

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

UNIVERSITE AKLI MOHAND OULHADJ BOUIRA

FACULTE DES SCIENCES ET DES SCIENCES APPLIQUEES

DEPARTEMENT GENIE ELECTRIQUE



Projet de fin d'études Master LMD

Filière : Electrotechnique

Spécialité :

Electromécanique et systèmes de commande

Thème :

*Réalisation d'une commande
pour une machine (SNC, 3D)*

Encadré par:

Réalisé par:

Mr CHATBI Hamid

Mme S.BENSMAIL

Mr KARI Djamal eddine

Promotion: 2017

Remerciements

Tout d'abord, Je tiens à remercier et glorifier الله le tout puissant et miséricordieux d'avoir guidé mes pas avec clairvoyance pour accomplir ce travail et de le mener à terme.

Je tiens à exprimer ma plus profonde reconnaissance à :

- ✚ Mes parents qui m'ont toujours entouré et motivé à sans cesse devenir meilleur ;
- ✚ Mes frères et sœurs qui m'ont assisté dans ces moments et m'ont servi d'exemple ;
- ✚ Ma promotrice Mme S.BENSMAIL pour son aide et sa précieuse attention ;
- ✚ Mon encadreur en stage pratique Mr KARI Djamal eddine ;
- ✚ Mes collègues qui n'ont cessé de m'encourager ;

Table des matières

Remerciment	I
Table des matières	II
Tables des figures	VI
Table des tableaux	IX
Introduction générale.....	1
CHAPITRE I : Généralités sur la machine-outil à commande numérique.	
I.1. Introduction.....	2
I.2. Historique de la Commande Numérique.....	2
I.3. Justification de la commande numérique	4
I.3.1. Automaticité	4
I.3.2. Flexibilité	4
I.3.3. Sécurité.....	5
I.3.4. Nécessités économiques et techniques	5
I.4. Comparaison entre MO conventionnelles et MOCN :	6
I.5. Les différentes techniques d'usinage avec les MOCN	7
I.5.1. Le perçage	8
I.5.2. Le fraisage	8
I.5.3. Le tournage.....	8
I.5.4. Le filetage.....	8
I.5.5. L'alésage	8
I.6. Domaine d'utilisation technique et économique des MOCN	9
I.6.1. Domaine d'utilisation technique	9
I.6.2. Domaine d'utilisation économique	9
I.7. Architecture et organisation générale d'une MOCN	11
I.8 Avantages et les inconvénients	13

I.8.1 les avantages.....	13
I.8.2. les inconvénients	13
I.9. Conclusion	14

CHAPITRE II : Conception de la machine-outil.

II.1. Introduction	15
II.2. Conception assistée par ordinateur (CAO).....	15
II.2.1. Description du logiciel « SOLIDWORKS ».....	15
II.3. Conception de la machine.....	16
II.3.1. Moteur pas à pas	16
II.3.2. Palier	17
II.3.3. Accouplement	17
II.3.4. Vis à bille	18
II.3.5. Glissière	19
II.3.6. Rail.....	19
II.3.7. Axe d'une CNC.....	20
II.3.7.1. Schéma cinématique d'un axe.....	20
II.3.8. La broche :	21
II.3.9. Le bâti	22
II.4. Conclusion	22

CHAPITRE III : Commande de la machine CNC.

III.1. Introduction	23
III.2. Présentation de la commande	23
III.3. G-Code	25
III.4. Interpréteurs G-Code et contrôleurs de CNC.....	26
III.4.1. Organisation du programme	27
III.4.2. Interpolation.....	30
III.4.3. Algorithme de tracé de segment de Bresenham	31

III.4.3.1. Principe	31
III.4.3.2. Détermination des ordonnées	31
III.5. LES MOTEURS PAS-À-PAS	33
III.5.1. Définition.....	33
III.5.2. Les différents types des moteurs pas à pas.....	34
III.5.2.1. Le moteur bipolaire.....	34
III.5.2.2. Le moteur unipolaire.....	35
III.5.2.3. Moteur à réluctance variable	36
III.5.2.4. Moteur à aimant permanent	37
III.5.2.5. Moteur hybride	37
III.5.3. MODE DE COMMANDE DES MOTEURS PAS À PAS	38
III.5.3.1. Alimentation d'une seule phase en pas entier (Mode1).....	38
III.5.3.3. Alimentation d'une ou deux phases en demi-pas (Mode3)	39
III.5.3.4. Commande optimisé en demi-pas (Mode4).....	39
III.5.3.5. Commande avancée en micro-pas (Mode5)	40
III.6. La carte de commande ARDUINO	40
III.6.1. ARDUINO MEGA 2560	40
III.6.2. Logiciel ARDUINO	42
III.6.3. Programmation sous ARDUINO	43
III.7. Circuit d'interface.....	43
III.8. Conclusion.....	45
CHAPITRE IV : Réalisation expérimentale de la machine CNC.	
IV.1. Introduction	46
IV.2. Eléments et caractéristiques de la partie commande	46
IV.2.1. Circuit d'interface.....	46
IV.2.1.1. Réalisation virtuelle sous « PROTEUS ».....	46
IV.2.1.2. Réalisation pratique du circuit :	48

IV.2.1.2.1. Impression du circuit sur la platine cuivrée.....	48
IV.2.1.2.2. Gravure du circuit.....	48
IV.2.1.2.3. Perçage et soudage	49
IV.2.2. Contrôleur du moteur pas à pas (driver) :.....	50
IV.2.3. L'armoire de commande.....	51
IV.3. Eléments et caractéristiques de la partie opérative de la machine	52
IV.3.1. Le bâti.....	52
IV.3.2. Caractéristiques de la broche.....	53
IV.3.3. Les axes X, Y	53
V.2. Test d'un simple usinage	54
IV.4. Conclusion	55
Conclusion générale	56

Table des figures

Figure I.1 : Première MOCN en 1952.....	3
Figure I.2 : Productivité comparée de diverses machines en fonction de leur degré d'automatisation	7
Figure I.3 : domaine d'utilisation des machines-outils à commande numérique	10
Figure I.3 : Structure d'une machine à commande numérique.	11
Figure I.4. Schéma d'une machine à commande numérique	12
Figure II.1 : Les trois fenêtres du logiciel SolidWorks.	16
Figure II.2 : Moteur pas à pas.	16
Figure II.3 : Palier.	17
Figure II.4 : Rendu final de l'accouplement	18
Figure II.5 : Vis à bille.	18
Figure II.6 : Glissière de guidage	19
Figure II.7 : Rail.	19
Figure II.8 : Axe d'une CNC.	20
Figure II.9 : Schéma cinématique d'un axe.	21
Figure II.10 : La broche.	21
Figure II.11 : Rendu final de la machine CNC.	22
Figure III.1 : Structure d'une commande CNC.	24
Figure III.2 : Organigramme de l'interpréteur	27
Figure III.3 : Organigramme de réception de commandes	28
Figure III.4 : Organigramme de traitement de commandes	29
Figure III.5 : approximation d'un arc en segments de droite.	30
Figure III.6 : Un segment de droite échantillonné	31

Figure III.7 : Algorithme de tracer de segment de droite	32
Figure III.8 : Eléments de la commande des moteurs pas à pas.	33
Figure III.9 : Schéma d'un moteur pas à pas bipolaire	34
Figure III.10 : Circuit de commande d'un moteur pas à pas bipolaire	35
Figure III.11 : Schéma d'un moteur pas à pas unipolaire	35
Figure III.12 : Circuit de commande d'un moteur pas à pas unipolaire	36
Figure III.13 : Schéma d'un moteur pas à pas à réluctance variable.	36
Figure III.14 : Schéma d'un moteur pas à pas à aimant permanent	37
Figure III.15 : Schéma d'un moteur pas à pas hybride	38
Figure III.16 : Signaux de commande en pas entier, alimentation d'une seule phase	38
Figure III.17 : Signaux de commande en pas entier, alimentation de deux phases	39
Figure III.18 : Signaux de commande en demi-pas, alimentation d'une ou deux phases.	39
Figure III.19 : Signaux de commande en demi-pas optimisé.	40
Figure III.20 : ARDUINO MEGA 2560.	41
Figure III.21 : Interface du logiciel ARDUINO.	42
Figure III.22 : Structure de base d'un programme ARDUINO	43
Figure III.23 : Circuit de base pour interface entrées sorties	45
Figure IV.1 : Schéma du circuit d'interface	46
Figure IV.2 : Schéma PCB du circuit d'interface	47
Figure IV.3 : Schéma des pistes cuivrées à imprimer.	48
Figure IV.4 : Ensemble des produits chimiques utilisés	49
Figure IV.5 : Circuit d'interface réalisé	50
Figure IV.6 : Contrôleur moteur pas à pas	50

Figure IV.7 : armoire de commande	52
Figure IV.8 : Le bâti.	52
Figure IV.9 : La broche.	53
Figure IV.10 : Les axes X, Y	53
Figure IV.11 : Simulation d'usinage avec WinUnisoft.	54
Figure IV.12 : Résultat d'usinage.	55

Table des tableaux

Tableau I.1 : Comparaison entre MO conventionnelles et MOCN	6
Tableau III.1 : Exemple de commande G-code et M-code.	26
Tableau III.2 : Caractéristiques de l'ARDUINO MEGA 2560.	41
Tableau IV.1 : Liste des composants utilisés dans le circuit d'interface	47
Tableau IV.2 : Description des pins du contrôleur	51

Introduction générale

L'usinage ou l'obtention de pièces mécaniques sous contrôle numérique s'étend désormais à l'ensemble des secteurs de l'industrie. Réservée, il y'a pas si longtemps, à certaines industries de pointe, les machines-outils à commande numérique sont maintenant utilisées dans toutes les usines de fabrications mécaniques et sont accessibles aux petites et moyennes entreprises. La connaissance de ces machines est devenue nécessaire dans l'industrie en générale.

Durant ces dernières années, la commande des machines électriques a subi des progrès significatifs. Ces progrès sont essentiellement dus à la révolution technologique en informatique industriel, ce qui a permis le développement des solutions numériques efficaces avec une possibilité d'implanter des algorithmes plus complexes. Ces commandes sont en majorité basées sur les microprocesseurs, les DSP (Digital Signal Processor) et les microcontrôleurs PIC.

Aujourd'hui, les machines à commande numérique par ordinateur (CNC) permettent une production économique et rentable. Le contrôle des coûts reste une préoccupation importante.

Notre projet de fin d'étude a pour objectif de réaliser une machine-outil trois axes ainsi que sa commande numérique (CNC) d'une manière simple et performante. Pour cela notre travail est divisé en quatre parties :

Dans le premier chapitre on présentera d'une manière générale le fonctionnement d'une machine-outil où on donne l'importance de la commande numérique pour l'amélioration de la production.

Le deuxième chapitre est consacré à la présentation des éléments de la machine-outil on utilisant le logiciel SolidWorks.

Dans le troisième chapitre on s'intéressera à la commande de la machine-outil on présentant les différents éléments intervenant dans cette commande.

La réalisation de notre machine sera donnée dans le quatrième chapitre où on va regrouper toutes les parties pour avoir un fonctionnement rigoureux.

Chapitre I :

***GENERALITES SUR LA
MACHINE-OUTIL A
COMMANDE NUMERIQUE***

I.1. Introduction

La commande numérique désigne, dans le domaine de la fabrication mécanique, l'ensemble des matériels et logiciels ayant pour fonction de donner les instructions de mouvements à tous les éléments d'une machine-outil. La programmation de commande numérique (CN) permet de définir des séquences d'instructions permettant de piloter des machines-outils.

Une machine-outil à commande numérique (MOCN) est une machine-outil dotée d'une commande numérique, elle permet selon ses caractéristiques d'effectuer diverses opérations - percer, scier, rectifier, découper, fraiser, plier, graver, tarauder, souder, visser, déposer un matériau, etc - nécessitant des gestes précis et/ou répétitifs, sur des matériaux divers. Lorsque la commande numérique est assurée par un ordinateur, on parle de machine CNC pour computer numerical control, francisé en « commande numérique par ordinateur ». De nos jours, elle est très utilisée dans le domaine de la fabrication mécanique.

I.2. Historique de la Commande Numérique

En 1947, dans l'État du Michigan, John Parsons fabrique pour le compte de l'US Air Force des pales d'hélicoptère par reproduction. Pour façonner ses gabarits, il utilise une méthode consistant à percer plusieurs centaines de trous faiblement espacés de manière à approcher le profil théorique. Mais, lorsque l'US Air Force confie à ce même Parsons la réalisation de pièces de formes encore plus complexes pour ses futurs avions supersoniques, celui-ci réalise que sa méthode est trop approximative et que seul un usinage continu en 3 dimensions sera en mesure de donner satisfaction.

Au printemps 1949, il confie alors au Massachusetts Institute of Technology (MIT) le soin de développer des asservissements capables de piloter une machine qui recevra des instructions intermittentes à partir d'un lecteur de cartes.

Cette machine, une fraiseuse prototype à broche verticale figure (I.1), conçue pour exécuter des déplacements simultanés suivant 3 axes, est officiellement présentée en septembre 1952 dans le Servomechanisms Laboratory du MIT. L'information mathématique étant la base du concept, on lui donne le nom de numerical control. [1] [2]

Il faut encore attendre quelques années de vastes fonds de l'US Air Force et l'appui des chercheurs du MIT pour rendre la première MOCN réellement opérationnelle. Les différentes étapes de développement de la CN sont les suivantes :

- **1954** : Bendix acquiert le brevet de Parsons et fabrique la première CN industrielle
- **1955** : à Font du Lac (Wisconsin), le constructeur américain Giddins & Lewis commercialise la première MOCN.
- **1959** : apparition de la CN en Europe (foire de Hanovre). Le MIT annonce la création du langage de programmation APT (Automatic Programed Tools).
- **1960** : apparition du système DNC (Direct Numerical Control)
- **1964** : en France, la Télémécanique Électrique lance la CN NUM 100 conçue à base de relais Téléstatic.
- **1968** : la CN adopte les circuits intégrés ; elle devient plus compacte et plus puissante.
- **1972** : les mini-calculateurs remplacent les logiques câblées ; la CN devient CNC.
- **1976** développements des CN à microprocesseurs.
- **1984** : apparition de fonctions graphiques évoluées et du mode de programmation conversationnel. début de l'ère de la fabrication assistée par ordinateur (FAO).
- **1986** : les CN s'intègrent dans les réseaux de communication, début de l'ère de la fabrication flexible (CIM : computer integrated manufacturing).
- **1990** : développement des CN à microprocesseurs 32 bits.

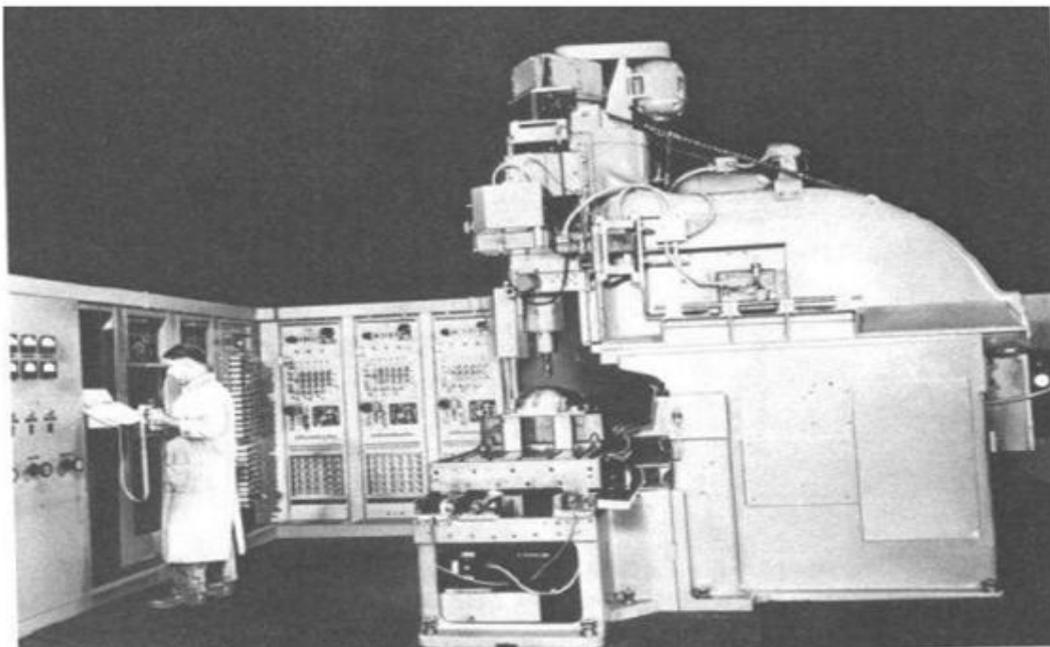


Figure I.1 : Première MOCN en 1952. [2]

I.3. Justification de la commande numérique

I.3.1. Automaticité

Le premier avantage d'une CN est d'offrir aux machines qui en sont équipées un très haut niveau d'automaticité. Sur de telles machines, l'intervention de l'opérateur nécessaire pour assurer la production de pièces peut être considérablement réduite voire supprimée.

De nombreuses MOCN peuvent ainsi fonctionner sans aucune assistance pendant toute la durée de leur cycle d'usinage, laissant l'opérateur libre d'accomplir d'autres tâches en dehors du poste de travail. Cette caractéristique présente un certain nombre d'avantages importants, tels que

- ✓ Une diminution notable de la fatigue de l'opérateur.
- ✓ Moins d'erreurs d'origine humaine.
- ✓ Un temps d'usinage constant et prévisible pour chaque pièce d'une même série. [2]

I.3.2. Flexibilité

Puisqu'elles sont pilotées à partir d'un programme, les MOCN peuvent usiner des pièces différentes aussi facilement que l'on charge un nouveau programme. Une fois vérifié puis exécuté pour la première série, ce programme peut être facilement rappelé lorsque la même série se représente.

