

الجمهورية الجزائرية الديمقراطية الشعبية

République algérienne démocratique et populaire

وزارة التعليم العالي والبحث العلمي

Ministère de l'enseignement supérieur et de la recherche scientifique

Université Badji Mokhtar Annaba

جامعة باجي مختار عنابة



Faculté : Science de l'Ingéniorat

Département : Electronique

Spécialité : Electronique des Systèmes Embarqués

Mémoire

Présenté pour l'obtention du diplôme de : Master

Thème :

Réalisation & manipulation d'une machine à commande numérique à base d'un microcontrôleur.

Présenté par : Madi Fateh

Encadré par : Benouaret Mohamed Professeur

Devant le jury :

Président : Azzouz_Mokhnache MCB

Examineur : Saliha_Hariz MCA

Examineur : F. Amara MCB

Année universitaire : 2018/2019

Résume :

Dans ce rapport, nous donnons un aperçu de la machine CNC (sa définition, son historique, sa structure, ses types et enfin certains produits de cette machine).

Dans le deuxième chapitre, nous passons à la phase de conception et l'assemblage de la machine.

Le troisième chapitre est consacré à la définition des équipements électroniques utilisés, y compris la carte arduino, les moteurs pas à pas, les contrôleurs des moteurs, la source d'alimentation, etc.

Le chapitre 4 explique les logiciels utilisés dans la programmation de la machine : (ArduinoIDE) Il est utilisé pour écrire et télécharger des logiciels sur des cartes compatibles avec Arduino, (Universal G-Code Sender), utilisé pour exécuter et contrôler une machine CNC, Logiciel inskcape est un logiciel libre de dessin vectoriel, (Grbl Controller) est un programme conçu pour envoyer du G Code aux machines à commande numérique.

Dans le dernier chapitre, nous testons la machine et donnons le résultat obtenu.

ملخص:

يتحدث هذا التقرير عن آلة التحكم الرقمي بالحاسوب وينقسم إلى خمسة فصول، في الفصل الأول نعطي لمحة عامة عن الآلة (تعريفها، تاريخها، مبدأ عملها، أنواعها، وأخيراً بعض منتجات هذه الآلة).

في الفصل الثاني نمر إلى مرحلة التصميم وتجميع الآلة

أما الفصل الثالث فهو مخصص للتعريف بالمعدات الإلكترونية المستخدمة نذكر منها بطاقة الـ اردوينو، المحرك خطوة خطوة، موجه المحرك، مصدر الطاقة الخ.

الفصل الرابع فيه شرح للبرامج المستخدمة في برمجة الآلة وهي

(Arduino IDE) يتم استخدامه لكتابة وتنزيل البرامج على بطاقات متوافقة مع Arduino

(Universal G-Code Sender) يستخدم هذا البرنامج لتشغيل آلة CNC و التحكم فيها

و برنامج (inskcape) وهو عبارة عن برنامج تصميم (Grbl). هو برنامج مصمم لإرسال GCode إلى آلات CNC ، مثل آلات الطحن ثلاثية الأبعاد.

في الفصل الأخير نقوم بتجريب الآلة و إعطاء النتيجة المتحصل عليها.

Abstract :

In this report, we give an overview of the CNC machine (its definition, its history, its structure, its types and finally some products of this machine).

In the second chapter, we move to the design phase and assembly of the machine.

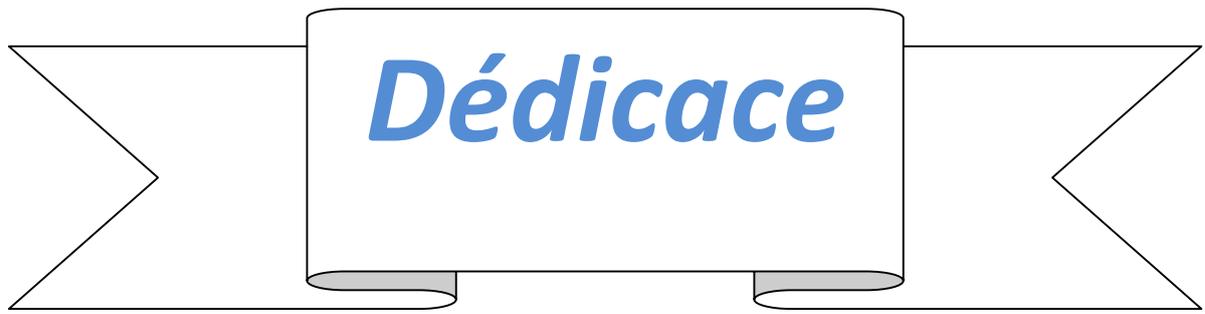
The third chapter is devoted to the definition of the electronic equipment used, including the arduino card, the stepper motors, the driver motor, the power supply, and so on.

Chapter 4 explains the software used in programming the machine:

(Arduino IDE) It is used to write and download software on cards compatible with

Arduino, (Universal G-Code Sender) used to run and control a CNC machine, Software inskcape is a free vector drawing software, (Grbl Controller) is a program designed to send GCode to CNC machines.

In the last chapter, we test the machine and give the result.



Je dédie ce travail à :

- ❖ Mes très chers parents pour leurs sacrifices et leurs encouragements durant toutes mes études.
- ❖ Mes sœurs, Mes frères.
- ❖ Mon encadreur Mr. Benouaret Mohamed pour ses orientations, son encouragement et son aide.
- ❖ Mes amies, et Mes collègues.
- ❖ À tous les étudiants de ma promotion.

Remerciement

Je remercie Allah qui m'a aidé pour bien confectionner ce travail et je j'adresse également mes vifs remerciement à mes parents qui s'intéressaient à mon éducation, à mes frères et à toute ma famille.

Tous les enseignants qui m'ont enseigné tout au long de l'année scolaire, de l'école primaire à la fin des études universitaires, et j'espère ajouter ce que j'ai appris au service de la communauté et de mon pays.

Je remercie également tous mes amis et collègues qui ont partagé cette étape de ma vie.

Et à tous ceux qui m'ont aidé à atteindre mon objectif.

Sommaire

Résumé

Dédicace

Remerciement

Liste des tableaux

Liste des figures.

Introduction générale1

Chapitre I:Description générale de la machine a commande numérique(CNC).

I.1.introduction.....	2
I.2. historique de la machine CNC.....	3
I.3. Principe de fonctionnement d'une machine numérique	4
I.3.1 : Systèmes de coordonnées	5
I.3.2. Système de coordonnées absolues.....	5
I.3.3. Système de coordonnées de travail.....	6
I .4. G-code Langage de programmation.....	6
I .4.1 . Structure d'un programme pièce	7
I .4.1.1.Définition d'un Mot.....	7
I .4.1.1.1.Mot définissant une dimension.....	7
I .4.1.1.2.Mot définissant une fonction.....	7
I .4.1.2.Définition d'un Bloc.....	7
I .4.2 . Les instructions de G code	8
I .4.3 . Groupes de codes modaux	8
I .4.4 . M-code.....	8
I .4.5 . Structure d'un programme.....	9
I.5. Les différents types de machines à commande numérique	9
I.5.1. Routeur	9

I.5.2. Plasma	10
I.5.3.Laser.....	10
I.5.4. Imprimante 3D.....	11
I.5.5. Pick and Place.....	11
I.6.conclusion.....	11

Chapitre II : Conception et fabrication de notre machine.

II .1.Introduction.....	12
II.2.Les problèmes aux niveau de la fabrication.....	12
II .3.L'axe X	14
II.3.1.Assemblage de l'axe X.....	17
II.4.L'axe Y	18
II.4.1.Assemblage de l'axe Y	19
II.5.L'axe Z.....	21
II.5.1.Assemblage de l'axe Z.....	22
II.6.Recueillir les trois axes ensemble	23
II.7.Conclusion	23

Chapitre III : Description du Matériels électroniques

III.1.Introduction.....	24
III.2. la carte arduino.....	24
III.2.1.Qu'est-ce que le conseil Arduino.?......	24
III.2.2. la carte arduino.....	25
III.2.2.1.présentation de la carte arduino.....	25
III.2.2.2.Synthèse des caractéristiques.....	26
III.2.2.3. Brochage de la carte Uno.....	26
III.2.2.4.Utilisation de la carte.....	26
III.3. Les moteurs pas à pas.....	27
III.3.1. Introduction.....	27
III.3.2. Principe de commande en position d'un moteur à courant continu.....	28
III.3.3. Principe de commande d'un moteur pas à pas	28
III.3.4. Le moteur Nema 23	30
III.3.4.1. Vue d'ensemble.....	30
III.3.4.2. Caractéristiques	30
III.4.Contrôleurs de Moteurs Pas-à-Pas (driver)	31

III.4.1. Contrôleurs de Moteurs Pas-à-Pas tb6600	32
III .4.1.1. Caractéristiques.....	32
III.5. L'alimentation	32
III.6. la mini broyeur	33
III.6.1 : Caractéristiques.....	33
III .7. Interrupteur de fin de course	33
III.8. Armoire électrique	34
III.9. Conclusion.....	35.

