé initialement par l'auteur de GRBL), qui est cours de développement par Winder.

Universal Gcode sender (UGS) est un programme Java utilisé pour envoyer un fichier G-code vers le GRBL d'une machine à commande numérique.

Avec l'expéditeur G-Code Universel il est possible de contrôler le système avec des commandes G-code, comme le déplacement des moteurs pas à pas à la position initiale.

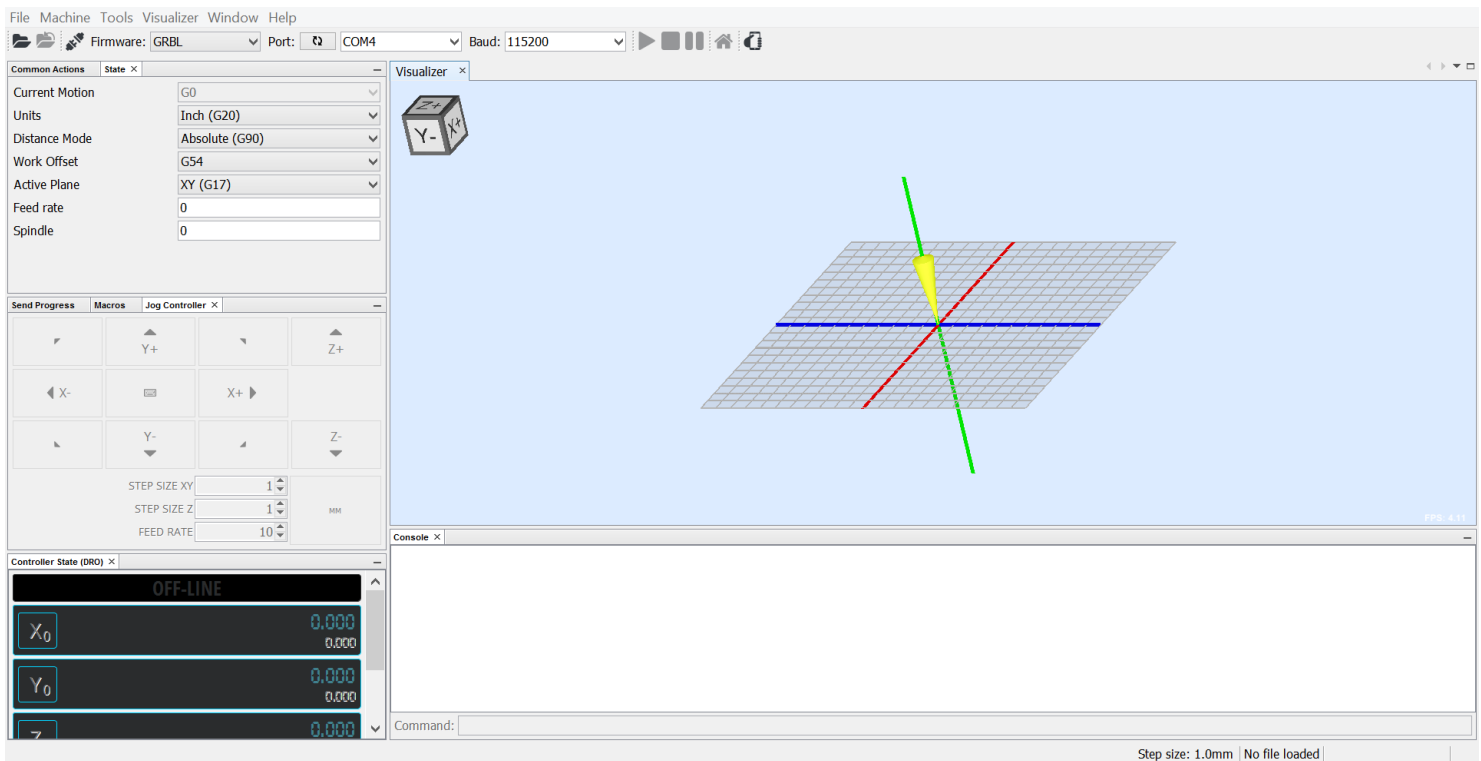


Figure : de Universal G-code Sender

Il y'a deux mode utilise dans ce programme soit mode manuel ou mode automatique pour commande le machine CNC,

II.5.4.1. Mode manuel

En mode manuel, l'opérateur peut commander les moteurs directement par les boutons (X+,X-,Y+,Y-,Z+,Z-,...) dans la programme de transfert G-code , si En cliquant sur ses boutons nous générons un instruction de code G qui envoyé à la carte de commande pour commander le moteur connecté avec ce bouton .