Une MOCN se caractérise en outre par des temps de réglage très courts qui répondent parfaitement aux impératifs de la production en flux tendus.

La grande souplesse d'utilisation de la CN entraîne une quantité non négligeable d'autres avantages :

- ✓ Changement aisé du programme d'usinage des pièces ;
- ✓ Réduction du temps de fabrication ;
- ✓ Réduction des outillages et suppression des gabarits ;
- ✓ Diminution du nombre des outils spéciaux et des outils de forme ;
- ✓ Réduction des temps de préparation et de réglage du poste de travail (la plupart des réglages, en particulier des outils, étant effectués hors machine) ;
- ✓ Prise en compte rapide des modifications d'usinage (il est plus facile de modifier une ligne de programme qu'un outillage spécial ou un gabarit) ;
- ✓ Définition plus rapide et plus fiable des conditions optimales d'usinage ;
- ✓ Réduction du nombre de prises de pièces du fait de l'universalité de la machine ;

- ✓ Diminution du temps d'attente entre les diverses machines d'usinage d'un atelier ;
- ✓ Gain sur les surfaces au sol occupées dans l'atelier ;
- ✓ Possibilité de réaliser des pièces complexes en gérant des déplacements simultanés sur plusieurs axes ;
- ✓ Définition plus rapide et plus fiable des conditions optimales d'usinage ;
- ✓ Contrôle automatique des outils et des dimensions de pièces avec prise en compte par la CN des corrections à effectuer. [2]

I.3.3. Sécurité

La CN a beaucoup contribué à améliorer la sécurité des machines :

- ✓ En premier lieu, parce qu'elle connaît très précisément l'enveloppe de travail dans laquelle doivent évoluer les outils (possibilité de mémorisation des courses maximales des organes mobiles) ;
- ✓ Elle permet une simulation graphique hors usinage des programmes nouvellement créés pour vérification et détection des risques éventuels de collision ;
- ✓ En exerçant une surveillance permanente de l'usinage en cours, elle peut décider d'en interrompre le déroulement et d'alerter l'opérateur en cas d'incident.

Les constructeurs prévoient des dispositifs de protection très élaborés (contre les projections de copeaux ou de liquide d'arrosage, notamment) qui ne s'imposent pas nécessairement sur une MO conventionnelle. [2]

I.3.4. Nécessités économiques et techniques

Symbole de précision, de répétabilité, de fiabilité et de flexibilité, qualités primordiales dans une économie de marché où les produits se caractérisent en termes de prix, de qualité et de délai de mise à disposition, la CN se montre économiquement intéressante pour produire à l'unité ou en série toutes les sortes de pièces, même les plus simples.

Une fois vérifié et validé, un programme assure la réalisation de 2, 10 ou 1 000 pièces identiques avec la même régularité de précision et la même qualité d'usinage, sans que l'habileté de l'opérateur n'intervienne. Il convient, en outre, de souligner que la CN ouvre de nouvelles perspectives en permettant la définition de pièces complexes qu'il est pratiquement impossible de concevoir et de fabriquer sur des MO conventionnelles. [2]

I.4. Comparaison entre MO conventionnelles et MOCN :

Un exemple pratique des temps de réalisation d'une pièce sur machine à commande numérique a abouti aux résultats donnés dans le tableau suivants :

	Usinage sur machine conventionnelle	Usinage sur perceuse CN
Usinage de la première pièce	<p>mise en train; 01h10 usinage; 05h45</p>	<p>gain; 01h15 mise en train; 01h10 usinage; 01h30 Programme; 03h00</p>
Usinage de la deuxième pièce	<p>usinage; 05h45</p>	<p>gain; 04h15 usinage; 01h30</p>
Usinage d'une série de 10 pièces	<p>Usinage 1ère pièce : 6h55 Usinage 2ème à 10ème pièces 9 fois 5h45 Total : 58h40</p>	<p>Usinage 1ère pièce : 5h40 Usinage 2ème à 10ème pièces 9 fois 1h30 Total : 19h10</p>

Tableau I.1 : Comparaison entre MO conventionnelles et MOCN. [1]

Une autre façon de percevoir l'intérêt économique de la commande numérique est d'évaluer le temps où une machine-outil effectue le travail pour lequel a été conçue à savoir enlever de la matière, c'est-à-dire faire des copeaux.

Analyse sur les temps de production effective :

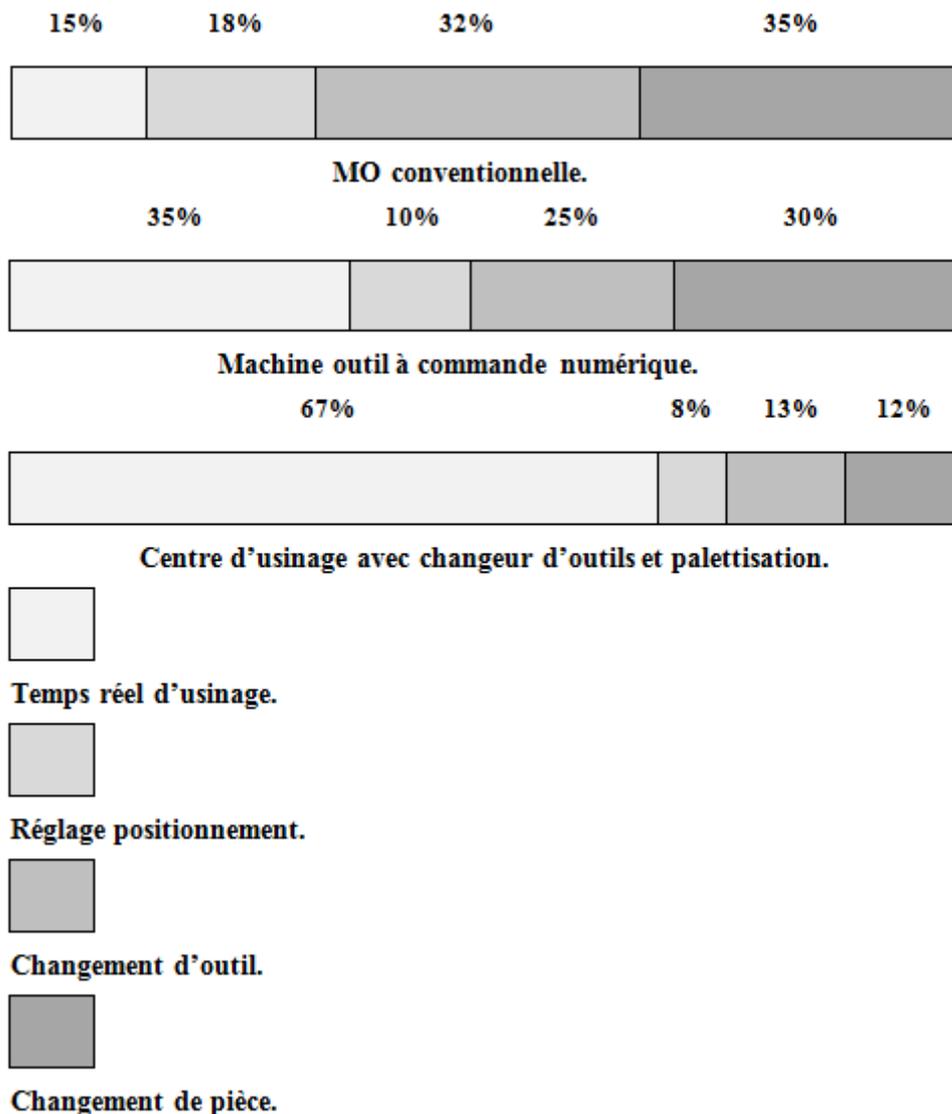


Figure I.2 : Productivité comparée de diverses machines en fonction de leur degré d'automation. [1]

I.5. Les différentes techniques d'usinage avec les MOCN [3]

Pour donner à une matière brute une forme précise, l'usinage fait appel à de nombreuses techniques complémentaires comme le perçage, le fraisage, le tournage, l'alésage ...etc.

I.5.1. Le perçage

Le perçage consiste à faire un trou dans la matière à l'aide d'un outil de coupe en rotation. Le perçage est un usinage basique et stratégique en mécanique. 25% des usinages sont de fait des perçages. Les trous de perçage peuvent être réalisés avec une forêt ou une fraise. Selon le diamètre du trou à percer, les outils seront plus ou moins précis.

I.5.2. Le fraisage

Le fraisage consiste à enlever de la matière sur une pièce mécanique pour pratiquer notamment une gorge, un trou cylindrique, et globalement tout type de formes en creux même complexes. Le fraisage est réalisé à l'aide d'une fraise que l'on monte sur une fraiseuse. Cette machine-outil dédiée fait partie des indispensables au sein de toutes les usines mécaniques. La fraise rogne la matière en formant des copeaux. Le fraisage combine deux mouvements : la rotation de la fraise et l'avance de la pièce à usiner. Le fraisage et le tournage sont généralement des techniques complémentaires utilisées par des tourneurs-fraiseurs.

I.5.3. Le tournage

Le tournage consiste comme le fraisage à enlever de la matière sur une pièce mécanique. Le tournage se distingue toutefois du fraisage par le fait qu'il ne permet qu'un mouvement rotatif. La pièce à façonner est en effet fixée dans un mandrin ou une pince. C'est elle qui tourne. L'outil coupant peut rogner l'intérieur ou l'extérieur de la pièce selon un mouvement d'avance. Les outils utilisés sur un tour permettent d'obtenir le plus souvent des cylindres pleins ou creux mais aussi toutes sortes de formes même complexes. Un tour de décolletage par exemple peut fabriquer des vis, des boulons... etc.

I.5.4. Le filetage

Le filetage consiste à réaliser dans un trou une empreinte fileté de type spirale. Le filetage permet notamment de préparer une pièce à recevoir un assemblage par vis ou tige fileté.

I.5.5. L'alésage

L'alésage consiste à retoucher avec précision l'intérieur d'un cylindre ou d'une pièce creuse pour obtenir une surface lisse. L'alésage est une étape de finition qui intervient après un perçage le plus souvent. Le perçage forme l'ébauche, l'alésage rectifie de façon homogène la surface percée en éliminant notamment les bavures et les imperfections existantes. La

précision de régularité de surface obtenue dépend des outils utilisés : foret, alésoir, ou outils spéciaux.

I.6. Domaine d'utilisation technique et économique des MOCN

I.6.1. Domaine d'utilisation technique

Les MOCN sont employées dans de nombreux secteurs industriels. Elle est aussi associée à de nouvelles technologies de façonnage (Laser, Électro-érosion, Jet d'eau).

Les principaux débouchés de l'industrie des MOCN sont les secteurs de :

- l'automobile,
- l'horlogerie traditionnelle,
- l'aéronautique,
- le médical,
- l'électronique,
- l'aérospatiale,
- l'électroménager,
- la connectique,
- le machinisme agricole,
- le bâtiment,
- les travaux publics,
- l'armement,
- l'industrie électrique et nucléaire.

I.6.2. Domaine d'utilisation économique

Le système de fabrication le plus rentable est celui qui engendre le coût d'une pièce le plus bas. Ce coût est calculé par la formule suivante :

$$C = Cu + \frac{Cr}{L} + \frac{Cp}{ZL} \quad (\text{I.1})$$

Avec :

- C : coût total de fabrication pour une pièce,
- C_u : coût d'usinage d'une pièce (matière, main d'œuvre directe, coût machine),
- C_r : coût de lancement de la série et des réglages des outils et de la machine,
- L : nombre de pièces d'une série,
- C_p : coût de préparation (gammes et programmes d'usinage) et des outillages,
- Z : nombre de séries,
- ZL : nombre total de pièces fabriquées.

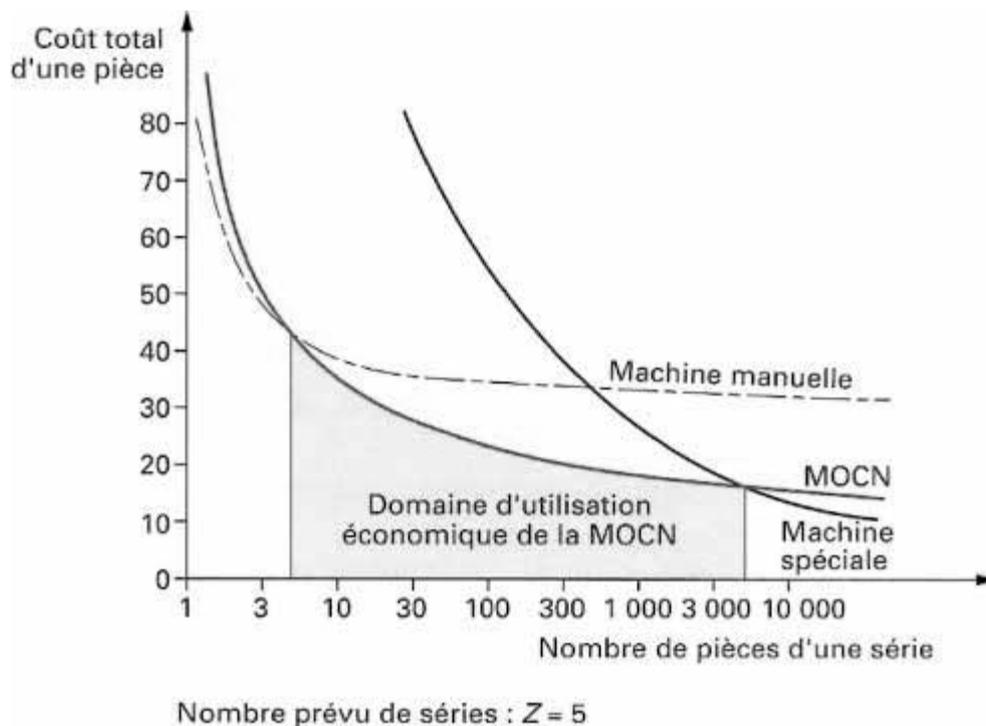


Figure I.3 : domaine d'utilisation des machines-outils à commande numérique [1]

On constate que le coût total de fabrication par pièce varie en fonction de la quantité ZL d'une manière hyperbolique.

Si l'on considère le nombre de pièces usinées, le domaine d'utilisation économique de la machine-outil à commande numérique se situe dans la petite et la moyenne séries. Les machines-outils conventionnelles restent rentables pour des opérations simples où elles ont malgré tout tendance à être remplacées par des MOCN d'entrée de gamme.

Pour les grandes séries, le recours à des machines spéciales à automatisation rigide se montre encore très avantageux.

Si l'on représente le coût d'une pièce en fonction du nombre d'exemplaires à fabriquer, on peut déterminer les limites économiques d'utilisation de la commande numérique. Dans l'exemple de la figure 3, au-dessus de 5 pièces par série, l'usinage sur une MO à commande manuelle est plus rentable que sur une MOCN ; de la même façon, une machine spéciale le sera au-dessus de 5 000 pièces par série. [1]

I.7. Architecture et organisation générale d'une MOCN

Comme pour tout système automatisé, nous pouvons décomposer la structure d'une MOCN en deux parties principales :

- Partie opérative : agit directement sur le produit, c'est la machine-outil.
- Partie commande : élabore des ordres en fonction des consignes.

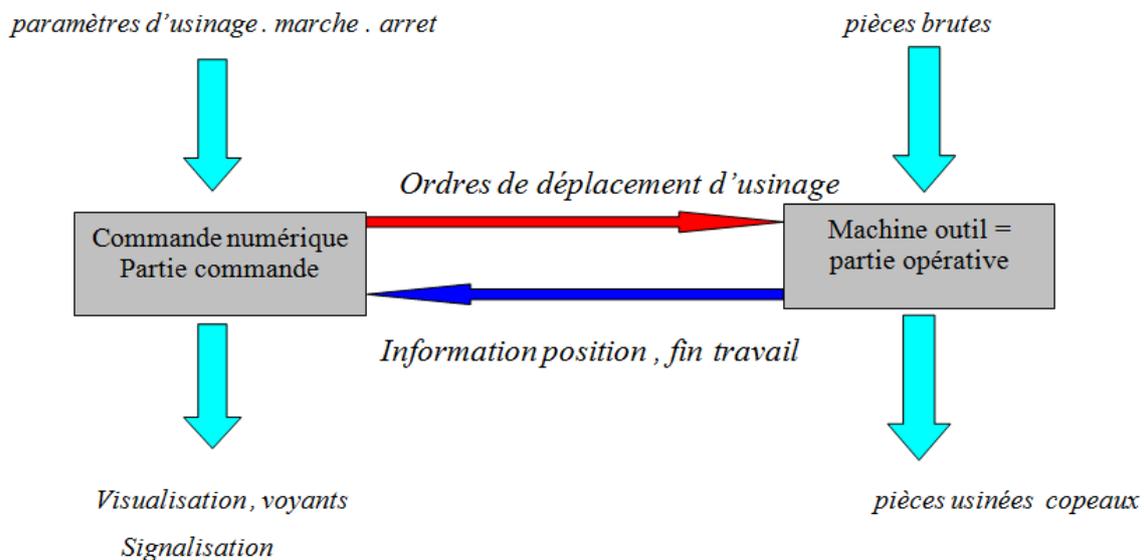


Figure I.3. Structure d'une machine à commande numérique.