Chapitre IV : Les logiciels de gestion de la machine CNC.

IV.1.introduction.....	36
IV.2.arduino IDE	36
IV.2.1.introduction.....	36
IV.2.2.Lancement du logiciel	36
IV.2.3.Présentation du logiciel.....	37
IV.2.4.Approche et utilisation du logiciel	38
IV.2.4.1.Le menu File.....	38
IV.2.4.2.Les boutons	39
IV.3.G code sender.....	39
IV.3.1.Fonctionnement.....	40
IV.3.2.UGS Classic.....	40
IV.3.3.Des commandes permettent de causer avec le logiciel et la carte.....	41
IV.3.4.Utilisation basique	41
IV.4.InKscape.....	41
IV.4.1. introduction.....	41
IV.4.2. interface.....	42
IV.5.grbl.....	43
IV.6. Xloader.....	43
IV.7.conclusion.....	44

Chapitre V: Test de notre machine

V .1.Introduction	45
V .2.Mode de fonctionnement	46
V .2.1.mode manuel.....	46

V .2.1.1.Exemple de commande en mode Manuel	47
V .2.2.Mode automatique	51
V.3 . Conclusion	55
Conclusion générale.....	56
Bibliographié.....	57
Annexes.....	58

Liste de figures :

Figure I.1 : Décomposition d'une machine numérique.....	5
Figure I.1 : sens de déplacements du la machine CNC.....	2
Figure I.2 : Décomposition d'une machine numérique.....	4
Figure I.3: Système de coordonnées.....	5
Figure I.4 : L'origine du système.....	6
Figure I.5 : exemple d'un programme G code.....	9
Figure I. 6 : Routeur CNC 3 axes.....	9
Figure I. 7 : coupeuses plasma CNC de marque samson.....	10
Figure I. 8 : CNC 3018 Engraving Machine 5500 mw laser.....	10
Figure I.9 : Imprimante 3D Zortrax M200.....	11
Figure I.10 : machines CNC Pick and Place.....	11
Figure II .1 : l'équipement de cnc(1).....	12
Figure II .2 : l'équipement de cnc(2).....	13
Figure II .3 : Assemblage de l'axe x.....	17
Figure II .4 : Assemblage de La vis sans fin-écrouavec Axes de guidage.....	19
Figure II .5 : Assemblage de l'axe de la face avant.....	20
Figure II .6 : Assemblage de l'axe de la face arrière	20
Figure II. 7: Assemblage de l'axe Z(1)	22
Figure II .8 : Assemblage de l'axe Z(2).....	22
Figure II .9 : assemblage des 3 axes(1).....	23
Figure II .10 : assemblage des 3 axes(2)	23
Figure III.1 : les domaines d'utilisations de la carte arduino.....	25
Figure III.2 : image réel de la carte arduino.....	25
Figure III.3 : Brochage de la carte Uno.....	26
Figure III.4 : image réel de moteur pas a pas	27
Figure III.5 : boucle d'un moteur à courant continu.....	28
Figure III.6 : boucle d'un moteur pas à pas.....	28
Figure III.7 : Illustration du moteur de l'intérieur.....	28
Figure III.8 : fonctionnement d'une pas complet.....	29
Figure III.9 : Circuit électronique montré le contrôle de moteur pas a pas.....	29
Figure III.10 : le principe du Pont en H	30
Figure III.11: moteur Nema 23	30

Figure III.12 : Schéma du pont en H	31
Figure III.13: câblage de contrôleur de moteur avec la carte arduino et le moteur.....	32
Figure III.14 : source d'alimentation: +12V ,10A.....	32
Figure III.15 : mini broyeur (CROWN CT 13428).....	33
Figure III.16 : Interrupteur de fin de course	34
Figure III.17 : Armoire électrique.....	34
Figure IV. 1 : L'écran de démarrage Arduino.....	36
Figure IV.2 : L'interface de l'IDE Arduino.....	37
Figure IV.3 : L'interface de l'IDE Arduino en détail.....	37
Figure IV.4 : Le menu Fichier.....	38
Figure IV.1.5 : La barre d'outils.....	39
Figure IV.6 : Fenêtre principale UGS Classic.....	40
Figure IV.7 : UGS Classic avec visualiser.....	40
Figure IV.8 : Fenêtre principale(1).....	43
Figure IV.9 : Fenêtre principale(2).....	43
Figure IV.10 : Capture d'écran de logiciel Xloader	44
Figure V.1: les équipements de la machine CNC.....	45
Figure V.2 : organigramme mode manuel.....	46
Figure V.3: programme arduino(1).....	47
Figure V.4: programme arduino(2).....	48
Figure V.5: programme arduino(3).....	48
Figure V.6: charger GRBL vers la carte arduino.....	49
Figure V.7 : Initialiser les axes au point zéro manuellement sur logiciel G code	49
Figure V.8 : Commander les moteurs directement.....	50
Figure V.9 : organigramme mode automatique.....	51
Figure V.10 : Concevez la forme à dessiner sur Inkscape,.....	52
Figure V.11 : convertissez ce forme de fichier G_code	52
Figure V.12 : Initialiser les axes au point zéro manuellement sur logiciel G code	53
Figure V.13 : Importer le fichier G-code de la pièce à usiner dans l'interface de commande.....	53

Figure V.14 :exemple de G-code.....	54
Figure V.15 :test réel.....	55

Liste des tableaux

Tableau I.1: produit de la machine CNC	10
Tableau II.1 : matériels utilisés au l'axe x	14
Tableau II.2 : matériels utilisés au l'axe Y.....	18
Tableau II.3 : matériels utilisés au l'axe Z.....	21
Tableau III.1 : Synthèse des caractéristiques de la carte arduino.....	26
Tableau III .2 : Tableau récapitulatif de l'ordre des phases.....	29
Tableau IV.1 : les versions de UGS	39

Introduction 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 de solutions numériques efficaces avec une possibilité d'implanter des algorithmes plus complexes.

Après une première génération de commandes numériques à logique câblée sont apparues les commandes numériques par ordinateur (CNC), ou par ordinateur, qui intègrent un ou plusieurs ordinateurs spécifiques pour réaliser tout ou partie des fonctions de commande. Tous les systèmes de commande numérique commercialisés actuellement contenant au moins un microprocesseur, les termes CN et CNC peuvent être considérés comme des synonymes.

Donc le but de ce projet est de présenter les étapes clés pour réaliser cette machine qui est capable de Dessin et découpe sur bois, sur aluminium, ainsi que sur plastique.

Chapitre I : Description générale de la machine à commande numérique(CNC) :

I.1. introduction:

La machine CNC (Computer Numerical Control) ou bien La machine à commande numérique est une machine contrôlée par ordinateur qui peut être utilisée pour fabriquer des pièces très précises.

La machine enlève le matériau de la pièce avec un outil de coupe rotatif.

Pour ce faire, la machine guide l'outil dans les trois directions du système de coordonnées cartésiennes, c'est-à-dire le long des axes X, Y et Z.

Les flèches dans la figure ci-dessous illustrent les trois directions dans lesquelles la fraiseuse CNC peut se déplacer.

En Blue la direction sur l'axe X.

En Rouge la direction sur l'axe Y.

En Vert la direction sur l'axe Z.

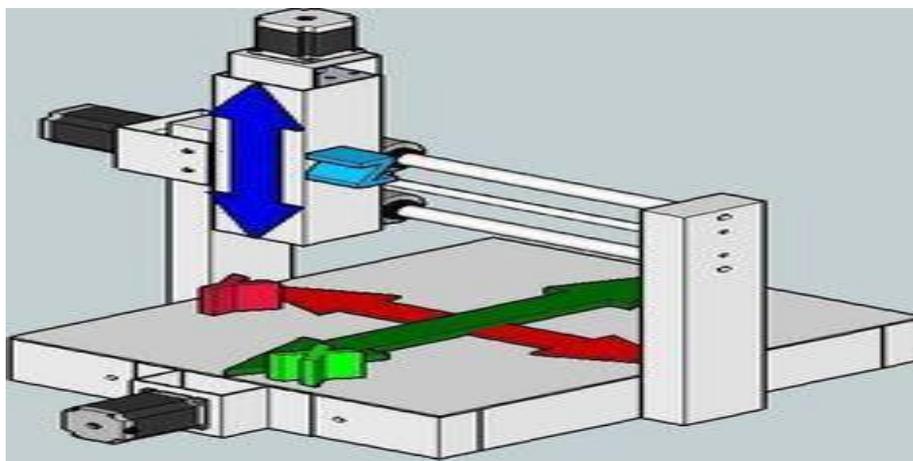


Figure I.1 :sens de déplacements du la machine CNC.