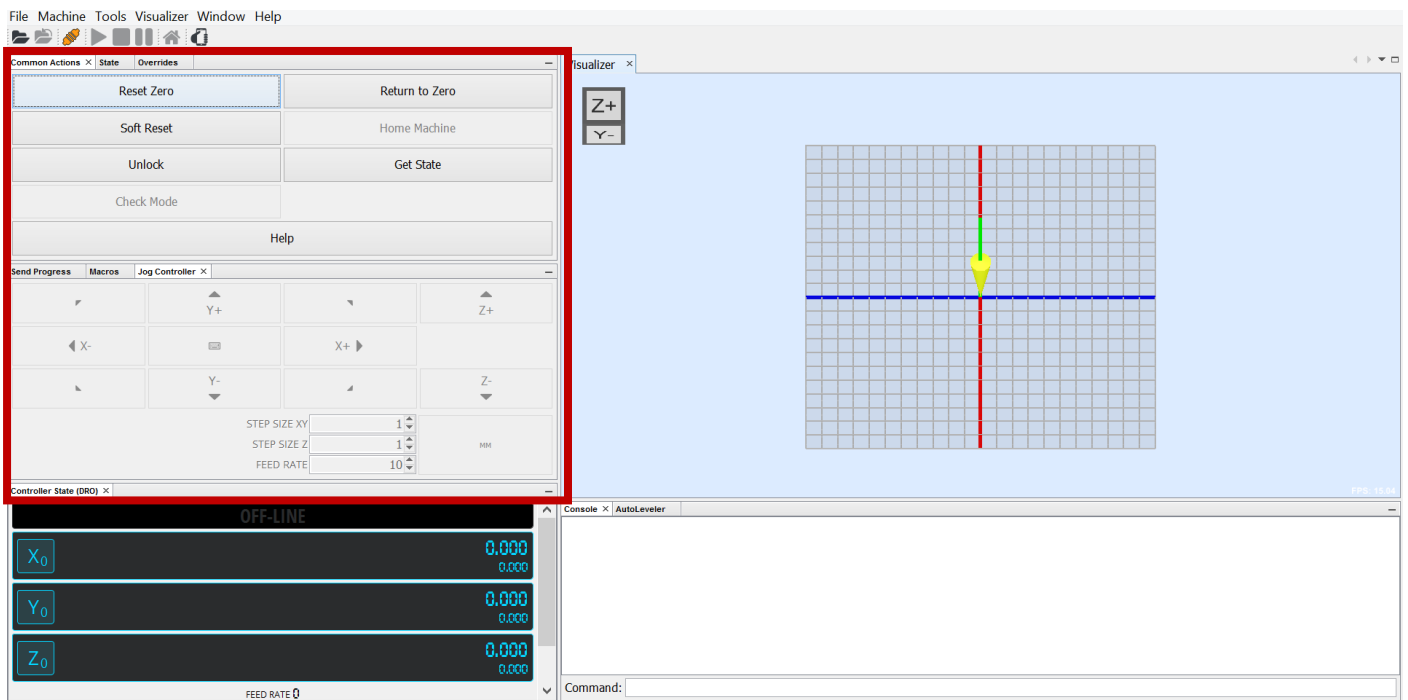


Figure : l'interface de “G-code Sender ”on mode manuel

II.5.4.2. Mode automatique

En mode automatique, la carte Arduino Uno reçoit les instructions en G-code et les exécute l'une après l'autre. Il y a plusieurs étapes pour achever cette tâche. Commenant par la réception de l'instruction jusqu'à la commande des moteurs et de la tête. Si l'opérateur clique sur le bouton « send » après le chargement de fichier G-code (soit texte ou image) dans le programme de transfert, l'Arduino fonctionne en mode automatique, et devient prêt pour recevoir les données.

L'Arduino fait l'enregistrement dans la mémoire RAM, Puis fait la compilation pour savoir l'opcode (code de l'opération) et évidemment ses paramètres s'ils existent. Chaque opcode de G-

code a une fonction spécifique, s'il s'agit d'un paramétrage, l'instruction affecte une variable en EEPROM tels que l'origine programme, origine pièce, vitesse de déplacement.... Enfin, le Arduino envoie le mot '»' pour indiquer au PC qu'il est prêt à recevoir l'instruction suivante. L'appui sur le bouton 'cancel' termine le dialogue.

Pour l'utilisation du mode automatique, nous Chargeons notre fichier G-code : dans l'onglet "file mode" nous sélectionnons le fichier G-code précédemment créé. En faisant "visualize" nous pourrions vérifier notre tracé et le chemin emprunté par l'outil.