Éléments constitutifs d'une CNC :

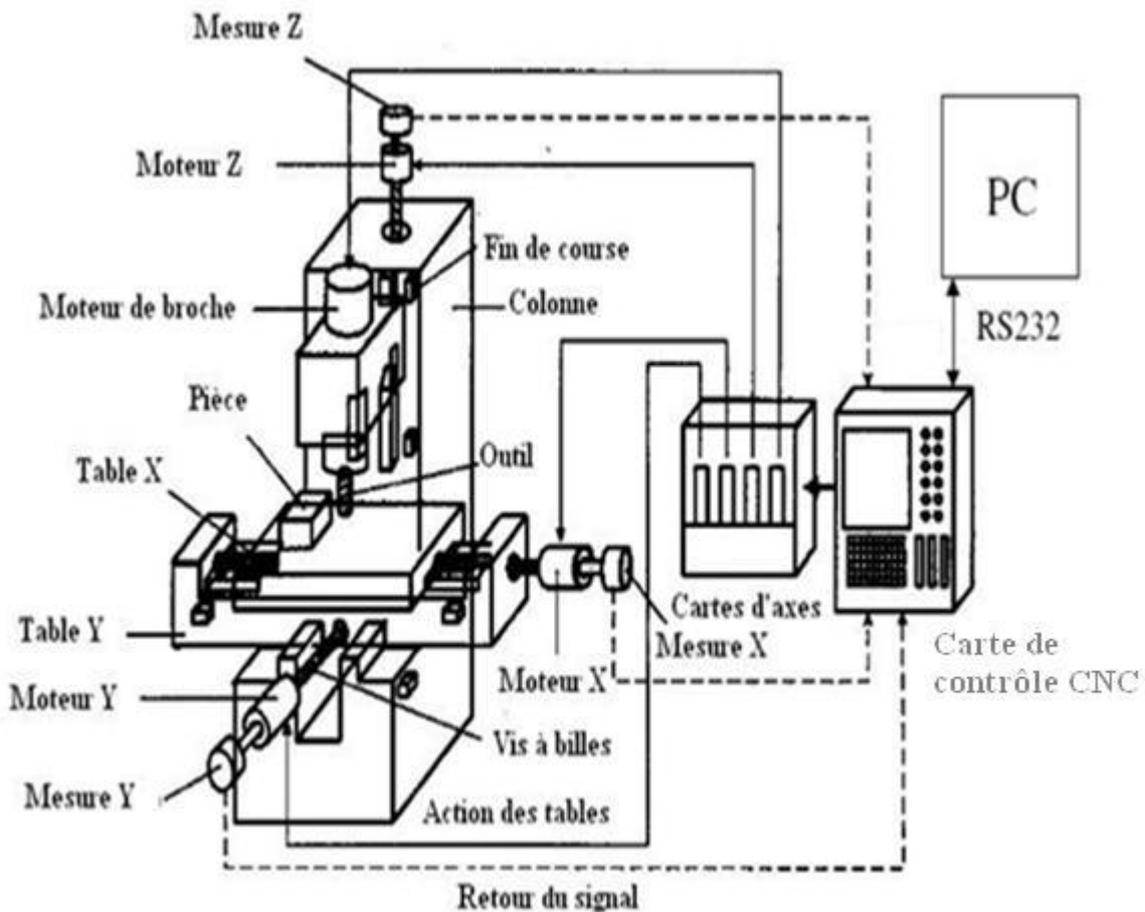


Figure I.4 : Schéma d'une machine à commande numérique.

Une CNC est composée principalement de :

- **Ordinateur :** contenant les logiciels nécessaires pour la conception de la pièce (CAO : Conception Assistée par Ordinateur), le calcul des parcours d'outil, sous forme d'un code (exemple: G-code), pour usiner cette pièce (FAO : Fabrication Assistée par Ordinateur).
- **Carte de Contrôle CNC :** sa fonction principale est d'interpréter le code généré par le FAO et d'envoyer les signaux nécessaires aux cartes d'axes pour commander les moteurs et la broche.
- **Les moteurs :** Les déplacements peuvent être assurés par plusieurs types de moteurs
 - Moteurs pas à pas.
 - Moteurs à courant continu.
 - Moteur à courant alternatif.

- **Les axes :** permettant le mouvement dans les trois directions X, Y, Z (pour une CNC 3 axes). le mouvement est assuré par des glissières posées sur des rails et une vise-écrou (ou vise-écrou à billes, poulie courroie) accouplée à un moteur.
- **La Broche :** l'ensemble formé par le système de fixation de l'outil et le moteur permettant la rotation de cet outil.
- **Les capteurs :**
 - Des capteurs de fin de course : pour ne pas dépasser certaines limites.
 - Des capteurs de position et/ou vitesse.

I.8 Avantages et les inconvénients [2]

I.8.1 les avantages

- Réduction du temps improductif
- Accroissement du degré de précision et de répétabilité
- Réduction des taux de rebuts
- Réduction de la nécessité de contrôle
- Possibilité de traiter des géométries complexes
- Facilité d'adapter les changements techniques
- Réduction des temps de mise en production
- Réduction des stocks
- Réduction des espaces requis
- Réduction du niveau de qualification des opérateurs

I.8.2. les inconvénients

- Coûts d'investissement élevés
- Plus d'effort doit être fourni pour la maintenance
- Nécessité de programmation des équipements de CN

- Pour bénéficier de la majorité des avantages précédents, il faut que tout le parc machine de l'entreprise soit des MOCN

I.9. Conclusion

On a présenté dans ce chapitre un aperçu général sur la technologie des machines – outils à commande numérique (MOCN), en citant les différentes techniques d'usinage, les avantages et leurs intérêt ce qui donne une bonne motivation pour se lancer dans ce projet de réalisation.

Chapitre II :

***CONCEPTION DE LA
MACHINE CNC***

II.1. Introduction

Dans ce chapitre on détaillera les composants de notre machine CNC, en présentant des modèles de pièces et d'assemblages réalisés par le logiciel SolidWorks. L'assemblage virtuel nous permet de détecter des éventuels défauts dans la structure de la machine.

II.2. Conception assistée par ordinateur (CAO)

La conception assistée par ordinateur (CAO) comprend l'ensemble des logiciels et des techniques de modélisation géométrique permettant de concevoir, de tester virtuellement à l'aide d'un ordinateur et des techniques de simulation numérique et de réaliser des produits manufacturés et les outils pour les fabriquer.

Un logiciel de CAO se compose généralement de quatre parties majeures qui peuvent être organisées comme suit :

- **Le modeleur géométrique** : il représente "la planche à dessin". Nous trouvons dans cette partie les composants géométriques essentiels : points, droites, cercles, ellipses, plans, sphères, cylindres, cônes, surfaces de révolution, surfaces de balayage, etc. Il intègre également les composants topologiques : sommets, faces, arêtes, orientations, coïncidences, adjacences, intersections, soustractions, unions, etc.
- **L'outil de visualisation.**
- **Un certain nombre d'applications** : nous retrouvons le calcul des grandeurs géométriques (distances, inerties, volumes, masses, etc.), les fonctions métiers : assemblage de pièces, production de plans, simulation d'usinage, moulage, fraisage, etc.
- **Un contrôleur** : il gère et manipule les intersections entre les trois outils cités précédemment [17].

II.2.1. Description du logiciel « SOLIDWORKS »

SOLIDWORKS est un modeleur 3D utilisant la conception paramétrique. Il génère 3 types de fichiers relatifs à trois concepts de base : la pièce, l'assemblage et la mise en plan. Ces fichiers sont en relation. Toute modification en un niveau sera transmise aux deux autres fichiers concernés.

Un dossier complet contenant l'ensemble des relatifs à un même système constitue une maquette numérique. De nombreux logiciels viennent compléter l'éditeur SOLIDWORKS. Des utilitaires orientés métiers (tôlerie, bois, BTP...), mais aussi des applications de

simulation mécanique ou d'image de synthèse travaillent à partir des éléments de la maquette virtuelle.

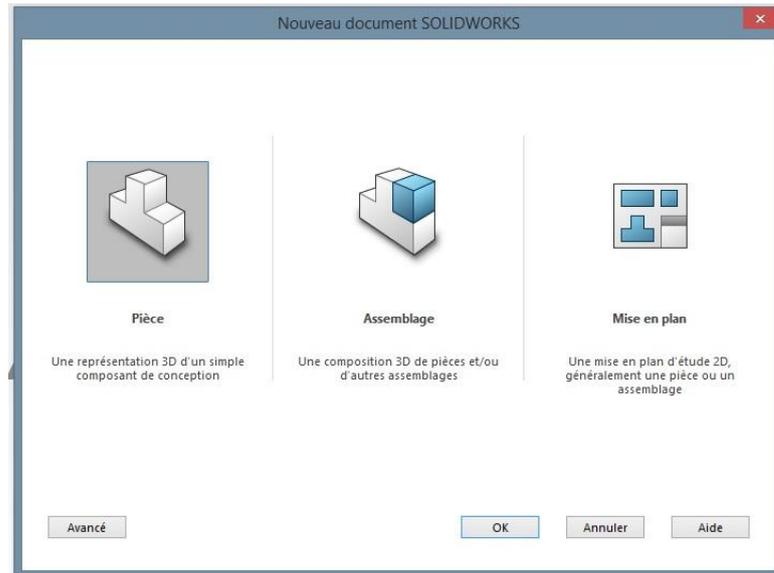


Figure II.1 : Les trois fenêtres du logiciel SolidWorks.

II.3. Conception de la machine

II.3.1. Moteur pas à pas

Les moteurs pas-à-pas diffèrent, par leur mode de commande, des moteurs classiques. Les moteurs pas à pas permettent de transformer un signal électrique numérique en un mouvement angulaire. Chaque impulsion envoyée par le système de commande au module de puissance se traduit par la rotation d'un pas du moteur. La résolution angulaire d'un moteur pas à pas va de 4 à 400 pas.



Figure II.2 : Moteur pas à pas.

II.3.2. Palier

Les paliers sont des organes utilisés en construction mécanique pour supporter et guider, en rotation, des arbres de transmission.

Suivant l'usage désiré, ces paliers peuvent être :

- ✓ Lisses où les arbres qui reposent sur des coussinets sont soumis au frottement de glissement entre les surfaces en contact.
- ✓ À roulement où le contact s'effectue par l'intermédiaire de billes ou de rouleaux contenus dans des cages. On a là un phénomène de résistance au roulement (parfois appelé improprement « frottement de roulement ») qui permet une plus grande charge sur les paliers et une plus grande vitesse de rotation.



Figure II.3 : Palier.

II.3.3. Accouplement

En mécanique, un accouplement ou joint de transmission est un dispositif de liaison entre deux arbres en rotation, permettant la transmission du couple.

Un accouplement mécanique peut être un accouplement élastique, un accouplement à membranes, un accouplement à denture métallique ou encore un accouplement à soufflet.

On a utilisé des accouplements articulés pour accoupler les moteurs pas à pas et la vis à billes dans notre système. Ils permettent de corriger des défauts d'alignement très importants et les décalages angulaires. Il est constitué de deux bagues solidaires et un patin lié aux bagues par deux liaisons prismatiques



Figure II.4 : Rendu final de l'accouplement.

II.3.4. Vis à bille

Une vis à billes est un mécanisme assurant la conversion d'un mouvement de rotation en un mouvement de translation (liaison hélicoïdale). C'est un équivalent du mécanisme de vis-écrou, où des billes sont intercalées entre les deux pièces.

La présence des billes permet de diminuer les frottements qu'on rencontre dans un système vis-écrou simple. Les vis à billes s'imposent donc dans les cas :

- ✓ De transmission de puissance.
- ✓ De mouvements de précision, le remplacement du frottement par le roulement amenant la quasi disparition de l'usure des surfaces, et donc des jeux (mécanique).
- ✓ Où on recherche une réversibilité de la conversion rotation / translation.

La vis à billes est posée sur deux paliers, l'extrémité libre comporte un seul roulement et l'autre extrémité qui est accouplé au moteur possède deux roulements.



Figure II.5 : Vis à bille.

II.3.5. Glissière

Pièce destinée à guider le mouvement glissant d'une pièce mobile.

Notre système est constitué de quatre glissières dans chaque axe, ces chariots supporte la charge exercée sur la table.



Figure II.6 : Glissière de guidage.

II.3.6. Rail

Profilé de métal servant de guide à un mouvement de translation.

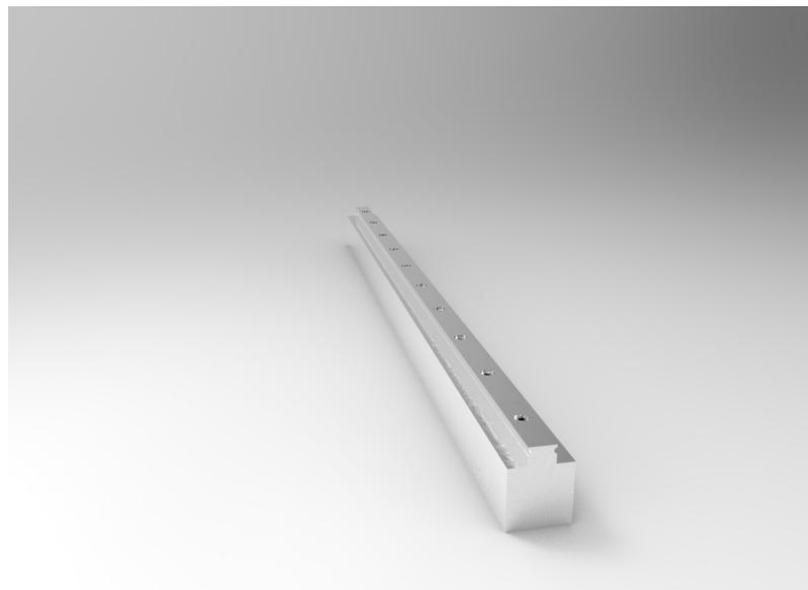


Figure II.7 : Rail.

II.3.7. Axe d'une CNC

Un degré de liberté d'un organe de machine est appelé AXE si l'actionneur du mouvement est asservi en vitesse et position, et s'il peut être synchronisé avec un autre degré de liberté pour obtenir un déplacement qui n'est pas parallèle à une direction principale du système de coordonnées.

Il permet d'obtenir une position, par une instruction numérique, à la résolution du moyen de mesure près.

Un axe est constitué de la façon suivante :

- ✓ Un chariot mobile sur glissières.
- ✓ Un système de transmission vis-écrou (vis à billes).
- ✓ Un moteur pas à pas.

La figure (II.8) représente le rendu final d'un axe réalisé par SolidWorks



Figure II.8 : Axe d'une CNC.

II.3.7.1. Schéma cinématique d'un axe

Chaque table présente quatre liaisons mécaniques qui sont :

- Une liaison encastrement qui représente l'accouplement du moteur.

- Deux liaisons pivot qui assurent la rotation de la vis dans les paliers à roulements.
- Une liaison hélicoïdale qui assure la rotation et translation du système vis-écrou à billes.
- Quatre liaisons glissière qui assurent la translation des chariots de guidage.

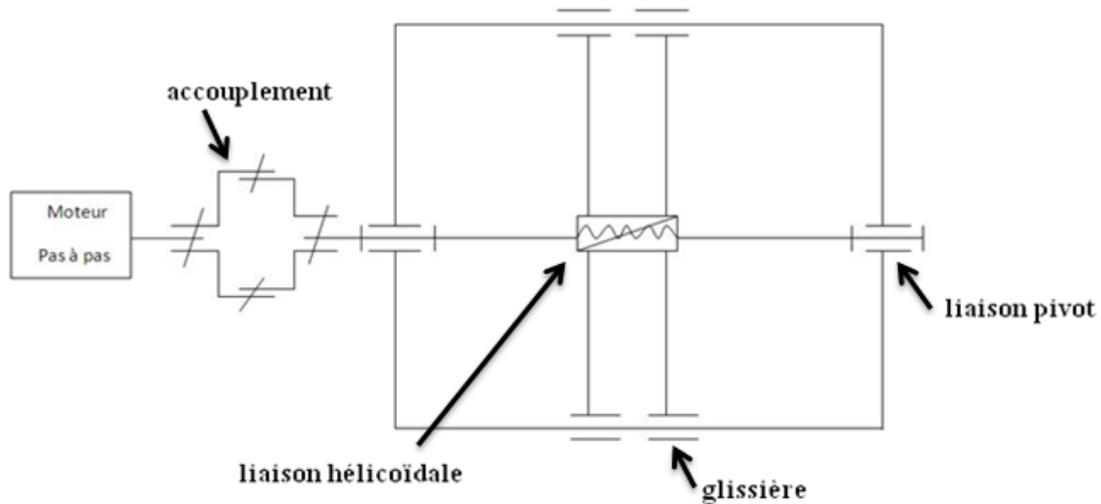


Figure II.9 : Schéma cinématique d'un axe.

II.3.8. La broche :

La broche de la machine est fixée sur le troisième axe, donc elle possède un mouvement vertical.

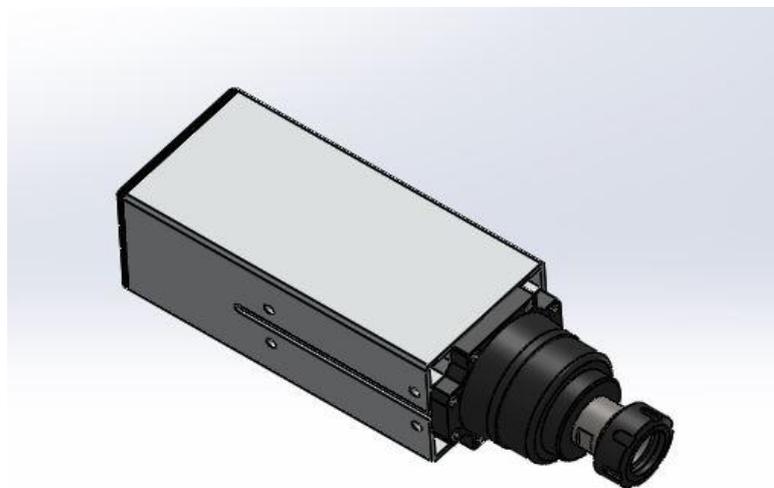


Figure II.10 : La broche.

II.3.9. Le bâti

Le bâti sert essentiellement à la fixation de la machine et la broche, et aussi à l'absorption des vibrations.