La fraiseuse CNC peut usiner de nombreux matériaux différents, tels que l'acier, l'aluminium, le laiton, le cuivre et le plastique. Un moulin n'est pas conçu pour couper du bois et ne devrait pas être utilisé pour cela.

Des exemples d'opérations pouvant être effectuées dans une fraiseuse à commande numérique consiste à découper une forme de profil, à graver du texte, à percer des trous et à créer des alvéoles de roulement.

I.2. historique du CNC :

Les premières machines commerciales à commande numérique ont été construites dans les années 1950 et ont évolué à travers plusieurs étapes :

1950 - MIT développé une fraiseuse à commande numérique (CN)

1952-Parsons dépose un brevet sur “Appareil de positionnement pour machine-outil à commande de moteur”. La date était le 5 mai 1952

1953 - À l'aide d'un système de lecture sur bande magnétique, un système de commande numérique nommé Numericord a été mis au point en avril 1953 par les efforts conjoints de G & L, du MIT et de General Electric Co. (Cuttingtoolengineering 2005

1956- Année de programmation automatique de l'usinage à commande numérique.

1960- Premier contrôleur avec technologie à transistor introduit.

1960 - Le contrôle numérique direct (DNC) élimine les programmes de perforation de la bande de papier et permet aux programmeurs d'envoyer des fichiers directement de l'ordinateur au contrôleur de la machine-outil.

1968- Premier centre d'usinage commercialisé par Kearney et Trecker (constructeurs de machines-outils).

Machine à commande numérique des années 1970 développée.

Années 1980 - Application informatique à base graphique développée.

1997- Introduction de systèmes OMAC (Open Modular Architecture Control) basés sur une fenêtre PC pour remplacer les contrôleurs «firmware».

Plus récemment, les microprocesseurs ont rendu les commandes CNC encore moins chères, avec comme point culminant la disponibilité de la commande numérique pour le marché des loisirs personnels et de la commande numérique par ordinateur. Le projet de contrôleur de machine amélioré, ou EMC2, était un projet visant à mettre en œuvre un contrôleur Open Source à commande numérique lancé par le NIST, l'Institut national de la normalisation et de la technologie, à titre de démonstration. Quelque temps en 2000, le projet est passé dans le domaine public et Open Source, et EMC2 est apparu peu de temps plus tard en 2003.

Le fondateur d'Arts Finnerty, fondateur d'Art soft, a développé Mach3 comme un dérivé des premières versions d'EMC, qui fonctionnait sous Windows au lieu de Linux, le rendant ainsi encore plus accessible au marché des commandes numériques personnelles. La société Art, Art Soft, a été fondée en 2001. Les logiciels EMC2 et Mach3 sont tous deux vivants et en plein essor, de même que de nombreuses autres technologies CNC.

I.3.Principe de fonctionnement d'une machine numérique :

Les machines à commande numérique (CNC) remplacent progressivement les machines traditionnelles. La Conception et Fabrication assistée par Ordinateur (CFAO) est surtout utilisée dans les productions en moyennes et grandes séries (quantité de pièces identiques), car le coût de mise en œuvre est élevé (personnel qualifié, programmation parfois longue). Un poste de CFAO est un système automatisé. La position de l'outil est connue à tout moment grâce à des "moteurs pas à pas" sur les axes Ox, Oy et Oz ,Ce type de machine se compose donc de deux parties complémentaires (figure I.1):

- La partie opérative.
- La partie commande.

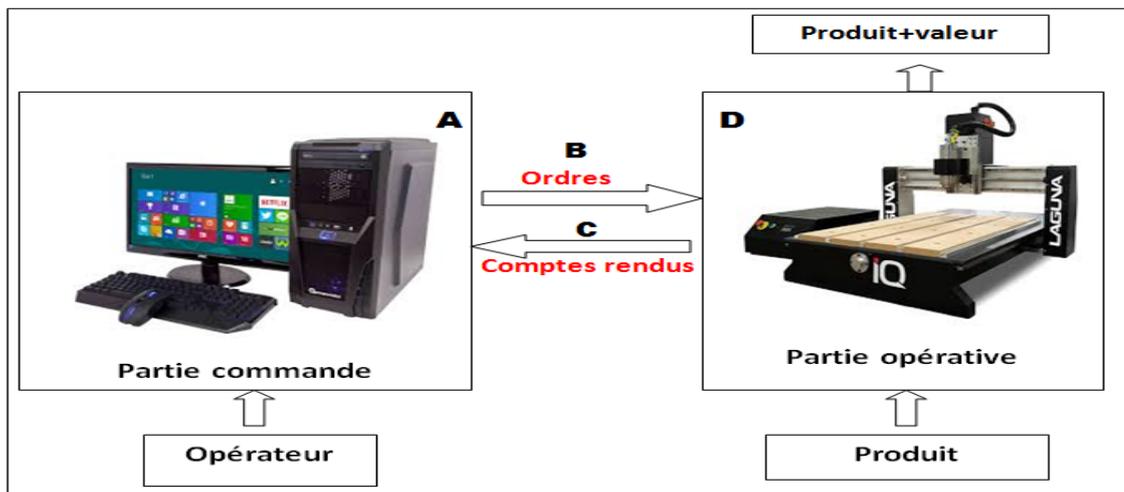


Figure I.2 : Décomposition d'une machine numérique.

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.

A la fin de l'usinage, on doit contrôler la pièce. On peut, alors, fabriquer un grand nombre de pièces identiques.

I.3.1 : Systèmes de coordonnées :

Comme s'est indiqué précédemment, la fraiseuse CNC peut se déplacer dans les directions X, Y et Z du système de coordonnées cartésiennes. Le système de coordonnées est défini comme s'est illustré ci-dessous

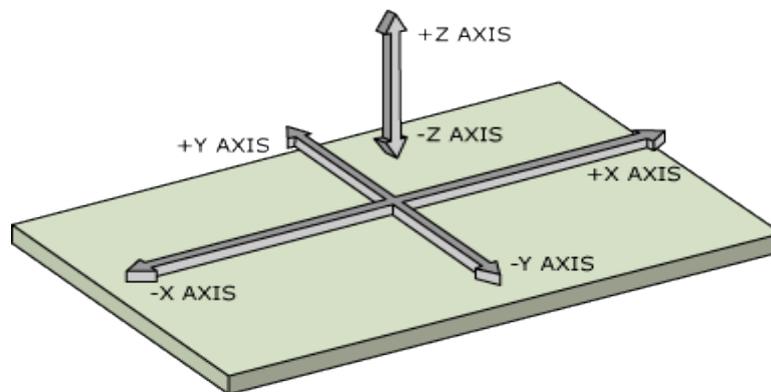


Figure I.3: Système de coordonnées.

I.3.2. Système de coordonnées absolues :

Le système de coordonnées absolu de la fraiseuse CNC est appelé système de coordonnées machine et son origine est située à l'extrémité des trois axes. Depuis l'origine de la machine, les axes X et Y ne peuvent se déplacer que dans le sens positif et l'axe Z ne peuvent pas se déplacer dans le sens négatif.

L'origine du système de coordonnées de la machine s'appelle Position d'origine. À chaque démarrage de la machine, l'opérateur doit effectuer un cycle de référencement pour trouver la position d'origine. Cela permet à la machine de savoir où elle se trouve. En effet, La machine trouve sa position d'origine en amenant lentement chaque axe à l'extrême limite (-X, -Y, +Z). Lorsque chaque axe atteint sa limite mécanique, un capteur déclenche et arrête le mouvement. Ainsi, lorsque cela a été effectué sur les trois axes, la position de la machine est à l'origine du système de coordonnées de la machine.

Il est très important que la machine soit référencée par rapport à sa position absolue car le

contrôleur via le programme G-Code va effectuer certains mouvements relativement à ce système de coordonnées. Les limites switches, en tant que capteurs de fin de courses, sont également repérés par rapport à ce système de coordonnées.

I.3.3. Système de coordonnées de travail :

Lors de la programmation de la fraiseuse CNC, il est utile de définir un système de coordonnées personnalisé pouvant être défini n'importe où sur la pièce à fraiser.

Un système de coordonnées de travail peut être configuré en déplaçant la machine de sorte que le massicot se trouve à l'emplacement approprié qui s'appelle souvent l'origine du système de coordonnées de travail.

L'origine du système de coordonnées de travail sera généralement définie au centre de la pièce et avec la pointe de la fraise directement au-dessus de la pièce.