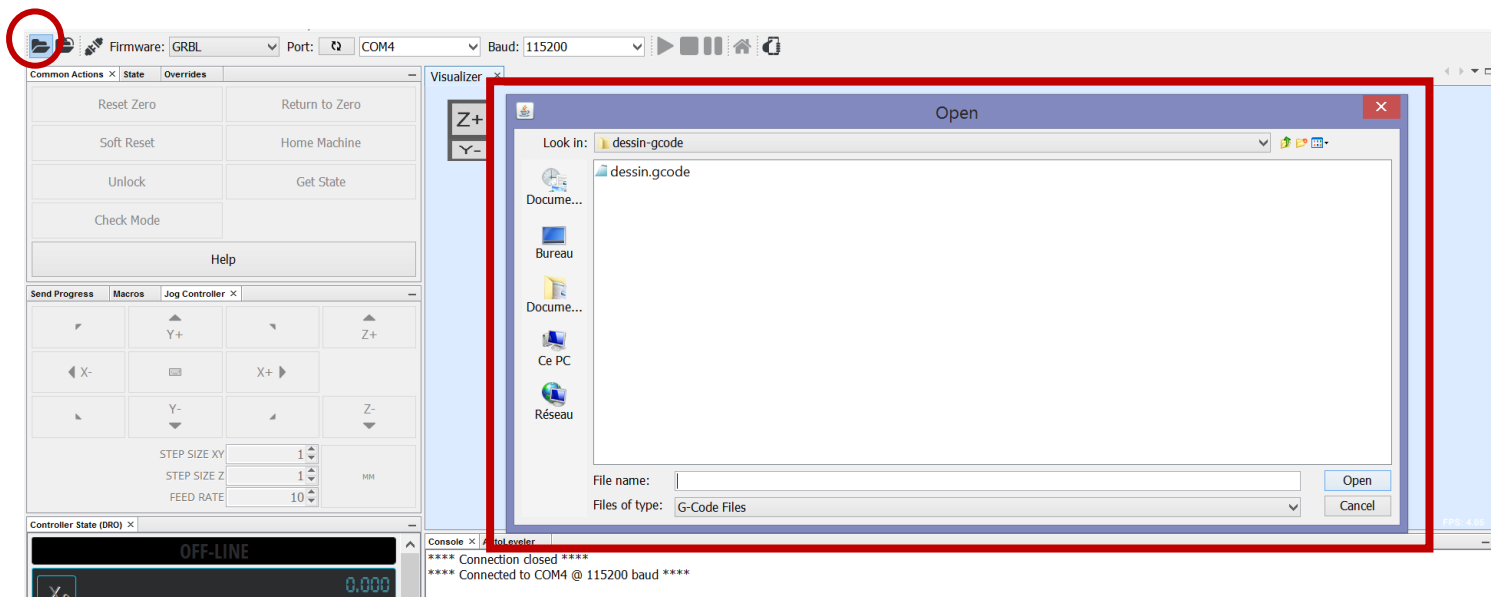


Figure : l'interface de "G-code Sender" choisir le fichier G-code

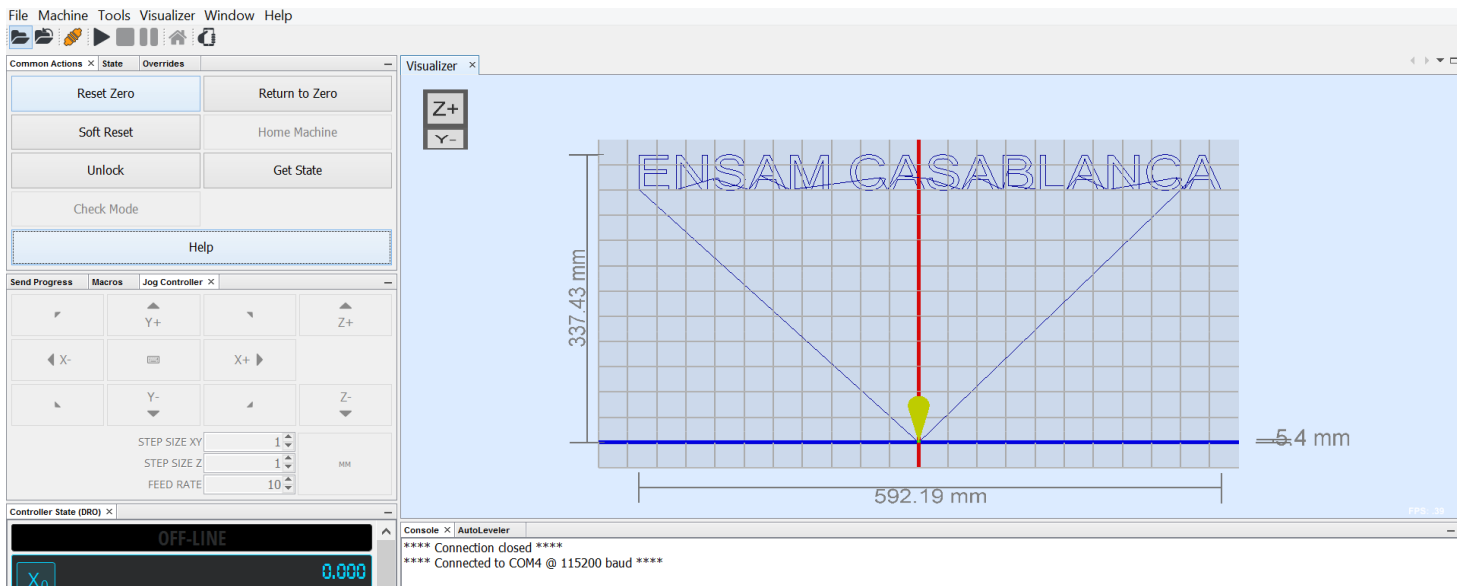


Figure : l'interface de "G-code Sender" on mode automatique

Faire le zéro (le "home"): dans l'onglet "machine control" déplacez l'outil avec les boutons X, Y et Z. Nous pouvons modifier le pas de déplacement en modifiant le "step size". Une fois l'outil positionner là où vous souhaitez établir le zéro (en contact avec notre support) cliquez sur "reset zéro". Pour lancer le travail : cliquez sur "Send".