La figure (II.11) représente le rendu final de la machine CNC réalisé par SolidWorks

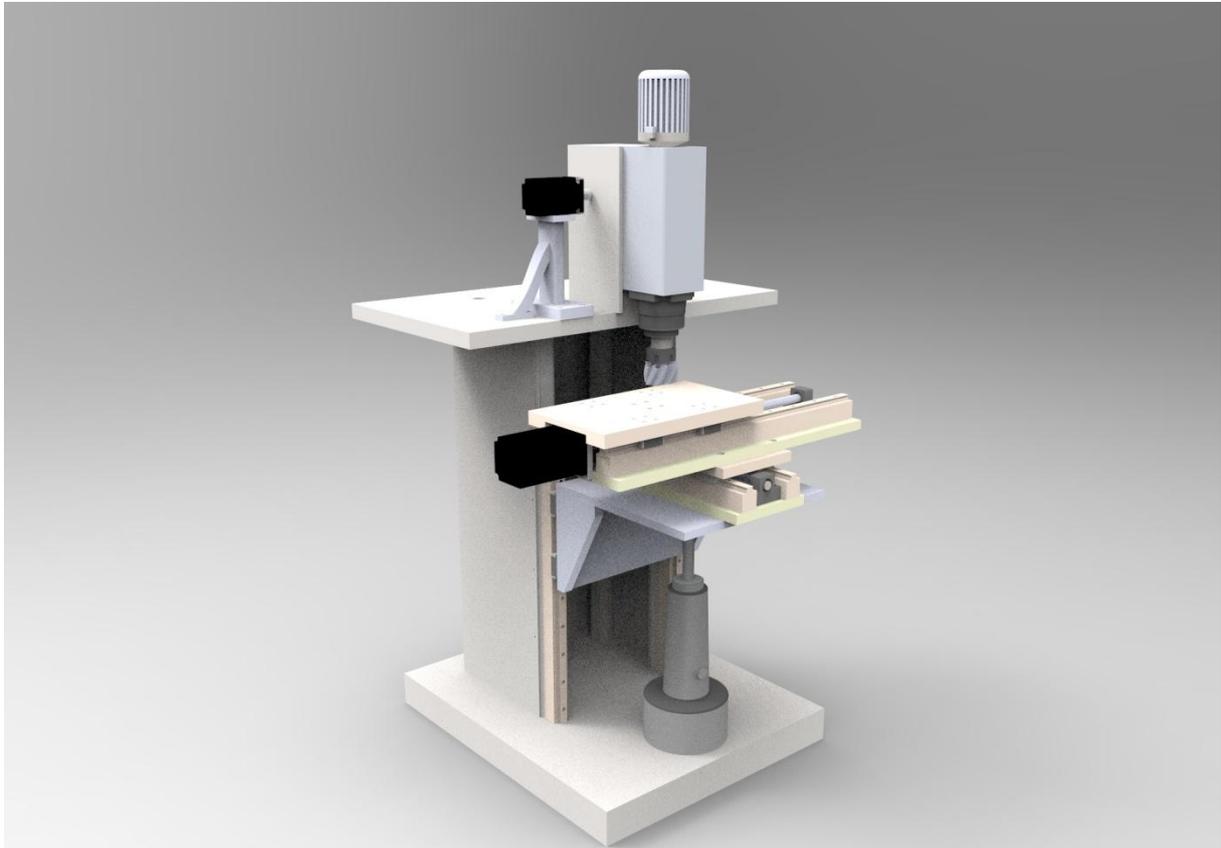


Figure II.11 : Rendu final de la machine CNC.

II.4. Conclusion

Dans ce chapitre nous avons réalisé les différentes pièces constituant notre machine en utilisant le logiciel SolidWorks ainsi nous avons expliqué l'utilité de chacune de ces pièces et vérifié la structure globale de la machine, ce qui nous permettra de passer à l'étape de réalisation aisément.

Chapitre III :

***COMMANDE DE LA
MACHINE CNC***

III.1. Introduction

Dans ce chapitre nous allons étudier les différents éléments intervenant dans la commande de notre machine-outil ainsi que l'organe principal commandé, à savoir le moteur pas à pas.

III.2. Présentation de la commande

On comprend que le moteur pas à pas répond à l'impératif de précision, mais le principe de commande est trop basique pour pouvoir être programmé directement par l'homme.

Dès lors, l'interface de commande de chaque contrôleur de moteur pas à pas sera reliée à un dispositif numérique afin de permettre un contrôle programmable par l'homme.

Ce rôle sera tenu soit par l'ordinateur, soit par des systèmes dédiés à cette tâche. Une troisième solution intermédiaire consiste à utiliser un ordinateur pour les calculs et l'affichage, couplé à un contrôleur de CNC pour le pilotage. La principale contrainte est d'être capable de délivrer les commandes au contrôleur de moteur (enable/step/direction) de manière très rapide et synchrone. L'ordinateur seul ne présente pas une solution idéale ; seul le port parallèle permet de répondre en partie à ces impératifs mais l'architecture matérielle n'est pas optimisée pour ce type d'utilisation et un ralentissement de quelques microsecondes ne peut être toléré. Compléter ou remplacer l'ordinateur par du matériel dédié est un peu plus coûteux mais permet un contrôle plus adapté aux contraintes des CNC. La figure (III.1) résume cette structure de commande.

Ce système de commande envoie donc des informations de contrôle des moteurs, mais peut aussi gérer l'activation et le réglage de la vitesse du moteur de la broche, l'aspiration des copeaux, l'adjonction d'eau ou d'huile de coupe, la ventilation des fumées, le changement d'outil automatique, etc. Le système de commande nécessite cependant d'être informé de certains événements, par exemple, la prise d'origine des axes (calibration du repère orthonormé). Cette opération requiert l'utilisation de contacteurs de position sur chaque axe, informant de manière précise le bout de course des axes.

Toutes ces possibilités peuvent être gérées par l'homme en utilisant un langage de programmation spécifique. Le langage le plus populaire aujourd'hui pour le contrôle des CNC

est le G-Code ; le dispositif numérique sera donc en mesure d'interpréter ce langage pour commander, entre autres, les contrôleurs de moteurs pas à pas. Ce langage met à disposition des instructions de déplacement de l'outil de coupe, de vitesse de coupe, de gestion de chemins, etc. Le langage peut-être écrit directement par l'homme d'après les dessins techniques du projet, en effectuant un certain nombre de calculs, pas toujours évidents, pour prévoir le passage des outils. Mais cette tâche de calcul du passage d'outils, ainsi que la création des dessins techniques sont aujourd'hui assistés par ordinateur.

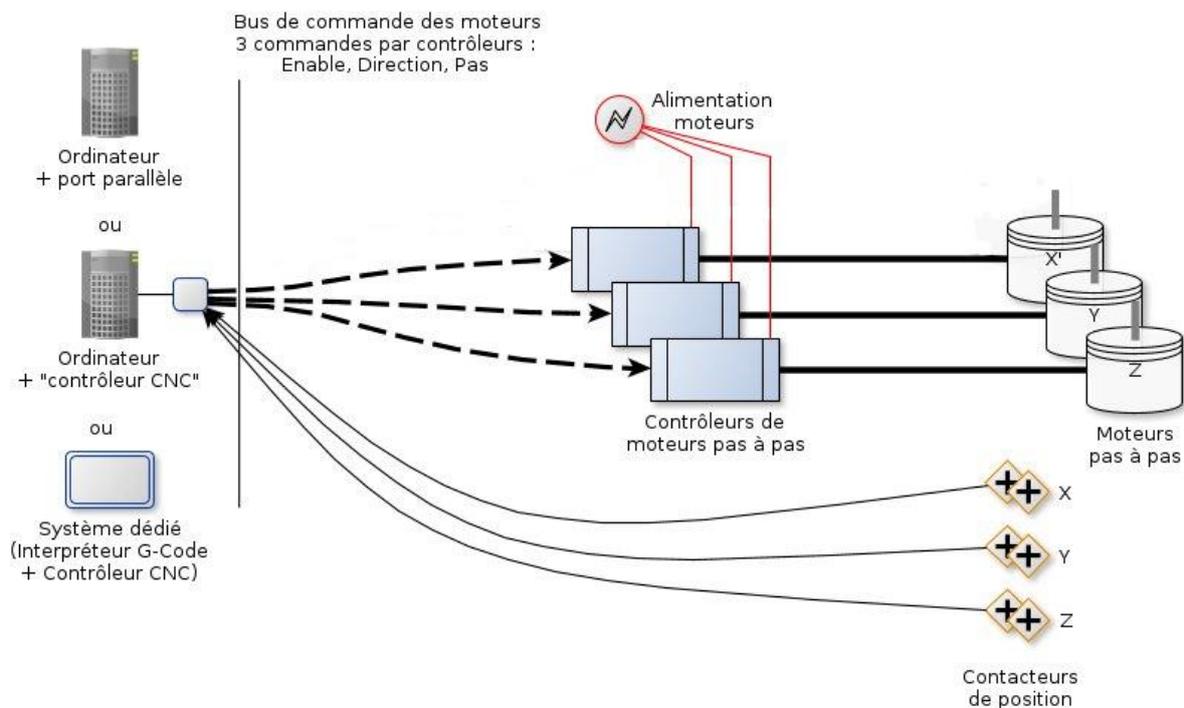


Figure III.1 : Structure d'une commande CNC.

Les outils informatiques qui interviennent dans le domaine des CNC sont rangés dans les catégories suivantes :

- **CAO (= CAD en anglais) :** Conception Assistée par Ordinateur. Permet de concevoir et d'élaborer les dessins techniques des pièces à réaliser. Les fichiers sont souvent des formats propriétaires contenant des informations de plans 2D ou de volumes 3D.
- **FAO (= CAM en anglais) :** Fabrication Assistée par Ordinateur. Permet de calculer les chemins des opérations d'usinages sous forme de parcours-outils. Les fichiers sont également propriétaires mais les outils permettent pour la plupart d'exporter ces parcours-outils en G-Code.

- **Interpréteurs G-Code et contrôleurs de CNC (G-Code parser and CNC controller en anglais)** : Permet d'interpréter le G-Code et de commander les contrôleurs de moteurs de la CNC.

III.3. G-Code

Le G-code est un langage de programmation des machines numériques il 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 se déplace 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 (G pour Général et M pour Miscellaneous (auxiliaire)). Les termes pour ces commandes sont G-codes et M-codes.

Le langage G-code n'a pas d'indicateur de début et de fin de programme. 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. [4]

Le tableau ci-dessous présente une liste non exhaustive de commandes G-code et M-code.

groupe	code
mode de mouvement	G0, G1, G2, G3, G38.2, G38.3, G38.4, G38.5, G80
Sélection du système de coordonnées	G54, G55, G56, G57, G58, G59
Sélection du plan	G17, G18, G19
Mode de déplacement (absolu/relatif)	G90, G91
Mode de déplacement en arc ijk	G91.1
Mode de vitesse de déplacement	G93, G94
Choix des unités	G20, G21
Compensation de rayon d'outil	G40
Compensation de longueur d'outil	G43.1, G49
contrôle programme	M0, M1, M2, M30
Contrôle de la broche	M3, M4, M5
Contrôle des arrosages	M7, M8, M9

Tableau III.1 : Exemple de commande G-code et M-code. [5][6][7]

III.4. Interpréteurs G-Code et contrôleurs de CNC

Comme nous venons de le voir, pour chaque pièce nous devons calculer les chemins de coupe sous forme d'un code G et de l'envoyer à la carte de commande. Il nous est donc nécessaire d'interpréter les stratégies générées en code G et de commander les déplacements, en pilotant directement les axes de la machine, en se basant sur cette interprétation pour pouvoir usiner la pièce. L'interprétation des commandes de mouvement est réalisée à l'aide des algorithmes d'interpolation.

III.4.1. Organisation du programme

La structure globale (simplifiée) de notre interpréteur, développé sous arduino, est donnée par l'organigramme suivant :



Figure III.2 : Organigramme de l'interpréteur.

A la mise sous tension de la carte arduino, le programme se lance et commence par une initialisation :

- Configuration de la vitesse du port série
- Configuration des pins E/S
- Initialisation des variables système

Une fois l'initialisation terminée, le programme fait appel à la fonction principale qui écoute le port série et fait appel à d'autres fonctions pour traiter les messages reçus. Ce processus se résume dans l'organigramme suivant :

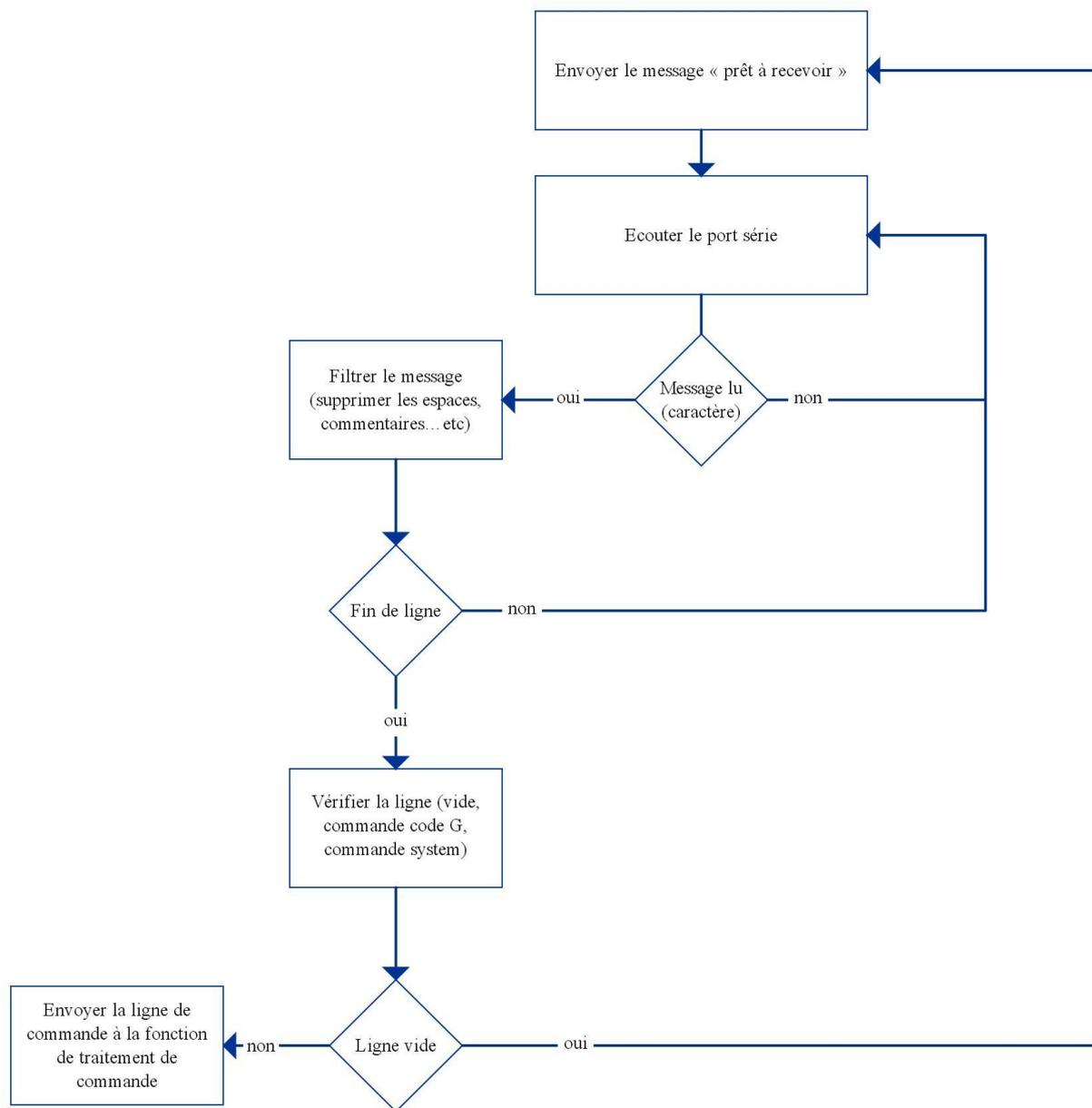


Figure III.3 : Organigramme de réception de commandes.