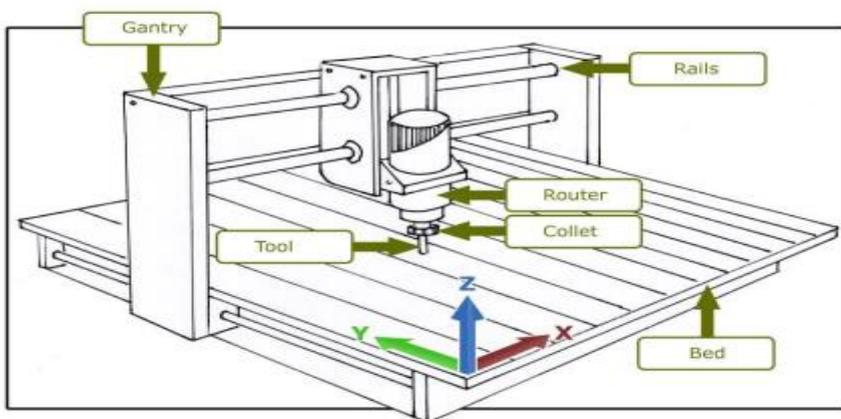


Figure I.4 : L'origine du système du travail

I.4. Langage de programmation (G -code):

Lorsqu'on désire fabriquer une pièce selon nos besoins, alors, comment faire bouger la machine et usiner une telle pièce ? Cela sera fait en exécutant un programme adapté à notre CNC. Un programme CNC est chargé dans le microcontrôleur cible à partir d'un environnement convivial en l'occurrence le « Universal G-code Sender », qui sera ensuite exécuté, ce qui entraîne l'exécution par la machine des mouvements programmés dans le fichier en question.

Le langage de programmation G-Code est le nom du langage que la machine CNC peut comprendre et convertir en mouvement réel de la machine. L'ordinateur de supervision lit le fichier G-Code ligne par ligne et demande à la machine, à travers le microcontrôleur, d'effectuer les mouvements escomptés. Le langage consiste en un ensemble de codes appelés codes G et codes M, chaque instruction ayant une fonction spécifique, par exemple effectuer un mouvement linéaire d'un point à un autre.

I.4.1 . Structure d'un programme pièce :

Un programme pièce en commande numérique se compose d'un ensemble de blocs ou instructions, ces blocs ou instructions sont constitués de mots composés de lettres majuscules et d'un format numérique

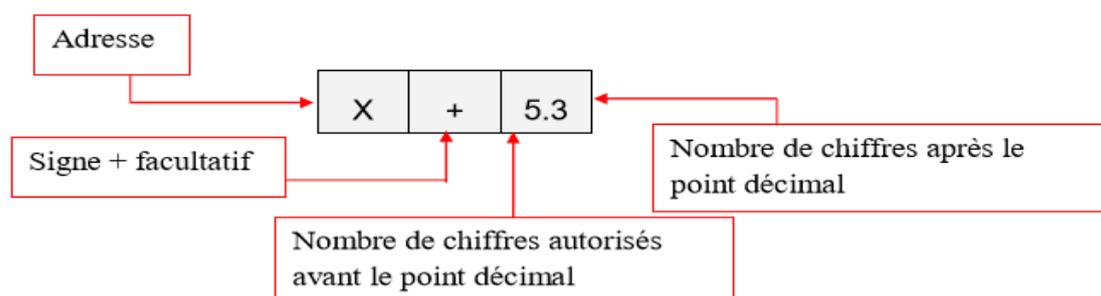
I.4.1.1.Définition d'un Mot :

Un mot définit une instruction ou une donnée à transmettre au système de commande. Il existe deux types de mots :

- Mots définissant des dimensions
- Mots définissant des fonctions

Adresse	Signe algébrique	Donnée numérique
1 Ou 2 lettres ou 1 caractères	. + -	0 1 2 3
Exemple :	X60.1 Z -30.725.	

I.4.1.1.1. Mot définissant une dimension :



I.4.1.1.2.Mot définissant une fonction :



I.4.1.2.Définition d'un Bloc :

Un bloc est une ligne d'instruction composée de mots codés à transmettre au système de commande.

Exemple : N20	G96	M4	F 500	S150
----------------------	------------	-----------	--------------	-------------

I.4.2 . Les instructions de G code :

(G-code signifie «code géométrique» et suit certaines variations du modèle alphanumérique).

N, G, X, Y, Z, F, S, T, M :

N: Numéro de ligne.

G: Mouvement.

X: Position horizontale.

Y: Position verticale.

Z: Profondeur.

F: Avance.

S: Vitesse de rotation de la broche.

T: Sélection de l'outil.

M: Fonctions diverses.

I et J: Centre incrémental d'un arc.

R: Rayon d'un arc.

I.4.3 . Groupes de codes modaux :

Les groupes de codes modaux permettent d'avoir plusieurs codes sur une seule ligne, mais il ne peut y avoir qu'un seul code de chaque groupe sur une ligne. En effet, les codes d'un groupe se substituent.

Les groupes modaux pour les codes g sont les suivants:

Groupe 1 (motion): G00, G01, G02, G03, G80, G81, G82, G84, G85, G86, G87, G88, G89

Groupe 2 (sélection des plans - XY, YZ, ZX): G17, G18, G19

Groupe 3 (mode absolu / incrémental): G90, G91

Groupe 4 (mode d'avance): G93, G94

Groupe 5 (unités - pouces / millimètres): G20, G21

Groupe6 (compensation du rayon de fraise - CRC): G40, G41, G42

Groupe 7 (correction de longueur d'outil - TLO): G43, G49

Groupe 8 (mode de retour dans les cycles fixes): G98, G99

Groupe 9 (sélection du système de coordonnées de travail): G54, G55, G56, G57, G58, G59)

I.4.4 . M-code :

La ligne d'un programme peut également inclure des m-codes, qui sont généralement des codes indiquant à une machine comment effectuer une action. Bien que la garantie ne soit pas identique sur toutes les machines, voici quelques-uns des codes m standard les plus courants:

M00: arrêt du programme

M01: arrêt de programme optionnel

M02: Fin du programme

M03: broche dans le sens horaire

M04: broche dans le sens antihoraire

M05: arrêt de la broche

M06: Changement d'outil

M08: Arroser le liquide de refroidissement

M09: Arroser le liquide de refroidissement

M30: Fin du programme / retour au début

M41: Gamme de vitesse basse de la broche

M42: Gamme de vitesse élevée de la broche

I.4.5 . Structure d'un programme :

Un programme pièce débute par le symbole % plus commentaire et finit par M30(fin de programme)

% 01

N10 T1D1

N20 G96 S150 M4

N30 G92 S1000

N40 G0 X60 Z10

N50 G1 X52 Y0 F0.2

N60 G68 X17 Y0 C0.5 D0.2 L0.1 M0.1 F0.2 H0.1 S80

N70 G0 X28 Y100 M30

I.5. Les types de CNC et domaines d'utilisations:

Lorsque la CNC a été inventée, cette technologie était adaptée aux machines existantes. Aujourd'hui, la technologie CNC est encore en train d'être adaptée à diverses machines-outils, mais de nombreuses machines sont créées dans le seul but de devenir des machines à commande numérique, Il en existe plusieurs types :

I.5. 1. Routeur :

Un routeur à commande numérique peut graver du bois, du métal ou du plastique. Il s'agit d'un type de machine à commande numérique très répandu. L'utilisateur ne manipule pas le routeur, il ne saisit que des informations dans l'ordinateur.

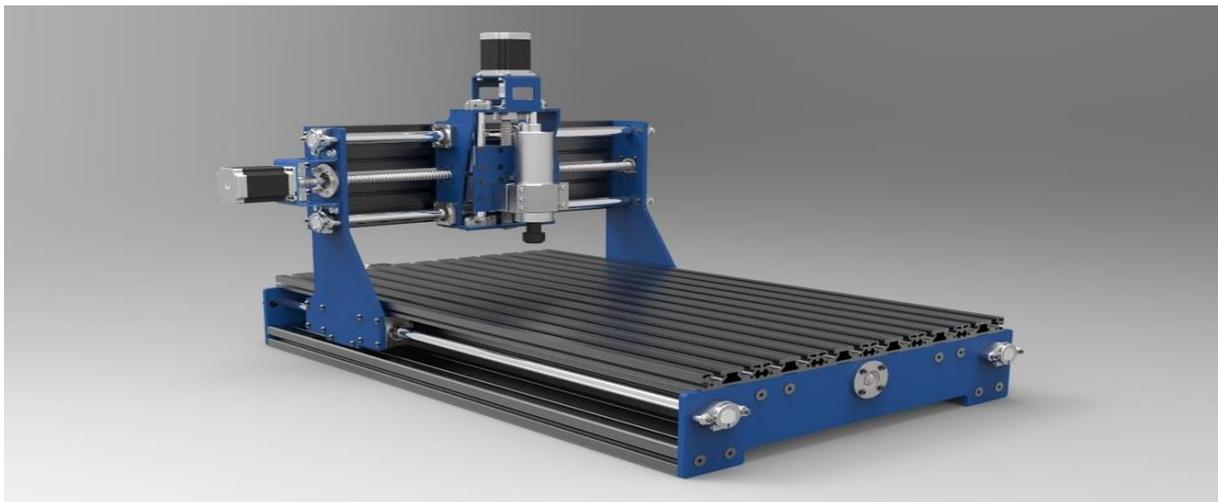


Figure I. 6 :Routeur CNC 3 axes

I.5. 2 .Plasma :

Les coupeuses plasma CNC sont utilisées pour couper le métal et le bois (2 dimensions) et ne nécessitent pas autant de puissance qu'un routeur CNC. Ces machines utilisent une torche à plasma pour pénétrer dans le bois ou la tôle.