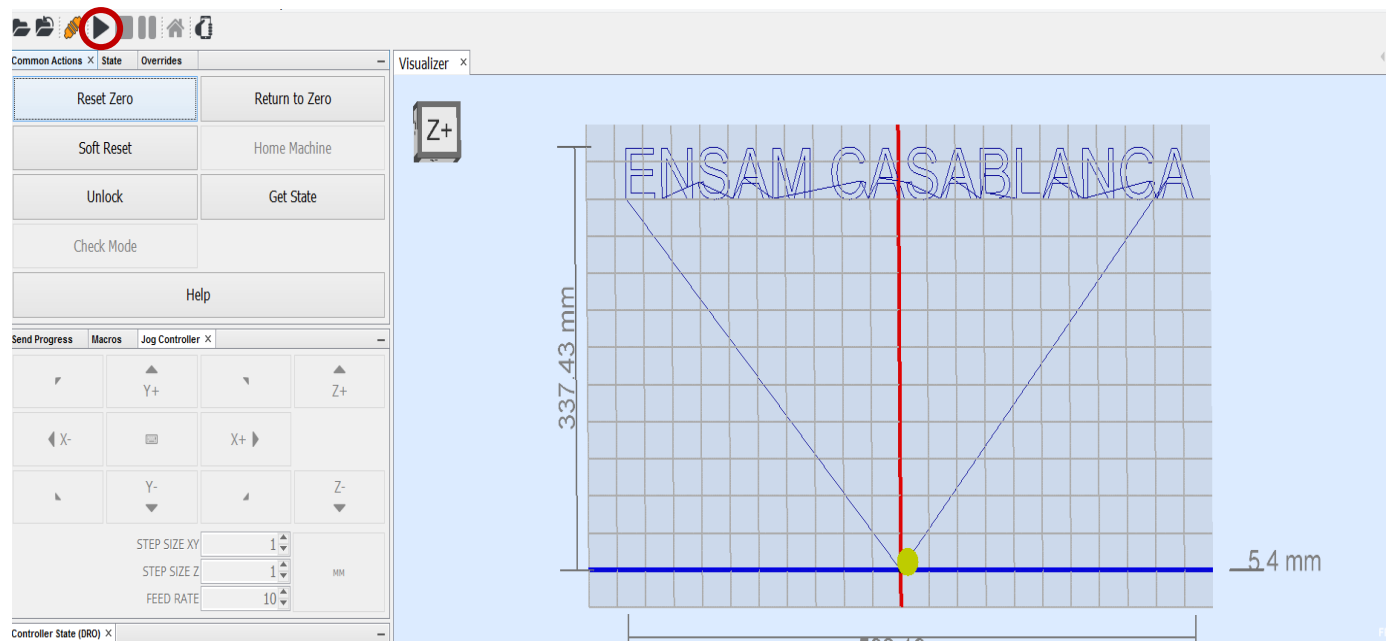


Figure : envoyé le fichier G code

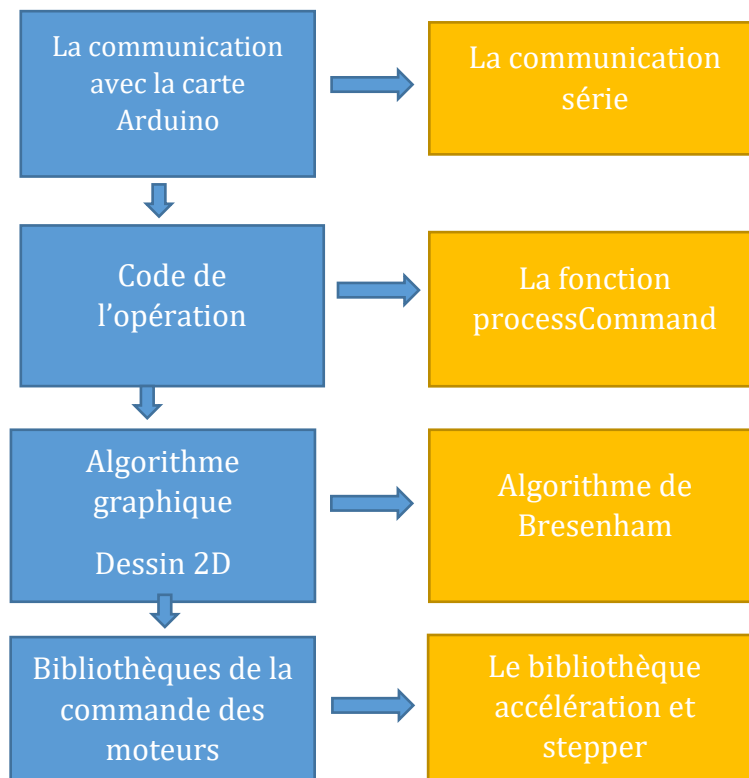
II.5.5. Langage de programmation la machine CNC

Un langage de programmation est un langage permettant à un être humain d'écrire un ensemble d'instructions (code source) qui seront directement converties en langage machine grâce à un compilateur (c'est la compilation). L'exécution d'un programme Arduino s'effectue de manière séquentielle, c'est-à-dire que les instructions sont exécutées les unes à la suite des autres. Voyons plus en détail la structure d'un programme écrit en Arduino. Le programme utilise l'équipe de Grbl est une le programme de commande dans la carte Arduino.

II.5.5.1 Programme Grbl

Grbl est un micrologiciel libre développé sur Arduino pour contrôler des graveuses CNC(Computer Numerical Control), i.e. des fraiseuses munis d'une tête mobile contrôlée en X, Y et Z par un ordinateur. Grbl interprète du G-code (cf. plus bas) et déplace en conséquence un outil sur 3 axes (X, Y et Z). Il comprend de multiples optimisations sur l'usage et le déplacement des moteurs afin de gérer correctement les accélérations, les trajectoires...

Voici l'organigramme qui assure ce micrologiciel :



II.5.5.1.1. La communication avec la carte Arduino

La communication série est indispensable pour dialoguer avec Arduino puisque c'est le mode de transmission utilisée pour communiquer avec la carte.

Les protocoles de communication série le plus connus sont : Le protocole USB.

- Serial.begin(speed) (configuration de la vitesse de communication Série).
- Serial.available() (donne combien de caractères disponibles dans la zone tampon Série) .
- Serial.read() (lit les données Série) .
- Serial.print(data) (envoie des données Série)
- Serial.println(data) (envoie des données Série suivies de caractères spécifiques).

L'avantage de la communication série est qu'elle nécessite moins de lignes, donc moins de broches, donc moins de composants. Son coût est donc plus faible.

II.5.5.1.2. Code de l'opération

Cette partie est le cœur du programme, qui définit les actions à effectuer à partir le traitement de code-G, qui contient plusieurs fonctions. L'Arduino fait l'enregistrement dans la mémoire RAM, Après la compilation nous obtenons l'opcode (code de l'opération) et évidemment ses paramètres s'ils existent. Chaque opcode du code-G a une fonction spécifique.

La fonction principale qui traduit le G code s'appelle : « processCommand() » qui contient l'opcode se la forme suivent:

```

void processCommand() {

int cmd = parsenumber('G',-1);
switch(cmd) {
case 0:
case 1:
{ // linefeedrate(parsenumber('F',fr);
  line( parsenumber('X',(mode_abs?px:0))+ (mode_abs?0:px),
    parsenumber('Y',(mode_abs?py:0)) + (mode_abs?0:py),
    parsenumber('Z',(mode_abs?pz:0)) +(mode_abs?0:pz),
    break;
}
case 2:
case 4:
  pause(parsenumber('P',0)*1000); break; // dwell
case 90: mode_abs=1; break; // absolute mode
case 91: mode_abs=0; break; // relative mode
case 92: // set logical position
    position(
        parsenumber('X',0),
        parsenumber('Y',0),
        parsenumber('Z',0));

  break;
default: break;
}
cmd = parsenumber('M',1);
switch(cmd) {

```



```
case 17: motor_enable(); break;
case 18: motor_disable(); break;
case 100: help(); break;
case 114: where(); break;
default: break;
}
}
```