La dernière étape consiste à traiter la ligne de commande reçue. La structure de ce traitement est donnée par l'organigramme suivant :

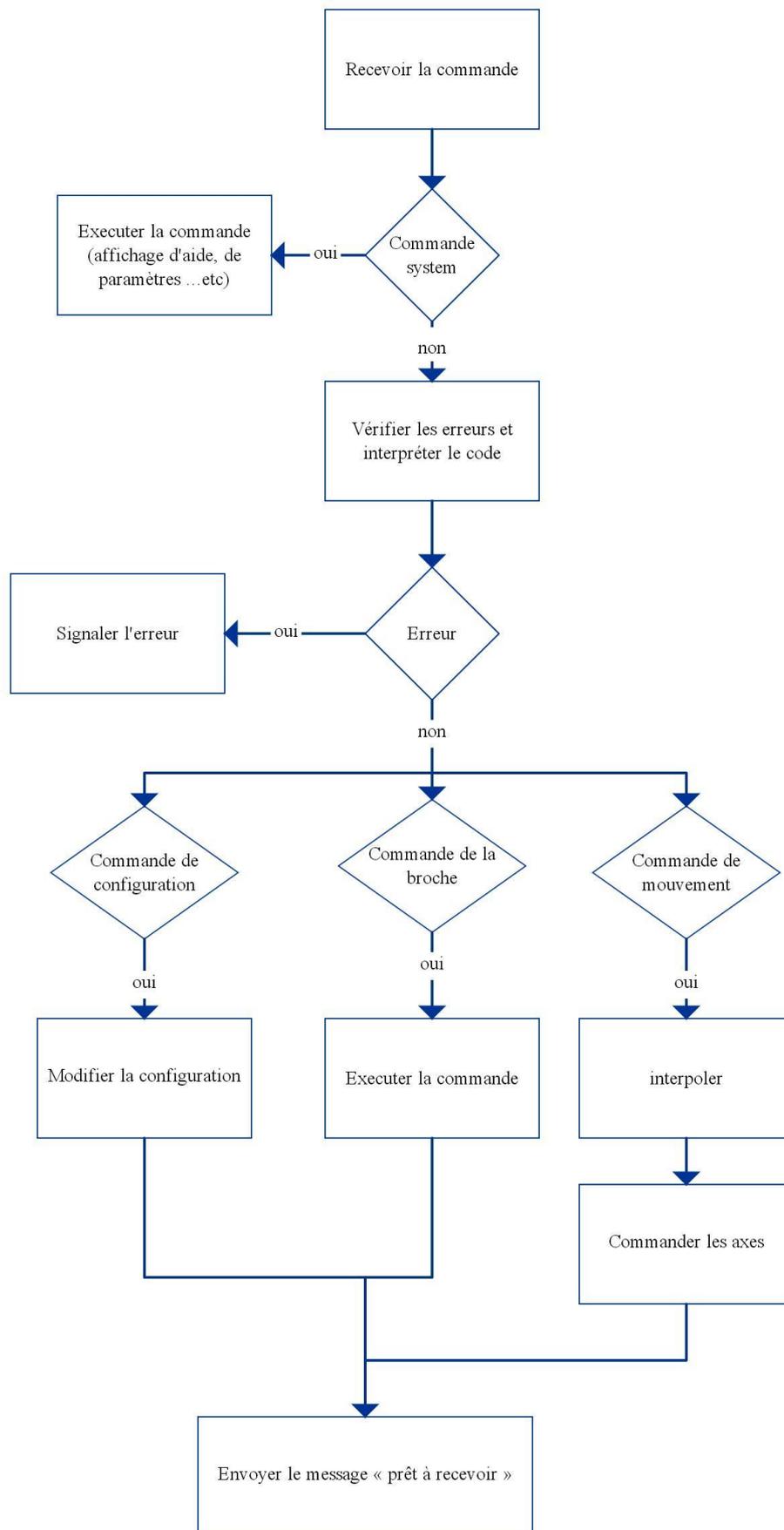


Figure III.4 : Organigramme de traitement de commandes.

III.4.2. Interpolation

En analyse numérique (et dans son application algorithmique discrète pour le calcul numérique), l'interpolation est une opération mathématique permettant de construire une courbe à partir de la donnée d'un nombre fini de points, ou une fonction à partir de la donnée d'un nombre fini de valeurs. La solution du problème d'interpolation passe par les points prescrits, et, suivant le type d'interpolation, il lui est demandé de vérifier des propriétés supplémentaires. Parmi les types d'interpolation on cite :

- Interpolation linéaire
- Interpolation cubique
- Interpolation polynomiale [10]

Dans ce projet on s'intéresse à deux types de mouvement :

- Linéaire : caractérisé par deux points, P1 (point de départ) et P2 (point d'arrivée). Les positions intermédiaires élémentaires (pas) sont calculées à l'aide d'une interpolation linéaire (dans notre cas l'algorithme de Bresenham).
- Circulaire : caractérisé par deux points (P1 et P2), un centre (O) et un sens de rotation (horaire ou antihoraire). Dans notre cas l'arc est décomposé en un nombre fini de segments de droite entre P1 et P2 (approximation de l'arc) comme le montre la figure, puis l'algorithme d'interpolation linéaire est utilisé pour calculer les déplacements élémentaires.

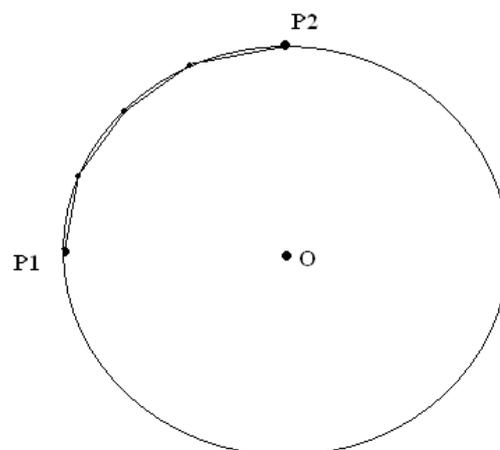


Figure III.5 : Approximation d'un arc en segments de droite.

III.4.3. Algorithme de tracé de segment de Bresenham

L'algorithme détermine quels sont les points d'un plan discret qui doivent être tracés afin de former une approximation de segment de droite entre deux points donnés. Cet algorithme est souvent utilisé pour dessiner des segments de droites sur l'écran d'un ordinateur ou une image calculée pour l'impression. Il est considéré comme l'un des premiers algorithmes découverts dans le domaine de la synthèse d'image. [11]

III.4.3.1. Principe

La ligne est tracée entre deux points (x_1, y_1) et (x_2, y_2) , où chaque paire indique la colonne et la rangée (des valeurs entières). Nous supposons au départ que notre segment monte vers le haut et la droite, et que la distance horizontale $x_2 - x_1$ excède la distance verticale $y_2 - y_1$ (c'est-à-dire que le segment a une pente inférieure ou égale à 1). Notre but est, pour chaque colonne x entre x_1 et x_2 , d'identifier la rangée y dans cette colonne qui est la plus proche du segment idéal et de tracer un pixel en (x, y) . [11]

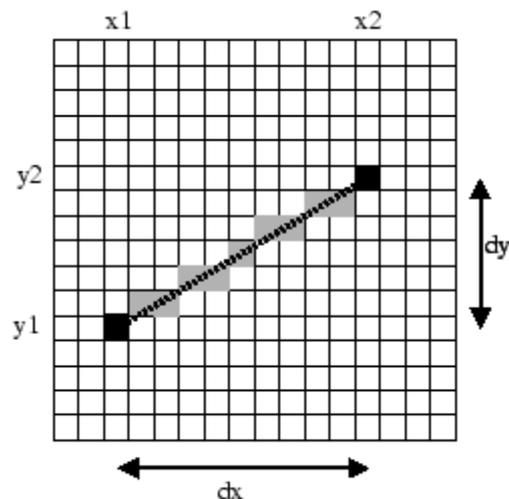


Figure III.6 : Un segment de droite échantillonné.

III.4.3.2. Détermination des ordonnées

La formule générale d'une droite entre les deux points est donnée par :

$$y = \frac{y_2 - y_1}{x_2 - x_1} \cdot (x - x_1) + y_1 \quad (1)$$

L'algorithme consiste à incrémenter l'abscisse de 1 et de choisir le pixel (valeur entière) le plus proche de la valeur idéal (valeur réelle), en arrondissant cette valeur vers l'entier le plus proche. L'équation précédente après incrément de x devient :

$$\hat{y} = \frac{y_2 - y_1}{x_2 - x_1} \cdot (x - x_1 + 1) + y_1 = \frac{y_2 - y_1}{x_2 - x_1} \cdot (x - x_1) + y_1 + \frac{y_2 - y_1}{x_2 - x_1} \quad (2)$$

$$\hat{y} = y + \frac{y_2 - y_1}{x_2 - x_1} \quad (3)$$

On commence en (x_1, y_1) , et que chaque fois que nous ajoutons 1 à l'abscisse, nous ajoutons la valeur constante $\frac{y_2 - y_1}{x_2 - x_1}$ à la valeur de l'ordonnée y du point de la droite correspondant. Cette valeur est la pente de la droite, et en vertu de notre hypothèse initiale, elle est comprise entre 0 et 1. Après l'arrondi, pour chaque colonne, nous utilisons donc soit la valeur précédente y (ordonnée du pixel d'abscisse $x-1$), soit cette valeur augmentée de 1. [11]

La procédure ressemble à ceci, en supposant que `tracerPixel(x, y)` est une primitive graphique traçant le pixel de rangée x et de colonne y ; exprimé en pseudo-code, l'algorithme de base est :

```

procédure tracerSegment(entier x1, entier y1, entier x2, entier y2) est
  déclarer entier x, y, dx, dy ;
  déclarer rationnel e, e(1,0), e(0,1) ; // valeur d'erreur et incréments
  dy ← y2 - y1 ;
  dx ← x2 - x1 ;
  y ← y1 ; // rangée initiale
  e ← 0,0 ; // valeur d'erreur initiale
  e(1,0) ← dy / dx ;
  e(0,1) ← -1.0 ;
  pour x variant de x1 jusqu'à x2 par incréments de 1 faire
    tracerPixel(x, y) ;
    si (e ← e + e(1,0)) ≥ 0,5 alors // erreur pour le pixel suivant de même rangée
      y ← y + 1 ; // choisir plutôt le pixel suivant dans la rangée supérieure
      e ← e + e(0,1) ; // ajuste l'erreur commise dans cette nouvelle rangée
    fin si ;
  fin pour ;
fin procédure ;

```

Figure III.7 : Algorithme de tracer de segment de droite. [11]

Cet algorithme est conçu pour générer des segments de droite dans le premier octant. Cependant il peut être généralisé pour tracer des segments sur les autres octants.

III.5. LES MOTEURS PAS-À-PAS

III.5.1. Définition

Le moteur pas à pas constitue un convertisseur électromécanique destiné à transformer le signal électrique (impulsion ou train d'impulsions de pilotage) en déplacement (angulaire ou linéaire) mécanique.

Le fonctionnement d'un moteur pas à pas nécessite la présence des éléments suivants :

- Une unité de commande (microprocesseur par exemple) qui fournit des impulsions dont la fréquence est proportionnelle à la vitesse de rotation du moteur, elle imposera également le sens de rotation.
- Un séquenceur, qui, à partir d'une alimentation, fournit l'énergie vers les enroulements appropriés du moteur. .
- Une alimentation de puissance.

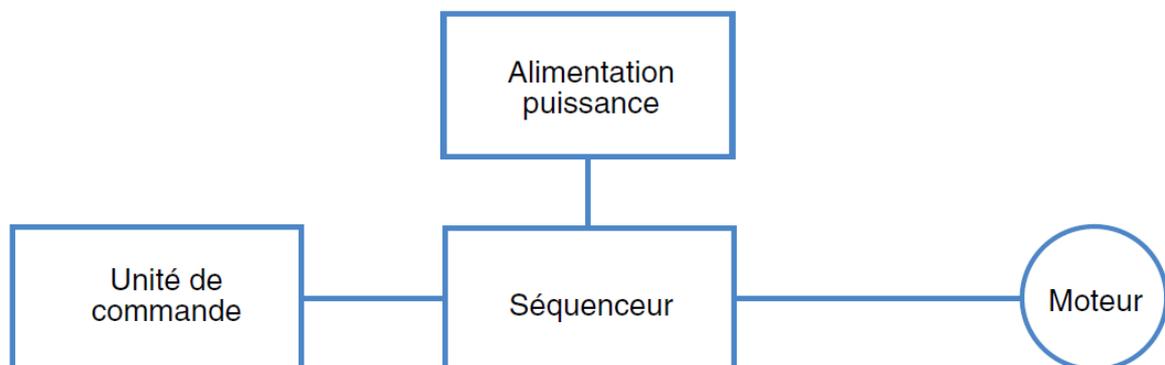


Figure III.8 : Eléments de la commande des moteurs pas à pas.

L'alimentation de chaque bobinage du moteur par une tension particulière provoque l'apparition d'un courant qui engendre un champ magnétique de direction précise. Le changement séquentiel des tensions particulières à chaque bobinage permet de déplacer la position du champ statorique selon une résolution élémentaire appelée pas. Autrement dit, toutes configurations des tensions aux bornes des bobinages correspondent à un déplacement de la position stable du rotor. Une série bien déterminée de commutation de tensions entraîne un déplacement sur un nombre correspondant de pas. La succession des configurations d'alimentation, à une fréquence donnée, impose un champ statorique tournant avec une résolution d'un micro pas, d'un demi pas ou d'un pas entier.

Le mouvement discret du champ d'entrefer est suivi par le rotor soumis au couple synchronisant. La nature du couple découle

- soit d'interaction champ (courant) du stator - fer du rotor, lorsque ce dernier présente une structure dentée passive ;
- soit d'interaction champ (courant) du stator - champ (aimant) du rotor, lorsque le rotor comporte un aimant ;
- soit les deux interactions précédentes simultanément. [12]

III.5.2. Les différents types des moteurs pas à pas

Les moteurs pas à pas peuvent être classés selon :

- L'alimentation des bobinages
 - Bipolaire
 - unipolaire
- La conception des moteurs
 - Moteur à aimant permanent
 - Moteur à réluctance variable
 - Moteur hybride combinant l'aimant et la réluctance variable

III.5.2.1. Le moteur bipolaire

Lorsque chaque enroulement est accessible par ses 2 connexions, le sens du courant peut être inversé : le moteur est bipolaire.

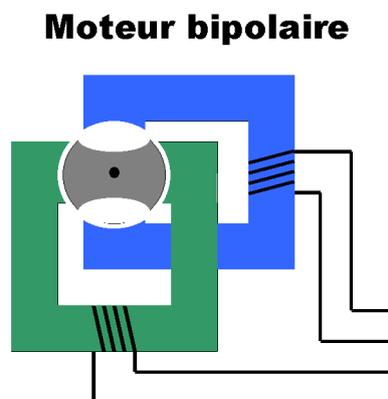


Figure III.9 : Schéma d'un moteur pas à pas bipolaire.

L'électronique de commande associée doit permettre la réversibilité du courant d'alimentation. La structure adaptée est celle du hacheur en pont double.

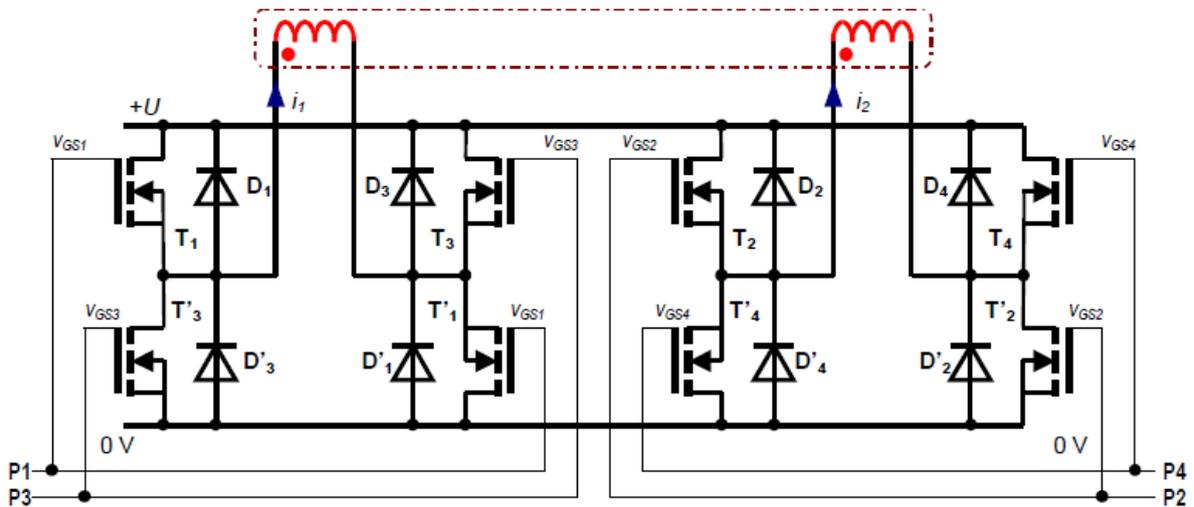


Figure III.10 : Circuit de commande d'un moteur pas à pas bipolaire. [13]

III.5.2.2. Le moteur unipolaire

Les phases peuvent être réalisées avec un seul enroulement scindé en deux. Pour optimiser les connexions, il est accessible par 3 fils (Figure 8). La borne commune impose le sens de parcours unique pour le courant.

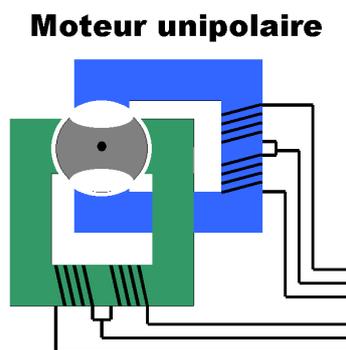


Figure III.11 : Schéma d'un moteur pas à pas unipolaire.

Il s'agit alors d'un moteur unipolaire. Il faut donc l'associer à une électronique de commande unidirectionnelle des courants. Pour cela, on utilise une structure de hacheur série.

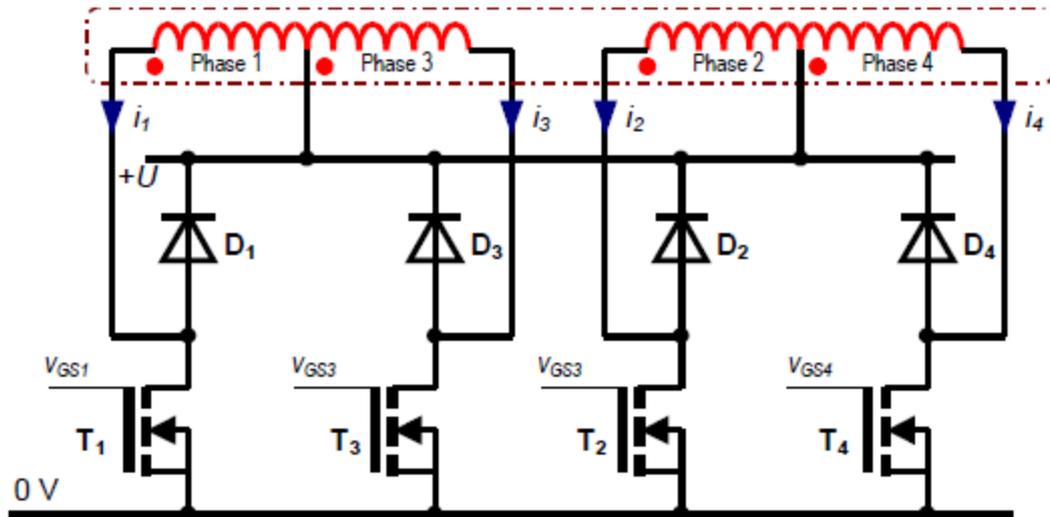


Figure III.12 : Circuit de commande d'un moteur pas à pas unipolaire. [13]

III.5.2.3. Moteur à réluctance variable

Un moteur à reluctance variable possède un rotor en fer doux non magnétisé avec moins de pôles que le stator. Le rotor va se placer de telle sorte que le flux magnétique qui le traverse soit maximum (il recherche la reluctance la plus faible possible).