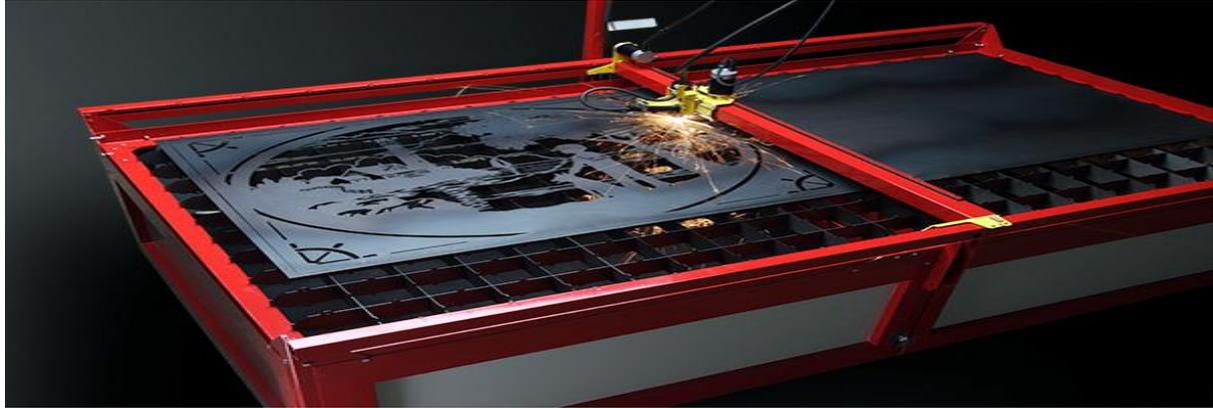


Figure I. 7 :coupeuses plasma CNC de marque samson.

I.5. 3. Laser :

- Les découpeuses laser CNC fonctionnent assez bien comme les découpeuses plasma CNC, mais au lieu de cela, un laser est utilisé pour couper le bois ou le métal. Les lasers peuvent également être utilisés pour couper du plastique.



Figure I. 8 :CNC 3018 Engraving Machine 5500 mw laser

I.5. 4. Imprimante 3D :

- Les imprimantes 3D utilisent la technologie CNC et utilisent une technologie similaire à celle des lasers à commande numérique, mais utilisent une extrudeuse pour déposer lentement le plastique dans un mouvement cohérent jusqu'à ce que le produit souhaité soit complet.

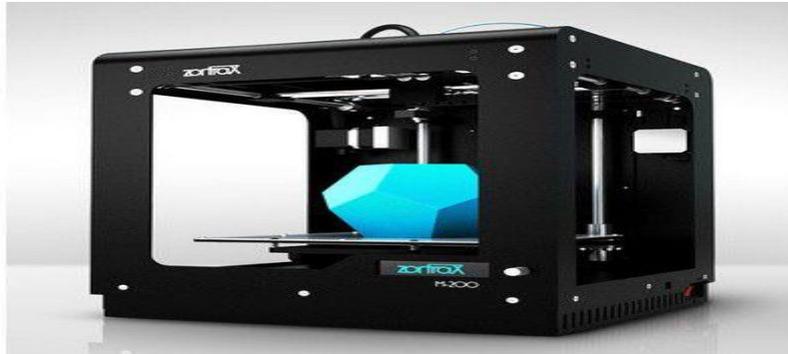


Figure I. 9 : Imprimante 3D Zorfox M200

I.5. 5. Pick and Place :

- Les machines CNC Pick and Place sont constituées de plusieurs buses qui captent les composants électriques des équipements électroniques et les placent à l'emplacement souhaité. Ceux-ci sont très couramment utilisés dans la construction de téléphones portables, ordinateurs, tablettes, etc.



Figure I.10 :machines CNC Pick and Place

I.6.conclusion :

Dans ce chapitre, nous avons donné un aperçu assez complet de la machine à commande numérique, telle que les étapes historiques de son développement, le mode de fonctionnement de la machine-outil, ainsi que les différents types de cette machine et de certains de ses produits.

Chapitre II : Conception et fabrication de notre machine :

II.1.Introduction :

Après avoir donné un aperçu détaillé sur le principe de la commande numérique, maintenant nous allons nous focaliser sur la phase de la conception de notre machine, ce processus passe par plusieurs étapes : l'étape de Conception et fabrication de notre machine , l'étape décrivant le matériel électronique nécessaire pour mener à bien notre réalisation, la phase visant à mettre en exergue les logiciels que nous allons utiliser, et enfin le test opérationnel de la machine avant d'évoquer la conclusion et les perspectives.

On va tout d'abord, commencer par la partie de la conception et la fabrication de notre machine. Dans cette partie-là un rapport holistique concernant la conception sera développé et détaillé, les problématiques du sujet et leurs solutions seront également abordées, ainsi que toutes les étapes essentielles concernant la fabrication des éléments de base de la machine à trois axes (axe X, axe Y, axe Z).

En réalité, un Axe d'une machine CNC est composé de la façon suivante :

- ❖ Un chariot mobile assisté par des guides mécaniques.
- ❖ Un système de transmission vis-écrou (vis à billes).
- ❖ Un moteur adéquat d'entraînement.
- ❖ Pièces de fixations

II.2. Les problèmes liés à la fabrication :

Les problèmes majeurs dans notre projet sont le manque flagrant d'équipements mécaniques sous formes des pièces élémentaires qui seront indispensables pour parachever notre application. Nous avons donc dû faire de nombreux ajustements d'équipements pour satisfaire nos besoins en matière des pièces nécessaires pour notre réalisation. Ains les images ci-dessous montrent l'équipement original de la machine à commande numérique.



Figure II.1 :L'équipement mécanique original de la machine CNC (1).



OX CNC Mechanical Kit Contents



OX CNC Plates



V-Slot Aluminum Profiles

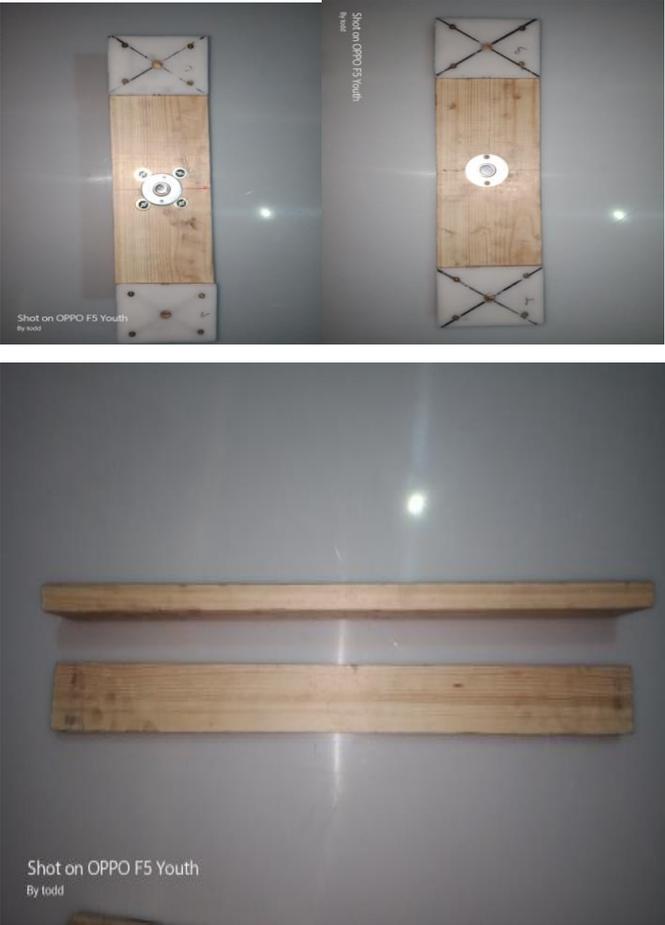
Hardwares Contents



Figure II.2 :l'équipement original de la CNC (2).

II.3.L'axe X :

Cet axe il est utilisé pour le déplacement en avant et en arrière, Ce tableau ci-dessous(**tableau II.1**) Montre le matériels utilisé.