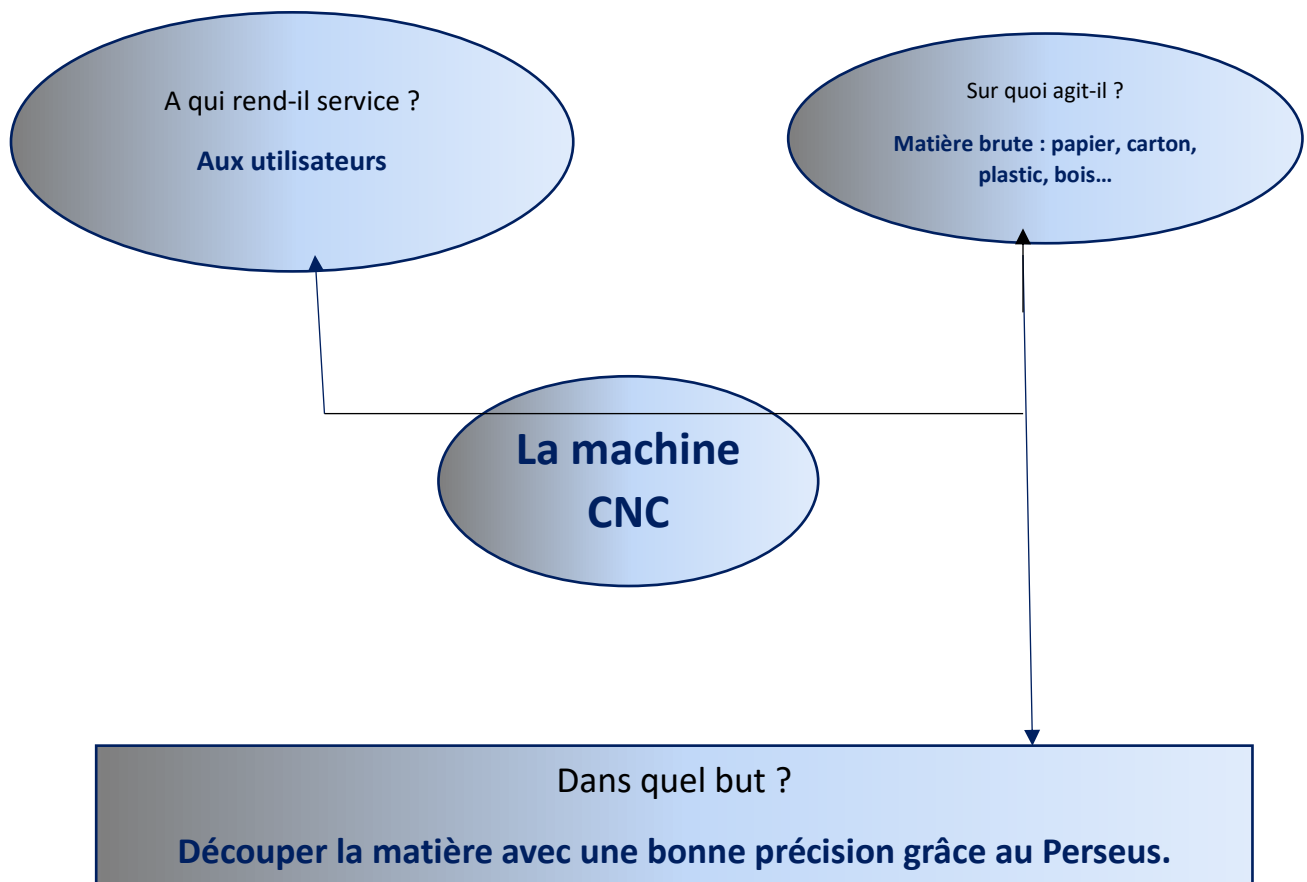
II.5.6. Conclusion

Ce chapitre est une étude générale du logiciel de commande graphique et le programme de commande de la machine CNC. L'ensemble des outils que nous avons mis au point ont pour la plupart été développés grâce au logiciel Inkscape, pour son utilisation simple et les nombreuses possibilités qu'il offre. La première étape de cette partie consiste en la lecture d'image pour la tracer et la génération fichier G code. Puis nous avons choisi logiciel G-code Sender pour transfert fichier G-code vers la carte Arduino Uno.

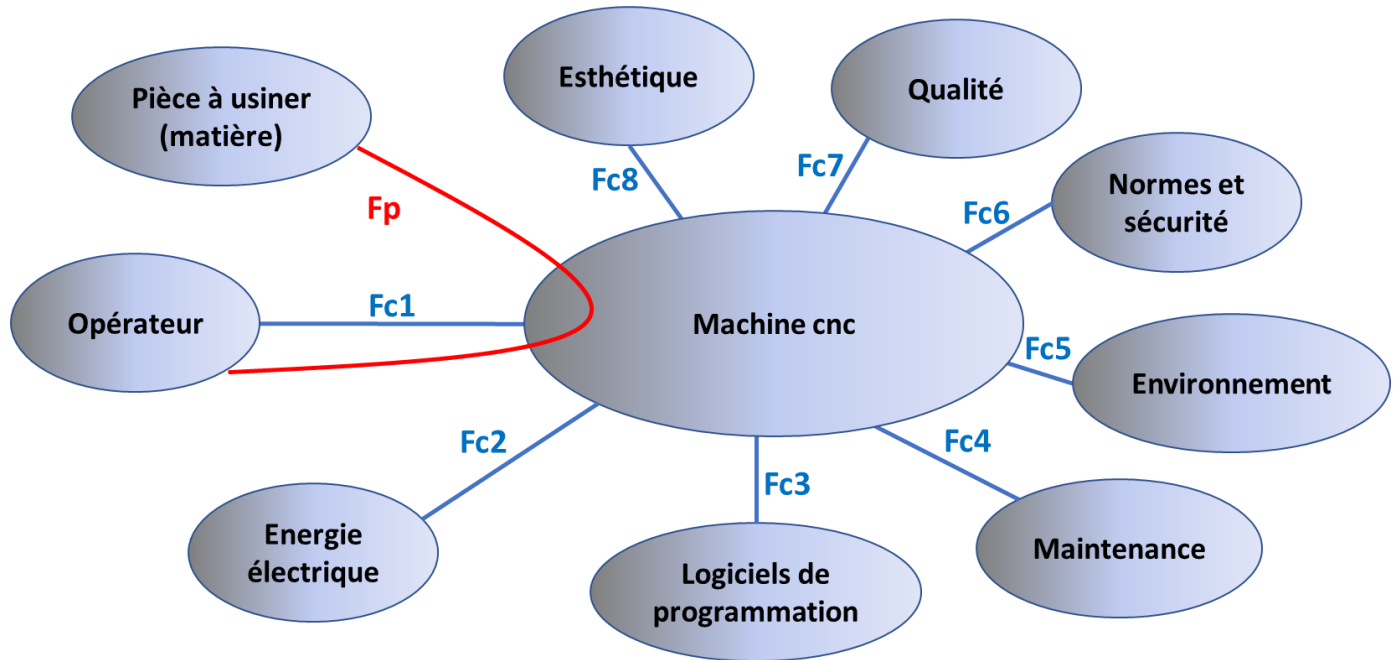
Chapitre 3 : Analyse fonctionnelle, Étude AMDEC et l'Analyse de défaillance (ADD)