Ce type de moteur est caractérisé par une structure dentée au niveau du rotor et du stator.

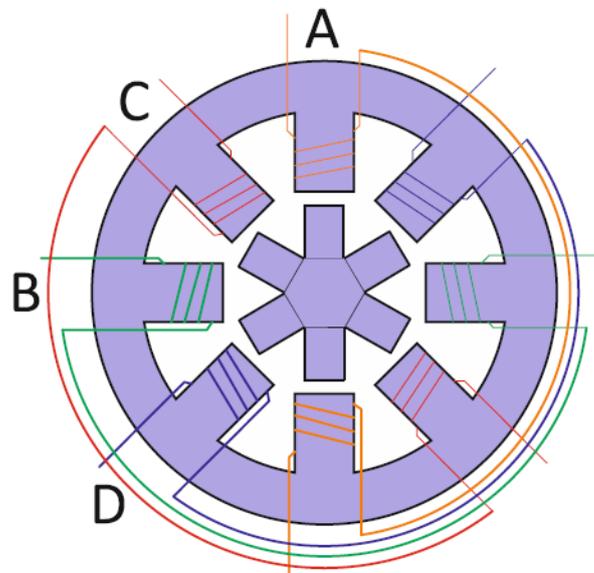


Figure III.13 : Schéma d'un moteur pas à pas à réluctance variable. [12]

III.5.2.4. Moteur à aimant permanent

C'est le modèle dont le fonctionnement est le plus simple. Le moteur est constitué d'un rotor à aimant permanent et un stator supportant les bobinages. À cause des aimants permanents, le rotor reste freiné à sa dernière position lorsque le bloc d'alimentation cesse de fournir des impulsions.

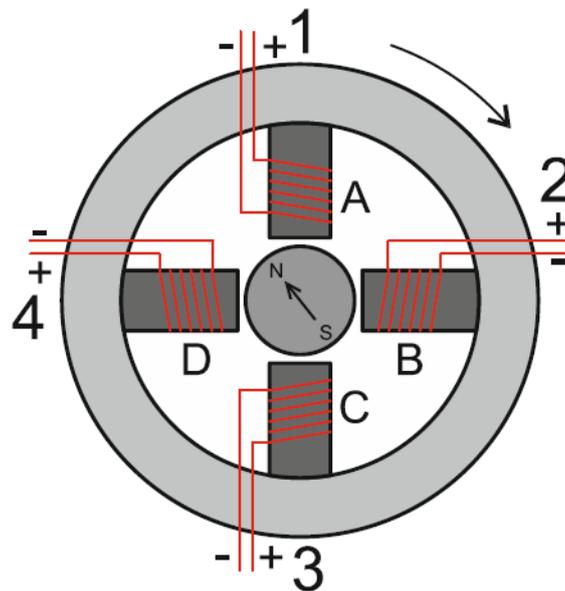


Figure III.14 : Schéma d'un moteur pas à pas à aimant permanent. [12]

III.5.2.5. Moteur hybride

Le moteur pas à pas hybride emprunte du moteur à aimant permanent et de la machine à réluctance variable. Il est donc à réluctance variable mais avec un rotor à aimants permanents.

Le rotor présente plusieurs dents comme pour un moteur pas à pas à réluctance variable, mais chaque dent est polarisée comme pour un moteur pas à pas à aimants permanents. Physiquement le rotor est composé de deux éléments identiques à un rotor de moteur à réluctance variable (rouge et bleu ici), reliés ensemble par un aimant permanent (noir), avec un déphasage d'une 1/2 dent. De ce fait ces deux éléments ont une polarisation différente (nord et sud) et vont réagir à la polarisation de chacune des dents du stator.

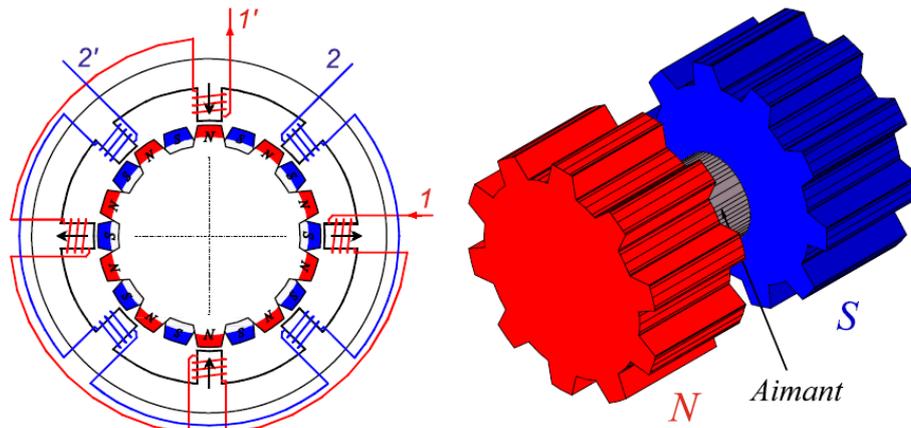


Figure III.15 : Schéma d'un moteur pas à pas hybride. [12]

III.5.3. MODE DE COMMANDE DES MOTEURS PAS À PAS

La rotation du moteur s'effectue par une séquence d'alimentation des divers enroulements de phases en unipolaire ou en bipolaire. La génération d'un couple est intimement liée aux courants circulant dans les phases. Le contrôle du moteur peut être optimisé en alimentant deux phases adjacentes afin d'augmenter le couple délivré par la machine ou diminuer le pas angulaire pour améliorer la résolution.

Pour généraliser la description, on se réfère aux commandes des commutateurs.

III.5.3.1. Alimentation d'une seule phase en pas entier (Mode1)

Une seule phase est alimentée à la fois par injection d'un courant nominal I_n . C'est dans ce mode que le fabricant définit le pas angulaire du moteur.

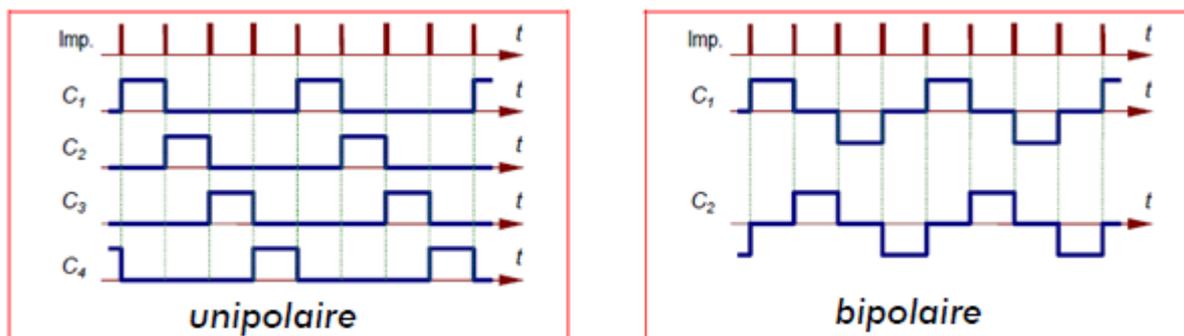


Figure III.16 : Signaux de commande en pas entier, alimentation d'une seule phase. [13]

III.5.3.2. Alimentation de deux phases en pas entier (Mode2)

Deux phases du moteur sont alimentées simultanément par un courant nominal I_n . Le couple moteur est multiplié par 2 par rapport au contrôle selon le mode 1. Le pas angulaire est identique au mode 1 mais les positions d'équilibre sont intermédiaires aux précédentes.

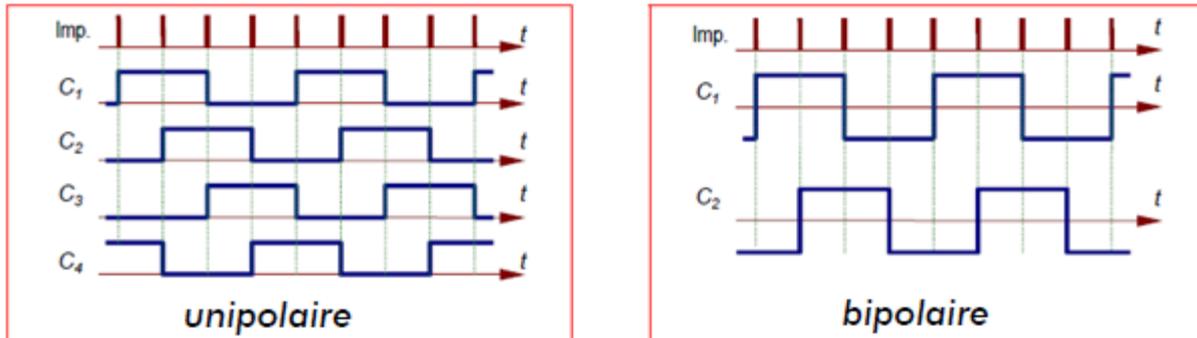


Figure III.17 : Signaux de commande en pas entier, alimentation de deux phases. [13]

III.5.3.3. Alimentation d'une ou deux phases en demi-pas (Mode3)

La combinaison en alternance du mode 1 et du mode 2 permet le fonctionnement en demi-pas. Pour effectuer un déplacement donné, il faut doubler le nombre d'impulsions de commande. Les couples ne sont pas identiques selon si une ou les deux phases sont alimentées.

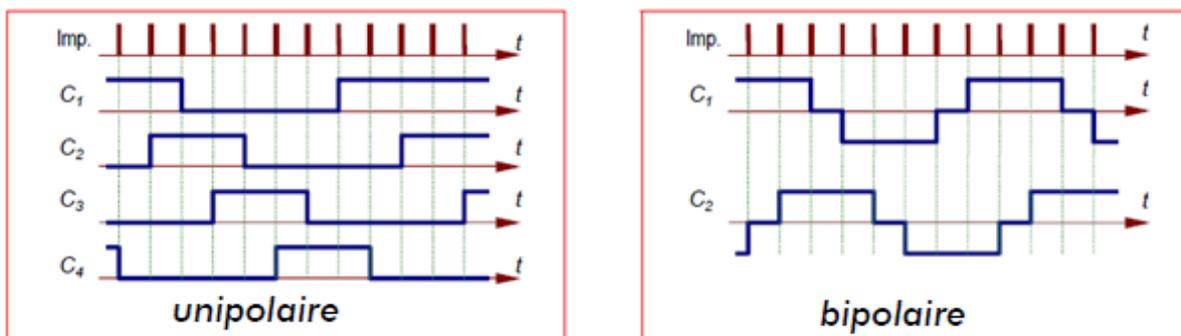


Figure III.18 : Signaux de commande en demi-pas, alimentation d'une ou deux phases. [13]

III.5.3.4. Commande optimisé en demi-pas (Mode4)

Pour un fonctionnement régulier en demi-pas, il faut augmenter le courant d'un facteur 2 lorsqu'une seule phase est alimentée. On a ainsi les performances de couple du mode 2 mais avec deux fois plus de pas.

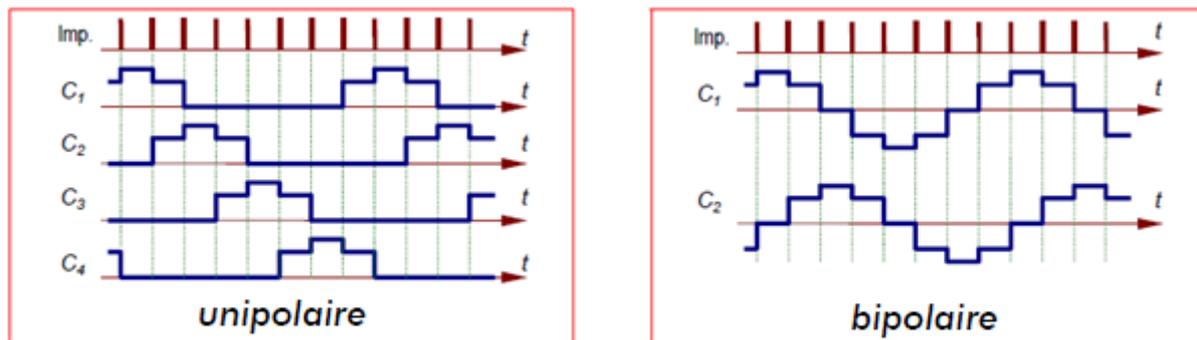


Figure III.19 : Signaux de commande en demi-pas optimisé. [13]

III.5.3.5. Commande avancée en micro-pas (Mode5)

Ce mode consiste à multiplier les positions intermédiaires en asservissant le courant de chaque phase. Il constitue une généralisation du mode 4. Par découpage à fréquence plus élevée, le courant dans chaque phase suit une loi sinusoïdale pour se rapprocher des machines synchrones de puissance.

III.6. La carte de commande ARDUINO

C'est un circuit imprimé comportant tous les composants électroniques nécessaires pour faire fonctionner un microcontrôleur associé à une interface USB lui permettant de communiquer avec un ordinateur.

Un microcontrôleur est l'équivalent d'un petit ordinateur. Il contient un microprocesseur (unité centrale) associé à différentes mémoires et des interfaces lui permettant de communiquer avec l'extérieur, et bien sûr une horloge pour cadencer l'ensemble. L'alimentation électrique peut être fournie par le port USB ou une alimentation externe.

Le microcontrôleur exécute en permanence un programme qui a été copié dans sa mémoire flash depuis un ordinateur via le port USB (interface série). Ce programme est initialisé à la mise sous tension de la carte ARDUINO ou lorsqu'on actionne le bouton poussoir "reset" sur la carte.

III.6.1. ARDUINO MEGA 2560

L'ARDUINO MEGA 2560 est un module basé sur le microcontrôleur ATmega 2560. Il dispose de 54 broches entrées / sorties numériques (dont 14 peuvent être utilisées comme

sorties PWM), 16 entrées analogiques, 4 UART (ports série matériels), un oscillateur à quartz de 16 MHz, une connexion USB, une embase d'alimentation, un connecteur ICSP et un bouton de réinitialisation. Il contient tous les éléments nécessaires pour le fonctionnement du microcontrôleur.

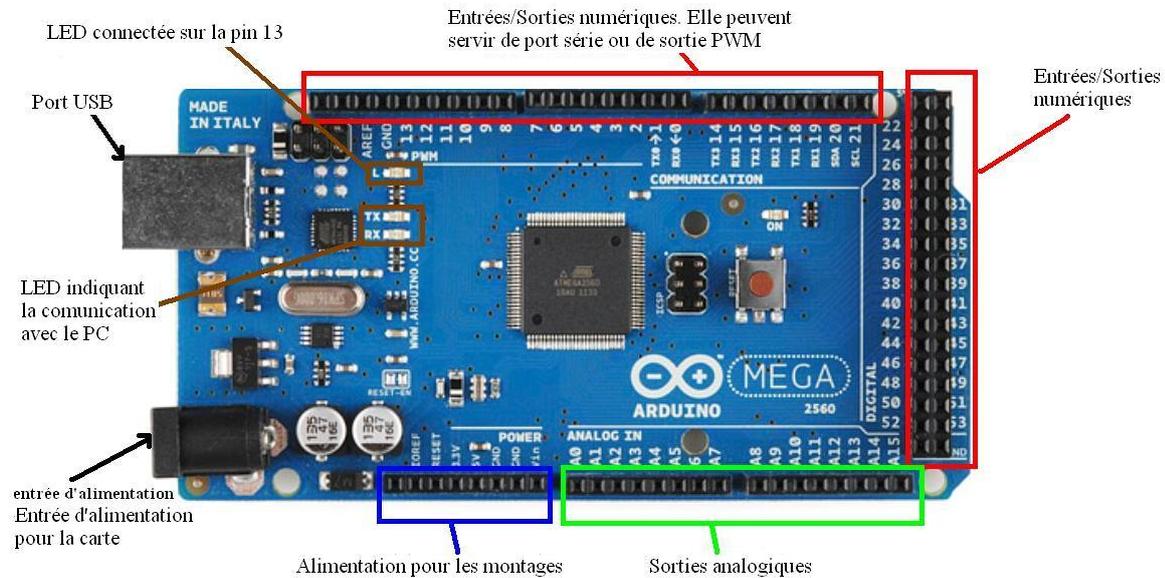


Figure III.20 : ARDUINO MEGA 2560.

Le tableau ci-dessous résume les caractéristiques de l'ARDUINO MEGA 2560.

Tension de fonctionnement	5V
Tension d'alimentation externe (recommandée)	7-12V
Tension d'alimentation externe (limites)	6-20V
Pins d'entrées/sorties	54 (dont 15 peuvent générer PWM)
Pins d'entrées analogiques	16
Courant continu par pin d'E/S	20 mA
Courant continu pour pin 3.3V	50 mA
Mémoire flash	256 KB dont 8 KB utilisée par le bootloader
SRAM	8 KB

EEPROM	4 KB
Fréquence d'horloge	16 MHz
LED_BUILTIN	13
Longueur	101.52 mm
largeur	53.3 mm
poids	37 g

Tableau III.2 : Caractéristiques de l'ARDUINO MEGA 2560. [14]

III.6.2. Logiciel ARDUINO

Comme n'importe quel langage de programmation, une interface souple et simple est exécutable sur n'importe quel système d'exploitation.