Composants	Description
<p data-bbox="199 528 577 562">Les pièces en bois de l'axe X</p> 	<p data-bbox="917 745 1375 837">Pour l'axe x vous devrez couper au total quatre pièces de bois.</p> <p data-bbox="917 857 1023 891">Ils sont:</p> <p data-bbox="917 911 1375 1111">Deux pièces afin de porter les deux tubes de guidage et le Vis sans fin-écrou et aussi la table qui portera l'axe Y.</p> <p data-bbox="917 1131 1331 1223">Les dimensions de chacun sont: 48 x 13 cm.</p> <p data-bbox="917 1243 1375 1442">Deux pièces latérales pour relier les deux premières pièces et réduire les vibrations, avec dimensions: 62,5 x 6,5 cm.</p>

La vis sans fin-écrou



La vis sans fin-écrou est une tige filetée qui permet à l'aide d'unécrou de transformer la vitesse de rotation Du moteur en un mouvement de translation, et aussi elle assure des déplacements de grande précision, longueur de 65 cm.

Diamètre : 1.6cm

Accouplement



Nous avons modifié la La vis sans fin-écrou et nous le suivons en interne pour la couplement avec le moteur.

Axes de guidage



Axes de guidage pour simplifier le mouvement selon les trois axes, Dimensions : 62 cm. Diamètre : 12 mm.

Écrou



permet de transformer le mouvement de rotation en mouvement droit.

<p>Vis de fixation</p> 	<p>Ils servent à relier et fixer les pièces entre elles, comme la fixation du moteur etc.</p>
<p>Les Roulement</p> 	<p>Les roulements sont des organes mécaniques servant à guider un arbre de transmission en rotation.</p>
<p>Les rondelles</p> 	<p>Pièce d'assemblage mécanique, de forme circulaire et de faible épaisseur, percée en son centre traversé d'un axe supportant une autre pièce, utilisé pour fixé les roulements.</p>
<p>Plaque en plastique</p> 	<p>Cette plaque il est utilisé pour Assurez-vous de ne pas secouer l'Axes de guidage.</p>

tableau II.1 :matériels utilisés au l'axe x .

II.3.1.Assemblage de l'axe x:

Après avoir décrit tous les équipements de cet axe, nous rassemblons maintenant ces pièces
Et voila le résultat :

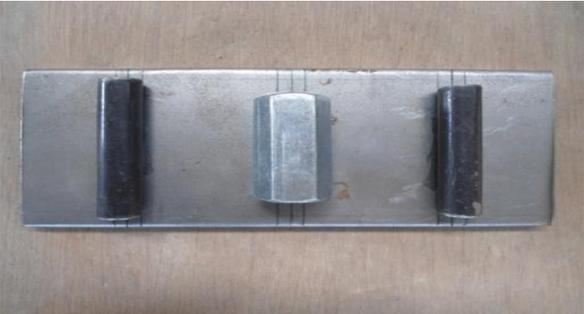


Figure II.3 :Assemblage de l'axe x .

Rq : Il y a certains équipements que nous utiliserons à nouveau dans l'axe Y et l'axe Z, ci pour ca Nous n'en parlerons pas.

II.4.L'axe Y :

Cet axe il est utilisé pour déplacer gauche et droit, Ce tableau ci-dessous(tableau II.2) montre les matériels utilisés.

Composants	Description
<p>Les pièces en bois de l'axe Y (les deux façades)</p> 	<p>Deux pièces afin de porter les deux tubes de guidage et le Vis sans fin-écrou</p> <p>Les dimensions de chacun sont: 35 x 13 cm. pour chaque une.</p>
<p>Table avec écrou et les 2 roulements linéaires</p> 	<p>L'écrou permet de transformer le mouvement de rotation en mouvement droit.</p> <p>roulements linéaires Permet au l'axe de guidage a déplacer et de réduire les vibrations</p>
<p>pièce latérale</p> 	<p>une pièce latérale pour relier les deux premières pièces en bois et réduire les vibrations, avec dimensions: 48 x 20 cm.</p>

<p>cornière en aluminium</p> 	<p>Cette cornière utilisé pour fixer les deux pièces en bois d'axe Y.</p>
	<p>Un regard sur comment placé chacun des Axes de guidage, La vis sans fin-écrou , Roulement et fixateur.</p>

Tableau II.2 :matériels utilisés au l'axe Y.

II.4.1.Assemblage de l'axe Y :

Après avoir décrit tous les équipements de cet axe, nous rassemblons maintenant ces pièces
Et voilà le résultat :



Figure II.4 : Assemblage de La vis sans fin-écrouavec Axes de guidage



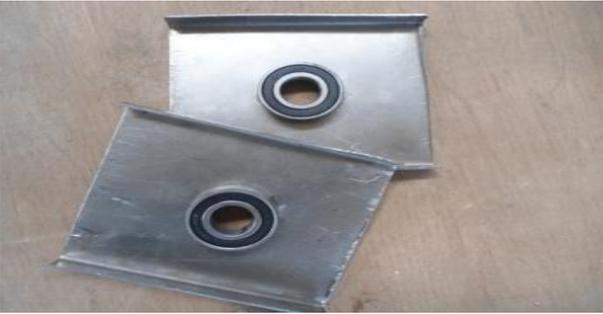
Figure II.5 :Assemblage de l'axe de la face avant.



Figure II.6 :Assemblage de l'axe de la facearrière .

II.5.L'axe Z :

Cette axe il est utilisé pour déplacer Monter et descendre, Ce tableau ci-dessous(**tableau II.3**) montre les matériels utilisés.

Composants	Description
<p>Deux pièces en métal</p>  	<p>Deux pièces afin de porter les deux tubes de guidage et le Vis sans fin-écrou</p> <p>Les dimensions de chacun sont: 8 x 7 cm. pour chaque une.</p>
	<p>Cette pièce il est utilisé pour assemblé les deux premières pièces</p> <p>Dimension : 20 x 8 cm.</p>
<p>Table avec écrou et les 2 roulements linéaires</p>  	<p>L'écrou permet de transformer le mouvement de rotation en mouvement droit.</p> <p>roulements linéaires Permet au l'axe de guidage a déplacer et de réduire les vibrations</p>

	<p>Une vue comment place les deux tubes de guidage et le Vis sans fin-écrou.</p>

tableau II.3 :matériels utilisés au l'axe Z.

II.5.1.Assemblage de l'axe Z:

Après avoir décrit tous les équipements de cet axe, nous rassemblons maintenant ces pièces
Et voilà le résultat :



Figure II . 7:Assemblage de l'axe Z(1) .



Figure II.8 :Assemblage de l'axe Z(2).

II.6.Recueillir les trois axes ensemble :

Une fois que chaque axe est installé, nous l'assemblons pour obtenir la forme finale de la machine.



Figure II.9 :assemblage des 3 axes(1) .



Figure II.10 :assemblage des 3 axes(2) .

II.7.Conclusion :

Dans ce chapitre, nous avons abordé le côté mécanique où nous avons préparé l'installation de la machine CNC, composée de trois axes,

Nous avons installé chaque axe individuellement, qui a sa propre spécificité, puis nous les combinons ensemble.

Chapitre III :Description du Matériels électroniques :

III.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 matériels qu'on a utilisés.

III.2. la carte arduino :

III.2.1.Qu'est-ce que le conseil Arduino? :

Arduino, et son synonyme Genuino2, est une marque qui couvre des cartes électroniques matériellement libres sur lesquelles se trouve un microcontrôleur

Il existe de nombreux panneaux Arduino avec des propriétés plus ou moins riches, Nous les mentionnons la carte ArduinoUno, Arduino Nano, ArduinoMega.....etc.

La caractéristique principale de la carte Arduino est un type de MCU ou microcontrôleur intégré. Actuellement, peut-être depuis longtemps, on utilise deux jeux de microcontrôleurs: les bras ATMEL 8 bits et les bras 32 bits Cortex-M, conçus par Advanced RISC Machines et fabriqués par différents fabricants, cartes ATMEL, mais également compatibles Freescale ou STMicro. À propos de la galaxie Arduino.

Pour résumer rapidement, les panneaux basés sur AVR d'Arduino sont plus simples et ont moins de mémoire que les panneaux basés sur ARM. Sa capacité informatique est faible mais généralement moins chère.

La carte la plus couramment utilisée est ArduinoUno. C'est une carte avec un petit contrôleur AVR 8 bits: ATMega328.