III.1. Analyse fonctionnelle

III.1. 1. Énoncé du besoin (*Bête à cornes*)

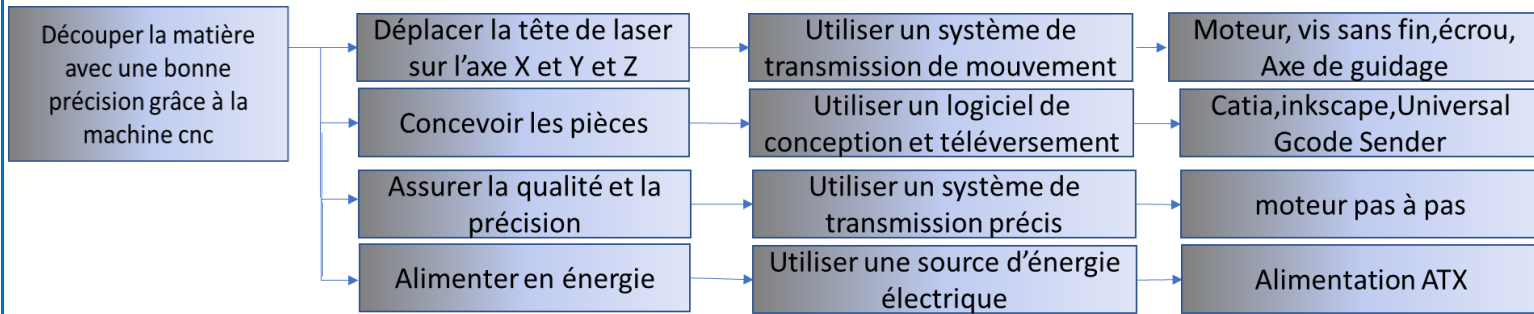


III.1. 2. Analyse du besoin : Diagramme Pieuvre



FP	Découper la matière avec une bonne précision grâce à la découpeuse laser.
Fc1	Faciliter l'utilisation de la machine.
Fc2	Utiliser une source d'énergie électrique.
Fc3	Utiliser des logiciels commerciaux pour faciliter la programmation.
Fc4	Faciliter la maintenance.
Fc5	Respecter l'environnement du travail et ne pas faire du bruit.
Fc6	Respecter les normes de sécurité.
Fc7	Assurer la qualité du produit.

III.1. 3. Analyse technique : Diagramme Fast



III.2. Étude AMDEC

III.2.1. Définition AFNOR :

Selon la définition AFNOR, AMDEC (l'Analyse des Modes de Défaillance, de leurs Effets et de leur Criticité) est une méthode d'analyse de la fiabilité qui permet de recenser les défaillances dont les conséquences affectent le fonctionnement du système dans le cadre d'une application donnée (de Sûreté de Fonctionnement (Sdf) et de gestion de la qualité).

L'Analyse des Modes de Défaillance, de leurs Effets et de leur Criticité (AMDEC) est avant tout une méthode d'analyse de systèmes s'appuyant sur un raisonnement inductif (causes conséquences), pour l'étude organisée des causes, des effets des défaillances et de leur criticité.

III.2.2. Types de l'AMDEC

Il existe trois principaux types d'AMDEC :

- L'AMDEC produit : permet de vérifier la viabilité d'un produit développé par rapport aux exigences du client ou de l'application.
- L'AMDEC processus : permet d'identifier les risques potentiels liés à un procédé de fabrication conduisant à des produits non conformes ou des pertes de cadence.
- L'AMDEC moyen de production : permet d'anticiper les risques liés au non fonctionnement ou au fonctionnement anormal d'un équipement, d'une machine.

III.2.3. Buts de l'AMDEC

L'AMDEC est une technique qui vise à :

- Évaluer et garantir la sûreté de fonctionnement (sécurité, fiabilité, maintenabilité et disponibilité).
- Obtenir au meilleur coût le rendement global maximum des équipements de production.
- Définir les actions à entreprendre pour éliminer ces défaillances, réduire leurs effets et pour en empêcher ou en détecter les causes.
- Prioriser les interventions d'amélioration continue.
- Réduire les risques les plus grands.
- Élaborer des plans d'actions et allouer les ressources de façon rationnelle.
- Déterminer les points faibles d'un système et apporter des remèdes.
- Prévenir les pannes.
- Réduire le temps d'indisponibilité après défaillance.

III.2.4. L'analyse des défaillances

Il s'agit d'identifier les schémas du type :

1-Le mode de défaillance

Il concerne la fonction et exprime de quelle manière cette fonction ne fait plus ce qu'elle est sensée faire. L'analyse fonctionnelle recense les fonctions, l'AMDEC envisage pour chacune d'entre-elles sa façon (ou ses façons car il peut y en avoir plusieurs) de ne plus se comporter correctement.

2-La cause

C'est l'anomalie qui conduit au mode de défaillance.

La défaillance est un écart par rapport à la norme de fonctionnement.