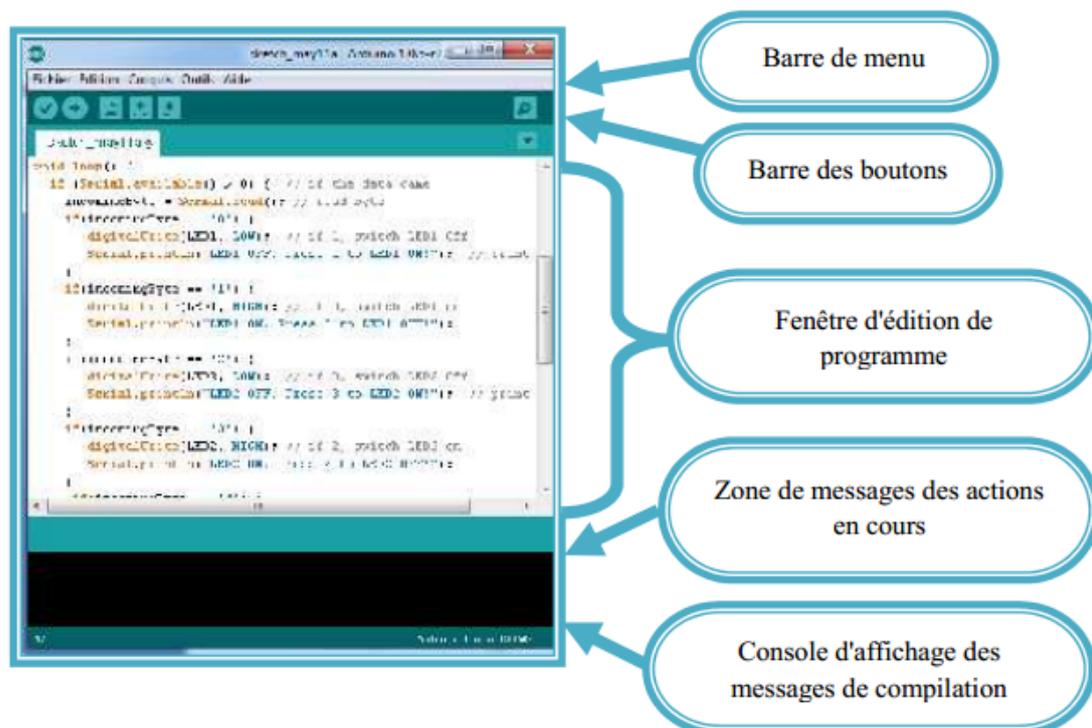


Figure III.21 : Interface du logiciel ARDUINO.

Il comporte une console de communication série pour l'échange de lignes de commandes avec la carte.

Avant d'envoyer un programme dans la carte, il est nécessaire de sélectionner le type de la carte (ARDUINO MEGA 2560) et le numéro de port USB dans le menu Outils.

III.6.3. Programmation sous ARDUINO

Un programme ARDUINO, basé sur la programmation en C, est une suite d'instructions élémentaires sous forme textuelle (ligne par ligne). La carte lit puis effectue les instructions les unes après les autres dans l'ordre défini par les lignes de codes.

Le programme ARDUINO comporte trois parties :

1. la partie déclaration des variables (optionnelle)
2. la partie initialisation et configuration des entrées/sorties : la fonction **setup ()**
3. la partie principale qui s'exécute en boucle : la fonction **loop ()**

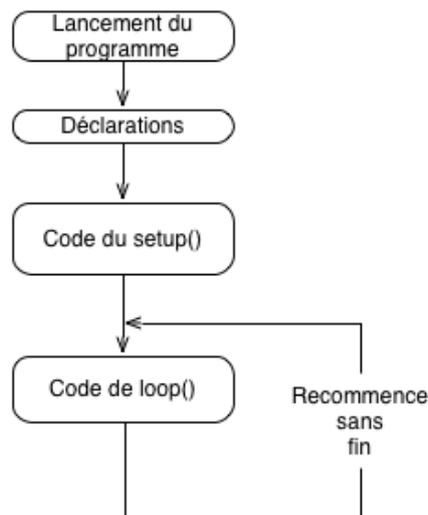


Figure III.22 : Structure de base d'un programme ARDUINO.

III.7. Circuit d'interface

Le circuit d'interface permet de séparer le circuit de commande (Arduino) de celui de puissance. Pour cela on utilise deux composants électroniques :

- Optocoupleur : utilisé pour protéger les entrées de l'ARDUINO contre les tensions et courants élevés et de supprimer les bruits sur les signaux d'entrées.
- Buffer : utilisé pour protéger les sorties de l'ARDUINO. Le buffer laisse passer les signaux dans un sens et les bloquer dans l'autre sens.

Le circuit est conçu pour supporter 8 entrées et 12 sorties définies comme suit :

➤ Entrées

- Bouton d'arrêt
- Bouton pause
- Bouton reprendre
- Contacte de fermeture des portes de sécurité (non utilisée)
- Contacte de détection de matériaux au touché (non utilisée)
- Capteur fin de course x
- Capteur fin de course y
- Capteur fin de course z

➤ Sorties

- Impulsion de pas axe_x
- Impulsion de pas axe_y
- Impulsion de pas axe_z
- direction de pas axe_x
- direction de pas axe_y
- direction de pas axe_z
- activer les drivers moteurs pas à pas
- activer la broche
- sens de rotation de l'outil (non utilisée)
- PWM pour la broche (non utilisée)
- Commande pompe à air (non utilisée)
- Commande pompe à liquide (non utilisée)

Les E/S marquées avec « (non utilisée) » ont été intégrées pour une éventuelle extension des fonctionnalités de la machine, en modifiant le programme, sans toucher au matériel.

L'image ci-dessous montre les circuits élémentaires utilisés pour les entrées sorties :

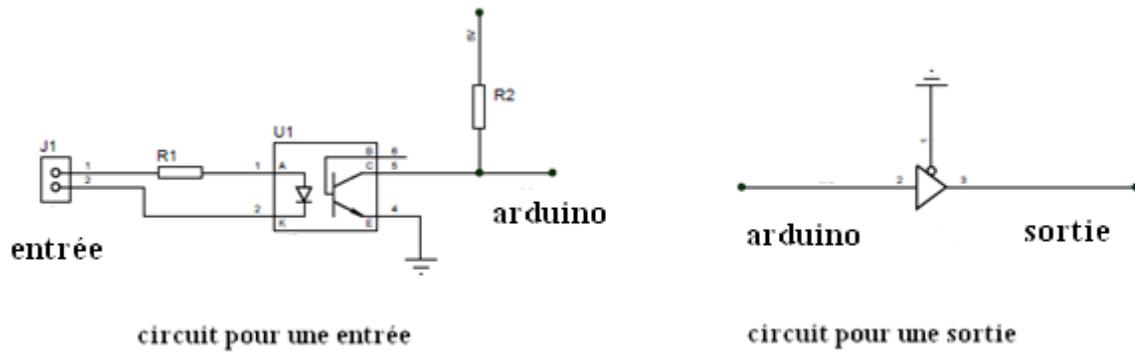


Figure III.23 : Circuit de base pour interface entrées sorties.

La dernière étape consiste à créer le circuit imprimé « PCB ». Ce circuit sera utilisé pour la réalisation de la carte.

III.8. Conclusion

Dans ce chapitre nous avons donné une présentation générale sur la commande numérique des machines-outils ainsi que la structure du programme qui sert à interpréter les commandes G-code et interpoler les parcours outil. Nous avons aussi étudié de manière générale, le moteur pas à pas, l'organe principal commandé et présenter le microcontrôleur utilisé ainsi que le circuit permettant d'interface ce microcontrôleur et les éléments externes.

Cette partie est nécessaire pour l'étape suivante qui consiste à réaliser l'ensemble de la machine et sa commande.

Chapitre IV :

***REALISATION
EXPERIMENTALE DE LA
MACHINE CNC***

IV.1. Introduction

Dans ce chapitre nous allons présenter une description réelle de la machine réalisée et finir avec un test d'usinage avec la machine.

IV.2. Eléments et caractéristiques de la partie commande

IV.2.1. Circuit d'interface

IV.2.1.1. Réalisation virtuelle sous « PROTEUS »

Avant de passer à la réalisation pratique, nous avons utilisé un CAO: il s'agit de « ISIS PROTEUS », c'est un CAO électronique conçu par « Labcenter Electronics » qui permet de dessiner des schémas électroniques, de les simuler et de réaliser le circuit imprimé correspondant. Il comporte de nombreux outils regroupés en modules au sein d'une interface unique.

L'image ci-dessous montre le circuit réalisé sous PROTEUS :

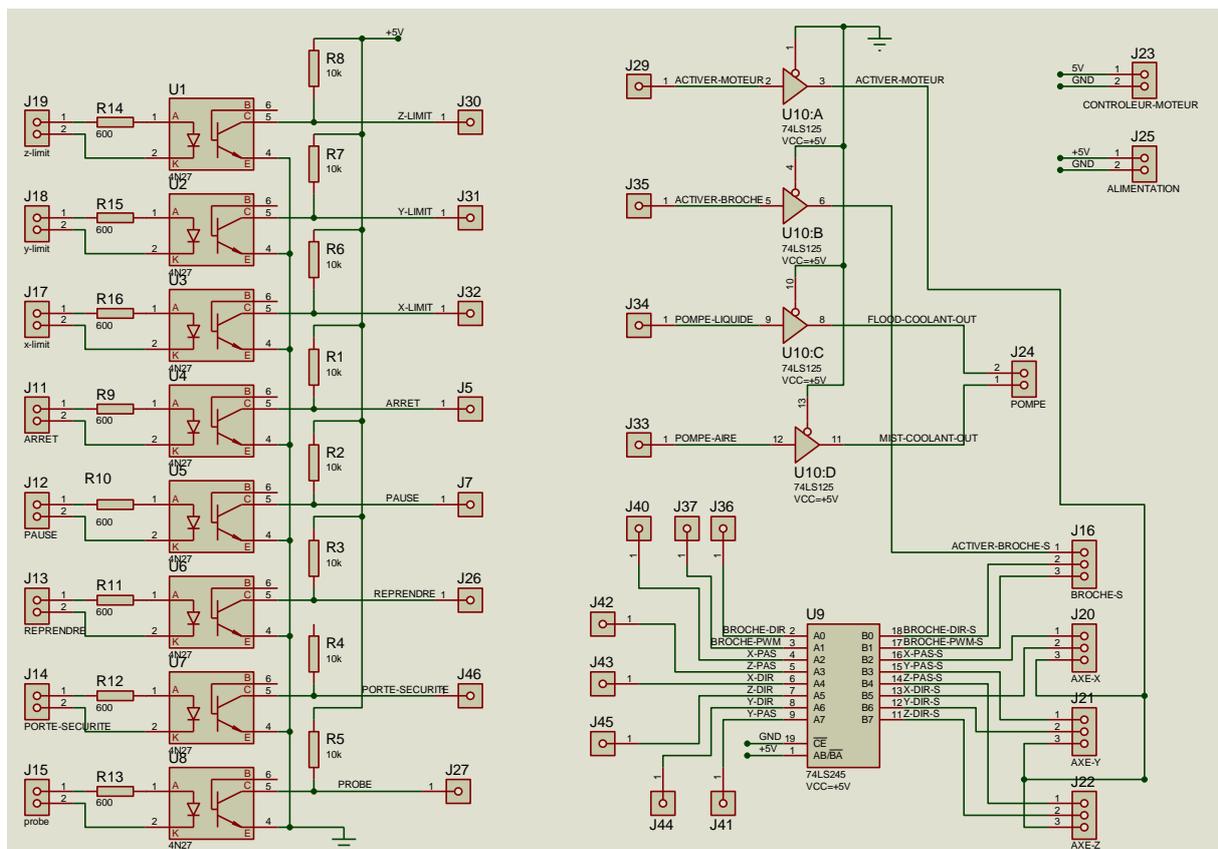


Figure IV.1 : Schéma du circuit d'interface.

Le tableau suivant regroupe les composants utilisés dans ce circuit.

Composant	Symbole	Valeur	Référence	Quantité
Résistance	R9...R16	600 Ohms		8
	R1...R8	10 K Ohms		8
Optocoupleur	U1...U8		4N27	8
Buffer	U9		74LS245	1
	U10		24LS125	1
Connecteur (2 pins)	J11...J19, J23, J25			10
Connecteur (3 pins)	J16, J20...J22			4

Tableau IV.1 : Liste des composants utilisés dans le circuit d’interface.

L’étape suivante consiste à créer le circuit imprimé « PCB » avec « PROTEUS ARES ». Ce circuit sera utilisé pour la réalisation pratique de la carte, en imprimant uniquement les pistes cuivrées. Les schémas obtenus sont présentés sur les deux figures suivantes :

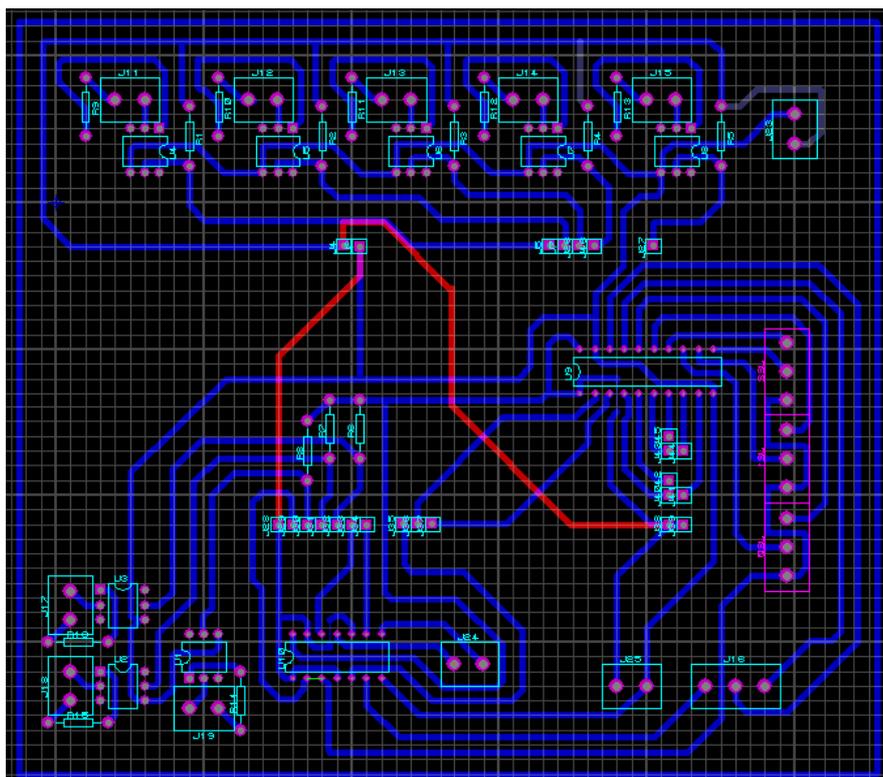


Figure IV.2 : Schéma PCB du circuit d’interface.

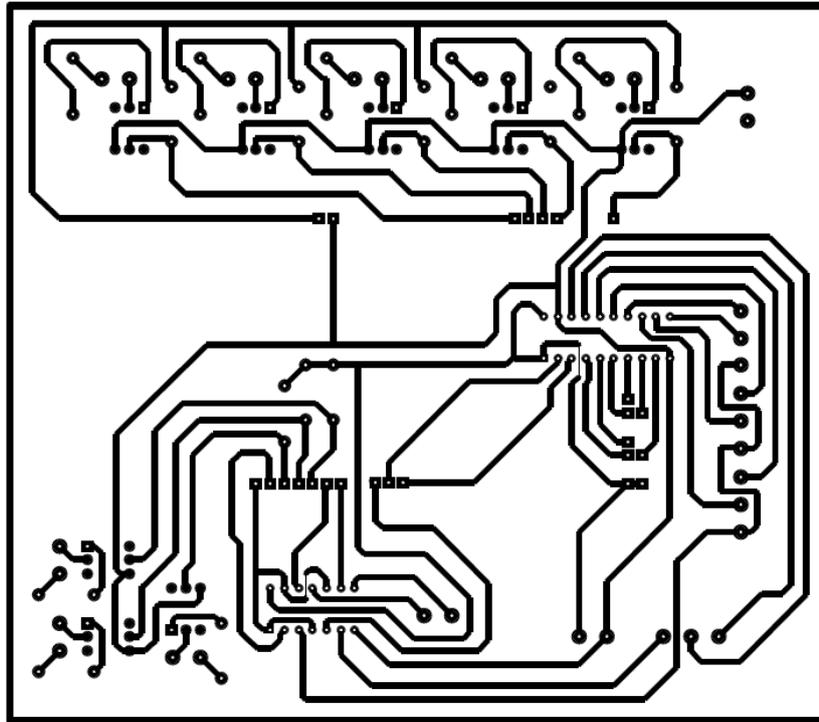


Figure IV.3 : Schéma des pistes cuivrées à imprimer.

IV.2.1.2. Réalisation pratique du circuit :

Nous avons utilisé une méthode simple sans avoir besoin du matériel et produits coûteux spécialisés pour circuits imprimés (l'insoleuse et le perchlore de fer).

IV.2.1.2.1. Impression du circuit sur la platine cuivrée

Pour le faire on suit les étapes suivantes :

- Enlever la résine photosensible à l'aide d'un chiffon et d'un peu d'acétone (diluant).
- Imprimer le circuit (figure (IV.3)) sur un papier photo avec une imprimante laser.
- Placer la feuille imprimée contre la platine cuivrée et appliquer au fer à repasser sur le circuit. Le temps dépend de la puissance du fer et du type de toner (encre).
- Plonger le circuit dans un récipient contenant de l'eau. laisser détrempé quelques minutes, puis peler délicatement le papier.
- Corriger avec un marqueur permanent les pistes ratées.