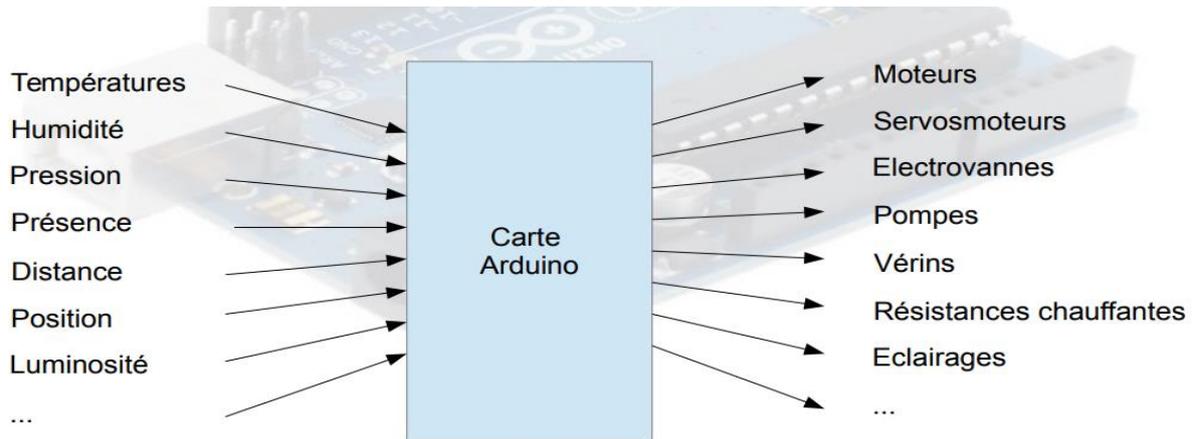


Figure III.1 : les domaines d'utilisations de la carte arduino.

III.2.2. la carte arduino :

III.2.2.1.présentation de la carte arduino :

La carte ArduinoUno est une carte à microcontrôleur Pour pouvoir l'utiliser, il suffit simplement de la connecter à un ordinateur à l'aide d'un câble USB, l'alimentation étant fournie par le port USB (on peut aussi l'alimenter avec un adaptateur secteur ou un jeu de piles via le connecteur jack).

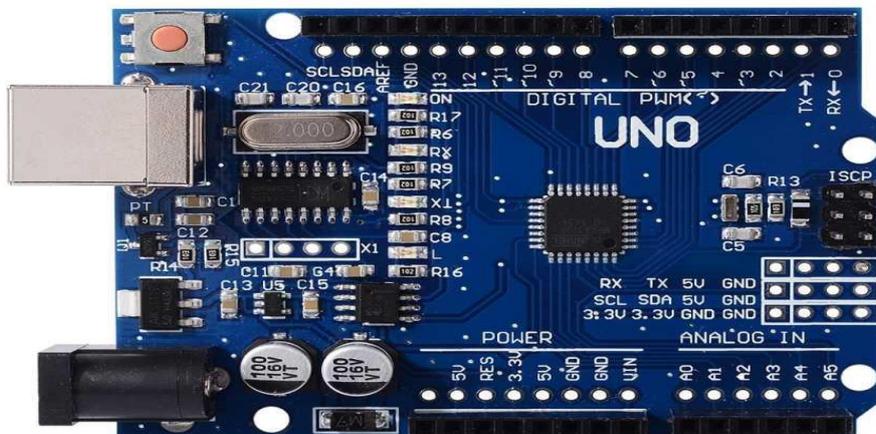


Figure III.2 :image réel de la carte arduino

Microprocesseur cadencé à 16MHZ

Mémoire flash \Leftrightarrow disque dur

Mémoire RAM \Leftrightarrow mémoire vive (RAM)

Mémoire EEPROM \Leftrightarrow mémoire morte (ROM)

Broches Entrée/Sortie \Leftrightarrow ports de communication

III.2.2.2.Synthèse des caractéristiques :

Microcontrôleur	ATmega328
Tension de fonctionnement	5V
Tension d'alimentation (recommandée)	7-12V
Consommation maxi admise sur port USB (5V)	500 mA avant déclenchement d'un fusible
Broches E/S numériques	14 (dont 6 disposent d'une sortie PWM pour commander les moteurs)
Broches d'entrées analogiques	6 (utilisables aussi en broches E/S numériques)
Intensité maxi disponible par broche E/S (5V)	40 mA par sortie, mais ATTENTION : 200mA cumulé pour l'ensemble des broches E/S)
Mémoire Programme Flash	32 Ko
Mémoire RAM (mémoire volatile)	2 Ko
Mémoire EEPROM (mémoire non volatile)	1 Ko
Vitesse d'horloge	16 MHz

Tableau III.1 : Synthèse des caractéristiques de la carte arduinouno .

III.2.2.3. Brochage de la carte Uno :

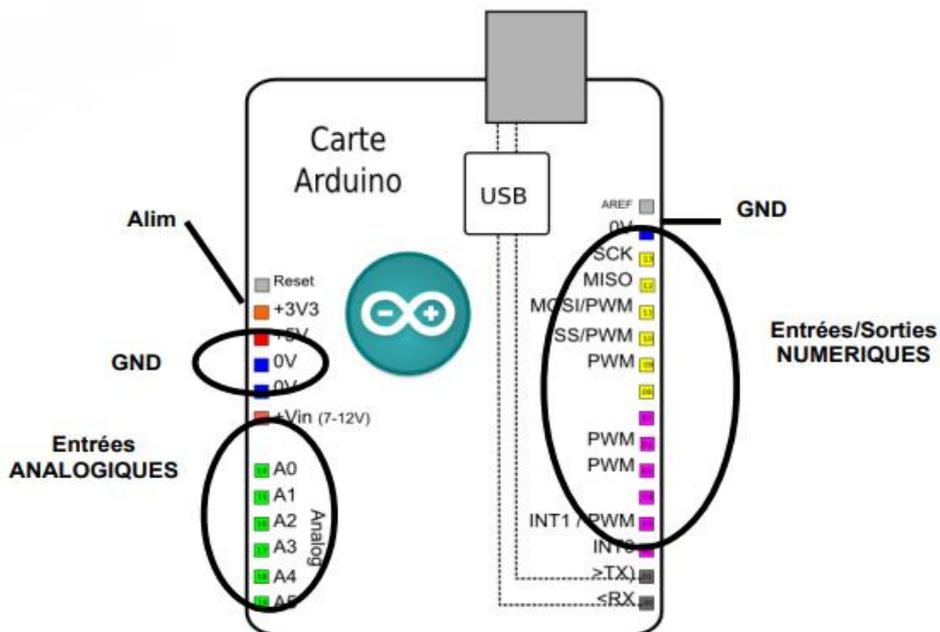


Figure III.3 : Brochage de la carte Uno .

III.2.2.4.Utilisation de la carte:

La carte arduino se programme avec le langage arduino créé à partir des langages C et C++. On ne programme pas au collège sur l'IDE arduino avec des lignes de code, mais nous le ferons en EPI ...

En mode découverte, on les fait fonctionner à partir du logiciel SCRATCH que vous connaissez déjà : Principe : la carte arduino est déjà programmée par le professeur. Une application appelée S2A (Scratch to Arduino) utilise ce programme appelé « firmware » pour établir une communication entre l'ordinateur et la carte via SCRATCH. La carte devient une simple interface entre l'ordinateur, et les composants branchés sur la carte. Ce système est agréable, car on teste « en direct » le programme sur la carte. Mais ce système est limité, car il ne permet pas la programmation réelle de la carte : on faut toujours rester connecté à l'ordinateur.

Autres possibilités : on utilise un logiciel de programmation visuelle (de type scratch) qui va traduire le programme réalisé, en lignes de code arduino. Il ne reste plus qu'à télé verser les lignes de code dans la carte via l'IDE arduino. Avantage : plus besoin d'ordinateur, la carte est autonome et devient "l'ordinateur" qui pilote. Inconvénient : si le programme ne fonctionne pas, il faut le reprendre à l'ordinateur, puis à nouveau traduire et télé verser, c'est plus long ...

III.3. Les moteurs pas à pas :

III.3.1. Introduction :

Le moteur pas à pas, ou encore stepper motor, est un moteur particulier pour lequel sa rotation s'effectue par pas. C'est à dire, qu'à l'inverse d'un petit moteur à courant continu comme on trouve dans nombreux jouets et qui tourne sans arrêt une fois alimenté, le moteur pas à pas tourne d'un angle spécifique à chaque fois qu'il reçoit une impulsion. Mais comme nous allons le découvrir, ces impulsions ont besoin d'un formatage bien particulier.



Figure III.4 : image réel de moteur pas apas .

III.3.2. Principe de commande en position d'un moteur à courant continu :

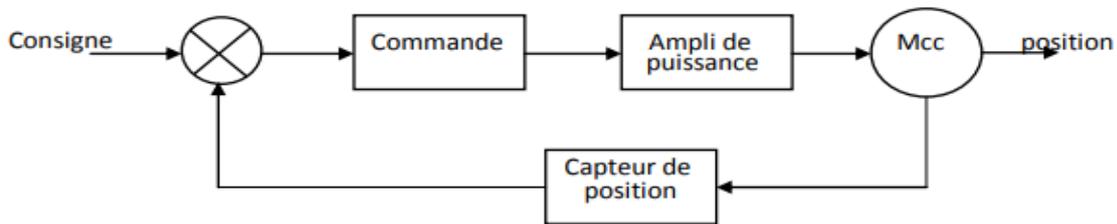


Figure III.5 : boucle d'un moteur à courant continu.