Les causes trouvent leurs sources dans cinq grandes familles. On en fait l'inventaire dans des diagrammes dits "diagrammes de causes à effets"

Chaque famille peut à son tour être décomposée en sous-familles.

Un mode de défaillance peut résulter de la combinaison de plusieurs causes.

Une cause peut être à l'origine de plusieurs modes de défaillances.

3-L'effet

L'effet concrétise la conséquence du mode de défaillance.

Il dépend du point de vue AMDEC que l'on adopte :

- effets sur la qualité du produit (AMDEC procédé),
- effets sur la productivité (AMDEC machine),
- effets sur la sécurité (AMDEC sécurité).

Un effet peut lui-même devenir la cause d'un autre mode de défaillance.

III.2.5. EVALUATION

L'évaluation se fait selon 3 critères principaux :

La gravité, la fréquence, la non-détection.

Ces critères ne sont pas limitatifs, le groupe de travail peut en définir d'autres plus judicieux par rapport au problème traité.

Chaque critère est évalué dans une plage de notes.

Cette plage est déterminée par le groupe de travail.

a-La gravité :

Elle exprime l'importance de l'effet sur la qualité du produit (AMDEC procédé) ou sur la productivité (AMDEC machine) ou sur la sécurité (AMDEC sécurité). Le groupe doit décider de la manière de mesurer l'effet.

b-La fréquence :

On estime la période à laquelle la défaillance est susceptible de se reproduire.

c-La non-détection :

Elle exprime l'efficacité du système permettant de détecter le problème.

d-La criticité :

Lorsque les 3 critères ont été évalués dans une ligne de la synthèse AMDEC, on fait le produit des 3 notes obtenues pour calculer la criticité.

$$C = G * F * N$$

Criticité Gravité Fréquence non-détection

Le groupe de travail doit alors décider d'un seuil de criticité.

Au-delà de ce seuil, l'effet de la défaillance n'est pas supportable. Une action est nécessaire.

III.2.6. les actions

La finalité de l'analyse AMDEC, après la mise en évidence des défaillances critiques, est de définir des actions de nature à traiter le problème identifié.

Les actions sont de 3 types :

- Actions préventives : on agit pour prévenir la défaillance avant qu'elle ne se produise, pour l'empêcher de se produire. Ces actions sont planifiées. La période d'application d'une action résulte de l'évaluation de la fréquence.
- Actions correctives : lorsque le problème n'est pas considéré comme critique, on agit au moment où il se présente. L'action doit alors être la plus courte possible pour une remise aux normes rapide.
- Actions amélioratives : il s'agit en général de modifications de procédé ou de modifications Technologiques du moyen de production destinées faire disparaître totalement le problème. Le coût de ce type d'action n'est pas négligeable et on le traite comme un investissement.

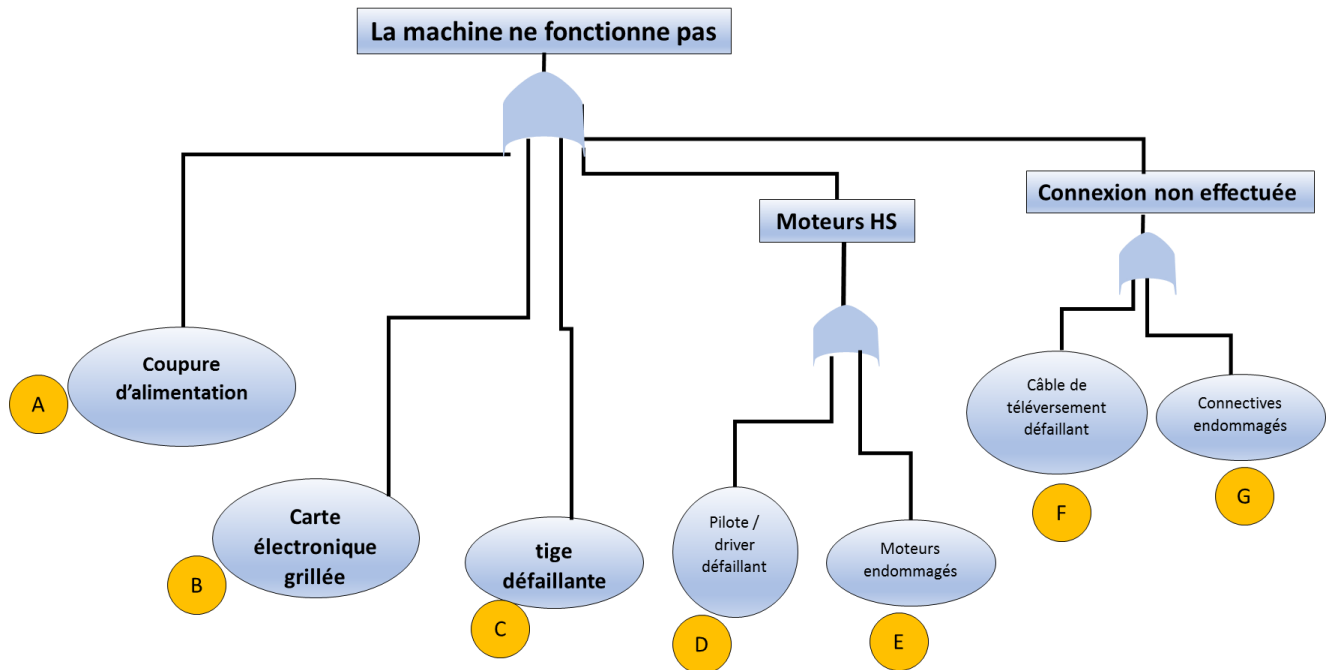
III.2.7. Application de l'AMDEC sur la machine CNC