IV.2.1.2.2. Gravure du circuit

La gravure consiste à ronger le cuivre aux endroits non protégés par l'encre, il existe pour ce faire plusieurs méthodes utilisant des produits différents.

La méthode utilisée dans notre cas demande des précautions d'usages, car elle emploie des matières dangereuses. Afin de limiter les risques, un équipement minimum est nécessaire pour se protéger :

- ✓ Gants résistants aux produits chimiques.
- ✓ Lunettes de protection.
- ✓ Masque de protection.

De préférence faire les manipulations en milieu ouvert et près d'une source d'eau.

Les produits chimiques à utiliser :

- ✓ Acide chlorhydrique 33% (esprit de sel)
- ✓ Eau oxygénée (peroxyde d'hydrogène) 10 Vol

On mélange un volume d'acide chlorhydrique et cinq volumes d'eau oxygénée dans un récipient en plastique ou en verre. Dès qu'on plonge notre plaque dans la solution, il se produira un intense bouillonnement avec dégagement de chaleur, et le cuivre sera entièrement rongé en l'espace de quelques minutes (environ 5 minutes).

La figure ci-dessous rassemble les produits chimiques utilisés pour réaliser la carte :



Figure IV.4 : Ensemble des produits chimiques utilisés.

IV.2.1.2.3. Perçage et soudage

La dernière étape consiste à percer les pastilles (emplacements correspondant aux pattes de composants), placer les composants et les souder avec de l'étain et du fer à souder. Le résultat final est donné par la figure suivante :

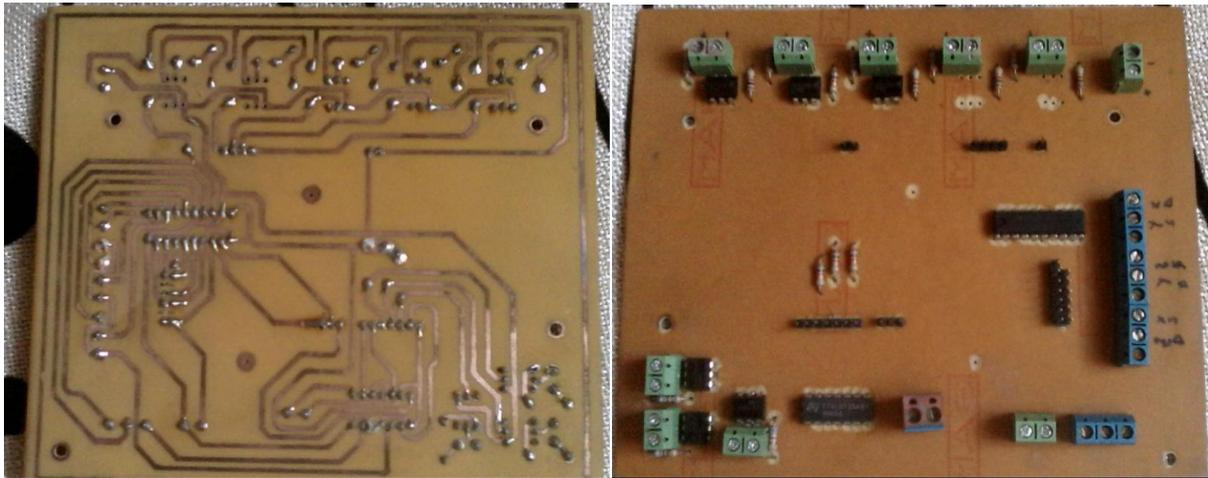


Figure IV.5 : Circuit d'interface réalisé.

IV.2.2. Contrôleur du moteur pas à pas (driver) :

Le contrôleur est un circuit spécialisé nécessaires pour transmettre de la puissance au moteur pas à pas. Le contrôleur qu'on a utilisé (DM882ES), présenté sur la figure (IV.6), est équipé de 2 connecteurs débloqués : le connecteur pour les signaux de contrôle et le connecteur pour l'alimentation et le raccordement du moteur. Le tableau (IV.2) donne une description des différentes fins du contrôleur.

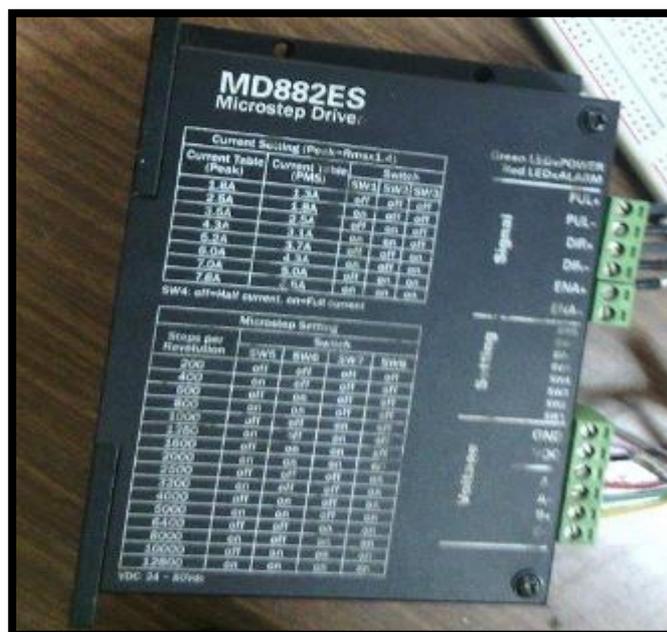


Figure IV.6 : Contrôleur moteur pas à pas.

Broche	Informations sur la broche
PUL+ PUL-	Une impulsion montante sur cette entrée fait avancer le moteur d'un pas. La tension de l'impulsion doit entrer de 4,5 à 5 V pour un état HAUT et 0 à 0,5 V pour un état BAS. La largeur d'impulsion doit être de minimum 2,5 μ s pour un fonctionnement correct. Une résistance en série doit être ajoutée pour limiter le courant en 12 V (1 k Ω) et en 24 V (2 k Ω).
DIR + DIR -	Ce signal représente les deux directions de rotation du moteur. Il doit être appliqué au minimum 5 μ s avant le signal PUL. La tension du signal DIR doit être de 4,5 à 5 V pour un état HAUT et 0 à 0,5 V pour un état BAS. La rotation dépend également du câblage du moteur.
ENA+ ENA-	Ce signal est utilisé pour permettre ou interdire l'utilisation du contrôleur. Un signal haut permet de l'utiliser tandis qu'un signal bas le bloque. Ces broches sont habituellement laissées déconnectées.
+Vcc	Tension d'alimentation : 20 à 40 Vcc
GND	Masse
A+ A-	Phase A du moteur
B+ B-	Phase B du moteur

Tableau IV.2 : description des pins du contrôleur.

IV.2.3. L'armoire de commande

L'armoire de commande rassemble tous les éléments électriques de la machine. Elle est représentée par la figure suivante :

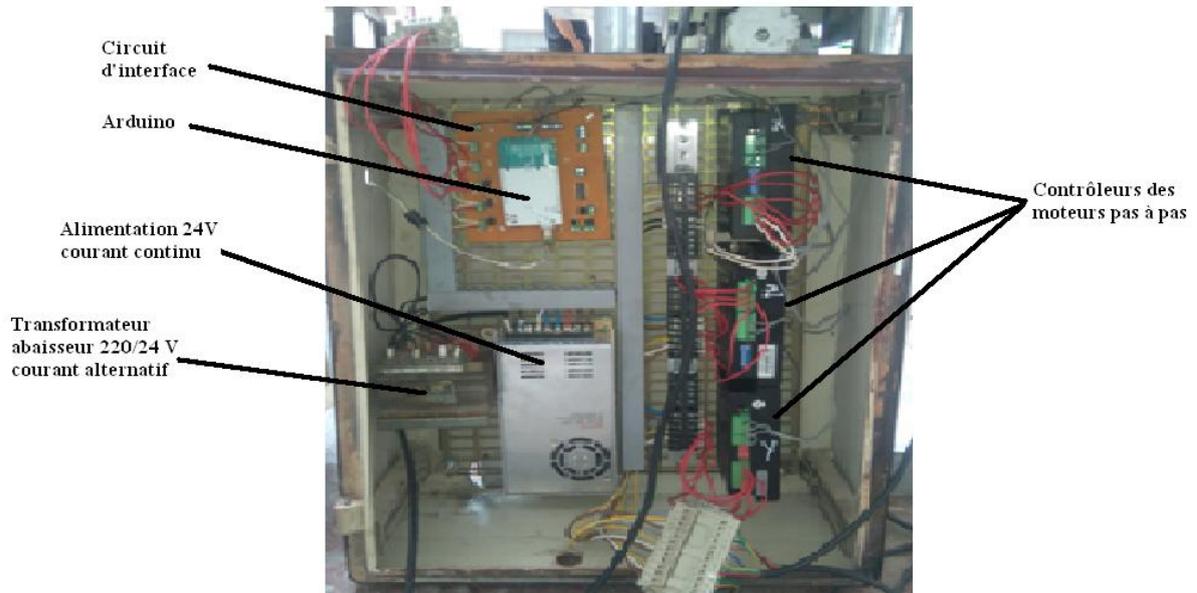


Figure IV.7 : armoire de commande.

IV.3. Eléments et caractéristiques de la partie opérative de la machine

IV.3.1. Le bâti

Le bâti doit être suffisamment rigide pour assurer une bonne stabilité de la machine.

Poids : 190kg.



Figure IV.8 : Le bâti.

IV.3.2. Caractéristiques de la broche

- ✓ Puissance de la broche : 1.6KW ;
- ✓ Vitesse de rotation : 300 tr/mn ;
- ✓ Fixation manuelle de l'outil ;
- ✓ Diamètre maximale de perçage 30 mm.
- ✓ Tension/Fréquence d'alimentation : 220V/50Hz.



Figure IV.9 : La broche.

IV.3.3. Les axes X, Y

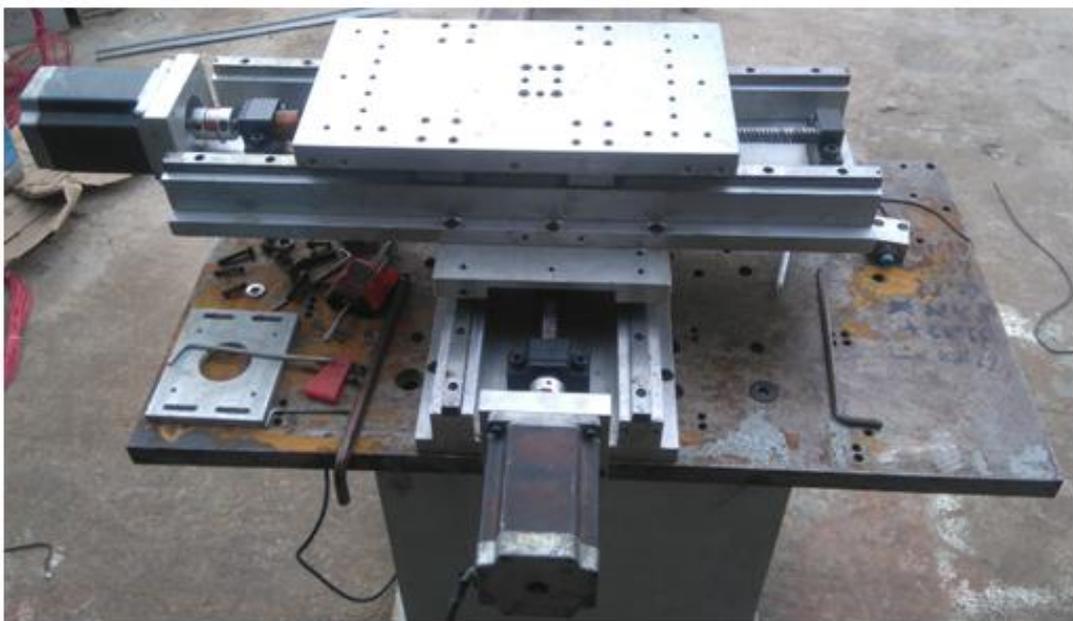


Figure IV.10 : Les axes X, Y.

- Le moteur pas à pas
 - ✓ Type : hybride
 - ✓ Courant nominal : 4.2 A/phase
 - ✓ Pas angulaire : 1.8°
 - ✓ Couple nominal : 8 N.m
- Espace de travail
 - ✓ Axe longitudinal : X 200 mm Maxi
 - ✓ Axe transversal : Y 300 mm Maxi
 - ✓ Axe vertical : Z 100 mm Maxi

V.2. Test d'un simple usinage

Pour tester notre machine, on a usiné une forme simple, générée par le logiciel WinUnisoft, sur un bloc d'aluminium. Le résultat de simulation virtuelle est donné par la figure suivante :

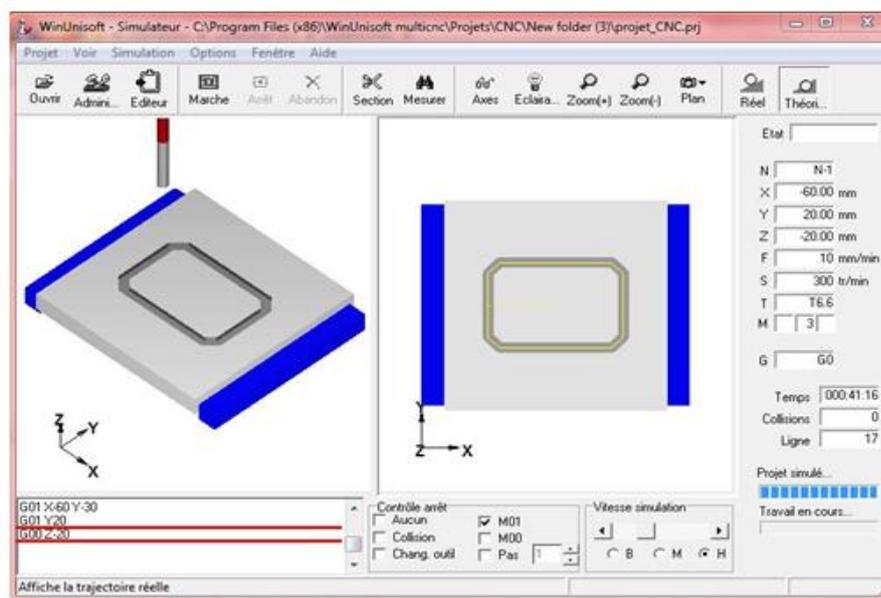


Figure IV.11 : Simulation d'usinage avec WinUnisoft.

Le résultat d'usinage réel de la pièce est donné par la figure ci-dessous :



Figure IV.12 : Résultat d'usinage.

IV.4. Conclusion

Dans ce chapitre la réalisation de la machine-outil et sa commande a été faite. Nous avons détaillé la procédure de réalisation du circuit d'interface et expliqué l'utilisation du contrôleur de moteur pas à pas (driver).

Nous avons au final réussi à assembler tous les éléments, dont les images ont été présentées, de la machine-outil et d'intégrer sa commande numérique. Le résultat de l'usinage est satisfaisant.

Conclusion générale

L'objectif de ce travail est la « conception et réalisation d'une machine-outil à commande numérique 3 axes (CNC, 3D) ».

Pour bien mener ce travail nous l'avons présenté en quatre parties :

Dans le premier chapitre nous avons présenté quelques généralités sur la machine-outil à commande numérique.

Dans le deuxième chapitre une pré-réalisation sous le logiciel SolidWorks a été faite avec explication de l'utilité de chaque élément.

Le troisième chapitre est consacré à la présentation de la commande ainsi que les outils nécessaires pour implémenter cette commande.

Dans le quatrième chapitre la réalisation finale de la machine a été faite. Les résultats sont satisfaisants.

Ce modeste travail nous met en mesure d'appréhender le monde de l'ingénierie et d'en comprendre les enjeux et les techniques.

Nous espérons avoir contribué à motiver les étudiants sur un domaine de connaissance passionnant et que notre travail pourra servir dans les futurs projets de fin d'étude.

Comme perspectives de ce travail on cite :

- Amélioration des algorithmes d'interpolation ;
- Intégrer une méthode de calcul des accélérations pour avoir des mouvements uniformes ;
- Optimisation du code source du programme en termes de temps d'exécution (réduire le temps de calcul) ;

Références bibliographiques

- [1] Farid ASMA ‘Introduction à la commande numérique’, Notes de cours, 2007/2008.
- [2] Hechmi CHERMITI ‘ Généralités sur les machines-outils à commande numérique’, ISET Kairouan.
- [3] <https://fr.wikipedia.org/wiki/Usinage>, 5 juin 2017.
- [4] http://linuxcnc.org/docs/html/gcode/overview_fr.html, 28 juin 2017.
- [5] Thomas R. Kramer, Frederick M. Proctor, Elena Messina ‘The NIST RS274NGC Interpreter - Version 3’ NISTIR 6556, 17 août 2000.
- [6] http://linuxcnc.org/docs/html/gcode/gcode_fr.html, 28 juin 2017
- [7] http://linuxcnc.org/docs/html/gcode/m-code_fr.html, 28 juin 2017
- [9] Raphael Laguionie ‘programmation avancée des machines-outils : implémentation de l’usinage trochoïdal et du tréfilage dans la chaîne numérique step-nc’, Thèse de Master, Ecole Centrale de Nantes et l’Université de Nantes, soutenue le 01/10/2007.
- [10] https://fr.wikipedia.org/wiki/Interpolation_num%C3%A9rique, 28 juin 2017.
- [11] https://fr.wikipedia.org/wiki/Algorithme_de_trac%C3%A9_de_segment_de_Bresenham, 28 juin 2017.
- [12] M. Correvon ‘Systèmes électromécaniques’, Notes de cours, Haute Ecole d’Ingénierie et de Gestion Du Canton du Vaud.
- [13] <https://fr.scribd.com/document/186717394/mpp1>, 28 juin 2017.
- [14] <https://store.arduino.cc/arduino-mega-2560-rev3>, 28 juin 2017.