Composant	Fonction	Mode de défaillance	Causes	Effets	Modes de détection	F	G	N	C	Actions correctives
Carte d'alimentation	Assure l'alimentation du moteur pas à pas	Grillage des résistances	-Surtension -Problème de capacité	Arrêt complet de la machine	Inspection visuelle	3	4	3	36	-Changement -Effectuer un contrôle périodique
Driver A4988	Piloter les moteurs	-Pas de rotation des roues	-Hors service	-Arrêt du robot	Visuelle	3	1	1	3	-changement
		-Décalage de vitesse	-le mal fonctionnement	-Déséquilibre du robot	Visuelle	1	1	3	3	
Arduino UNO	La commande de la machine	Fonctionnement irrégulier	-Rupture d'un fil de connexion	-Faire des actions inattendues	Visuelle					-changement
		-Pas de rotation des moteurs	-Coupure de l'alimentation -Grillage	-Arrêt du robot	Visuelle					
Vis sans fin / écrou	Transforme les Mouvements rotatifs à un mouvement de translation	-Craquelée/cassée	-Défaut de fabrication -Vieillessement	- Diminuer les performances de transmission	Bruit Vibration	1	2	2	4	
Accouplement	Assure la liaison entre l'arbre du moteur et l'arbre broche	Cassure	-vieillessement	Mauvaise liaison	Visuelle	2	1	1	2	

Moteur pas à pas	Assure le mouvement du miroir de renvoi grâce à des courroies	Court-Circuit	-Surcharge du moteur -Problème de carte d'alimentation défectueuse -Usure ou cassure des roulements -vieillessement	Arrêt de rotation du moteur.	Diminution du couple	1	4	2	8	Contrôle périodique de l'état générale du moteur.
		Aimant craqué/cassé	Usure normal	Perte de performance	Vibrations	1	4	3	12	
		Surchauffe	-Déphasage -Surcharge du moteur -Contaminants dans l'air ambiant	Perte de performance	Infrarouge	2	2	2	8	
		Rotation inverse	Erreur de câblage	Arrêt de la machine	Inspection visuelle	1	3	3	9	Consigne opérateur de maintenance

III.3. Analyse de défaillance (ADD)

III.3.1. Arbre de défaillance



III.3.2. Traitement qualitatif

- Équation Booléenne de l'arbre :

$$EI = A+B+C+D+E+F+G$$

III.3.3. Traitement quantitatif

- Quantification des événements de base

On quantifie les événements de base par l'affectation à chaque événement la probabilité d'apparition estimé pendant une durée de 100 heures de travail.

Événement	A	B	C	D	E	F	G
Probabilité	1.10^{-4}	1.10^{-1}	1.10^{-4}	1.10^{-3}	1.10^{-4}	1.10^{-5}	1.10^{-2}

- Calcul de la probabilité de l'élément redouté :

$$P(\text{La machine ne fonctionne pas}) = 0.11$$

- Pourcentage de contribution de chaque événement de base

Événement	A	B	C	D	E	F	G
Probabilité	1.10^{-4}	1.10^{-1}	1.10^{-4}	1.10^{-3}	1.10^{-4}	1.10^{-5}	1.10^{-2}
Pourcentage (%)	9.10^{-4}	0.9	9.10^{-4}	9.10^{-3}	9.10^{-4}	9.10^{-5}	9.10^{-2}

Après on a déterminé les pourcentages de contribution de chaque événement de base, On constate que l'événement B a une très grande chance d'apparition avec un pourcentage de 90%. Par conséquent, on doit centrer les actions de correction autour de ce dernier.

Conclusion Générale

En conclusion, je tiens à préciser que malgré toutes les contraintes qui m'ont empêché de trouver un stage, réaliser un projet et plus précisément la machine CNC fut pour moi une chance inouïe et un bénéfice. En effet, j'ai pu développer plusieurs de mes compétences soit dans le domaine de la mécanique comme par exemple au niveau de la conception mécanique et des mesures en étudiant les moteurs pas à pas ou dans le domaine de l'électronique au niveau de tout ce qui concerne le contrôle numérique et la zone de programmation à travers laquelle nous contrôlons la machine. De plus, je sais à présent ce que ça vaut de concevoir, d'améliorer et de trouver toutes les solutions aux problèmes comme un bon ingénieur le fera.

Références

1. <https://eduscol.education.fr/sti/sites/eduscol.education.fr/sti/files/ressources/pedagogiques/7572/7572-modelisation-dun-moteur-pas-pas-v2.pdf>
2. [file:///C:/Users/anas/Downloads/maxon-Entra%C3%AEnement%20vis_%C3%A9crou_11_FR_246%20\(1\).pdf](file:///C:/Users/anas/Downloads/maxon-Entra%C3%AEnement%20vis_%C3%A9crou_11_FR_246%20(1).pdf)
3. http://www.mon-fablab.fr/labtech/cartes_elec/cnc_shield
4. <http://si.lycee-desfontaines.eu/sequences-ts/we2/res/ressource-vis-a-bille.pdf>
5. <https://www.prudhomme-trans.com/wp-content/uploads/2013/05/198.pdf>
6. <https://shop.hpceurope.com/fr/produit.asp?prid=1243>
7. <file:///C:/Users/anas/Downloads/lfm102.pdf>
8. <https://www.motedis.fr/shop/Dynamic-unites-lineaire/Broche-filetee-trapezoidale-accessoires/Broche-filetee-trapezoidale/Acier-Trapezoidale-tige-filetee/Trap%E9zo%EFdale-tige-filet%E9e-RPTS-droite-TR-8x15-L%3D500mm::999993952.html>
9. <https://french.alibaba.com/product-detail/lm10uu-thk-linear-motion-bearing-351350092.html>
10. <https://www.amazon.com/Spindle-Cooled-Milling-Converter-Engraving/dp/B01LNBOCDA>
11. https://www.motedis.fr/shop/Dynamic-unites-lineaire/Arbres-de-precision/Precision-arbre-acier::99999213_99999295_99999783.html
12. https://ar.banggood.com/Nema17-59Ncm-2A-1_84-lead-48mm-Stepper-Motor-For-3D-Printer-CNC-p-1252415.html?p=PW041611183930201706&custlinkid=146881&cur_warehouse=CN
13. <http://cnc25.free.fr/documentscnc25.htm>