III.3.3. Principe de commande d'un moteur pas à pas :

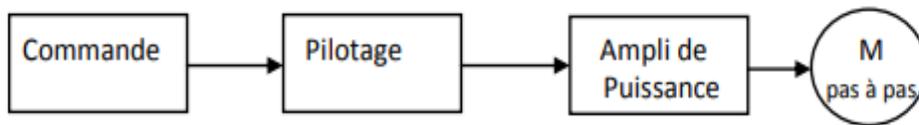


Figure III.6 : boucle d'un moteur pas à pas.

On distingue 3 groupes de moteur pas à pas :

- Les moteurs à aimant permanent
- Les moteurs à reluctance variable.
- Les moteurs hybrides (qui est une combinaison des deux technologies précédentes).

Rq : dans notre projet nous avons utilisé des moteurs hybrides (Moteur bipolaire) .

Ce type de moteurs a été choisi car il nous permet de contrôler la vitesse ou la position en boucle ouverte. il est composé par deux bobines (bobine A, bobine B), chacune contient deux pôles (A+, A-/B+, B-), ces figures en bas montrent le principe de pilotage des bobines :

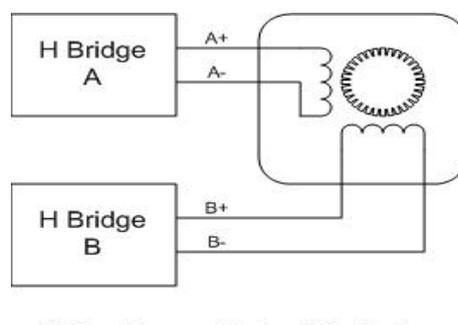


Figure III.7 Illustration du moteur de l'intérieur

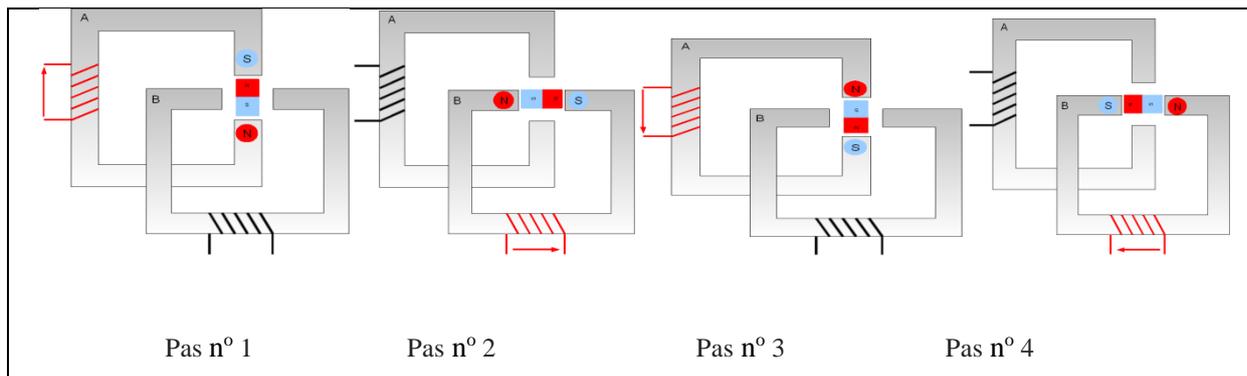


Figure III.8 : fonctionnement d'un pas complet.

Impulsion	Bobine A	Bobine A	Bobine B	Bobine B
T1	+	-		
T2			+	-
T3	-	+		
T4			-	+

Tableau III .2 :Tableau récapitulatif de l'ordre des phases.

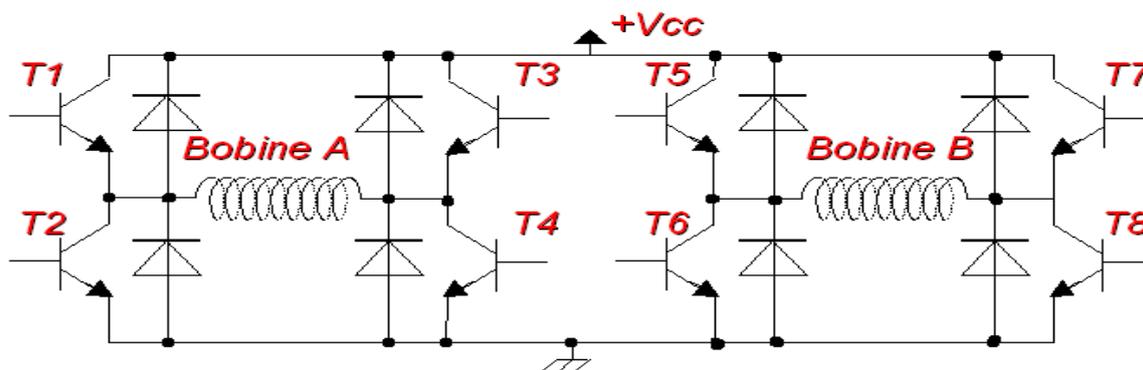


Figure III.9 : Circuit électronique montrant le contrôle de moteur pas à pas.

C'est le principe du [Pont en H](#), si on commande T1 et T4, alors on alimente dans un sens, soit on alimente en T2 et T3, on change le sens de l'alimentation, donc le sens du courant

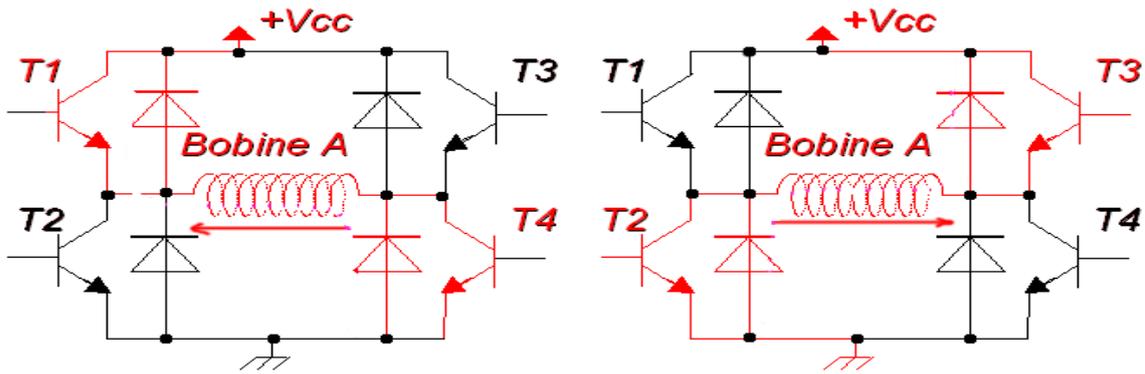


Figure III.10 :le principe du Pont en H .

III.3.4. Le moteur Nema 23 :

III.3.4.1. Vue d'ensemble :

Ce moteur pas à pas hybride à couple élevé a un angle de pas de $1,8^\circ$ (200 pas / tour). Chaque phase consomme 2,8 A à 3,2 V, ce qui permet un couple de maintien de 19 kg-cm (270 oz-in). Le moteur a quatre fils de couleur codés terminés par des fils nus: les fils noir et vert se connectent à une bobine; rouge et bleu se connectent à l'autre. Il peut être contrôlé par une paire de ponts en H appropriés (un par bobine), mais nous recommandons d'utiliser un pilote de moteur pas à pas bipolaire.



Figure III.11:moteur Nema 23 .

III.3.4.2. Caractéristiques :

Taille: 56,4 mm × 76 mm, arbre non compris (NEMA 23)

Poids: 1 kg (35 oz)

Diamètre de l'arbre: 6,35 mm (0,25 ") "D"

Pas par tour: 200

Courant nominal: 2,8 A par bobine

Tension nominale: 3,2 V

Résistance: 1,13 Ω par bobine

Couple de maintien: 19 kg-cm (270 oz-in)
 Inductance: 3,6 mH par bobine
 Longueur du fil: 30 cm (12 ")
 Arbre de sortie supporté par deux roulements à billes
 Dimensions :4 ,75 ×4,75.

III.4. Contrôleurs de Moteurs Pas-à-Pas (driver) :

Le contrôleur de moteur C'est un ensemble qui entre le moteur et l'utilisateur qui permet de convertir des signaux électriques pour que le moteur puisse les interpréter, Il contrôle la vitesse et la rotation du moteur, composée par deux partie la partie séquenceur qui responsable a le contact avec le microcontrôleur (la carte arduino) ,et les deux ponts H(Un pont en H permet d'alimenter un bobinage du moteur (une phase) en permettant de changer le sens du courant. Voici (figure –III .12-) le schéma de principe d'un pont en H :

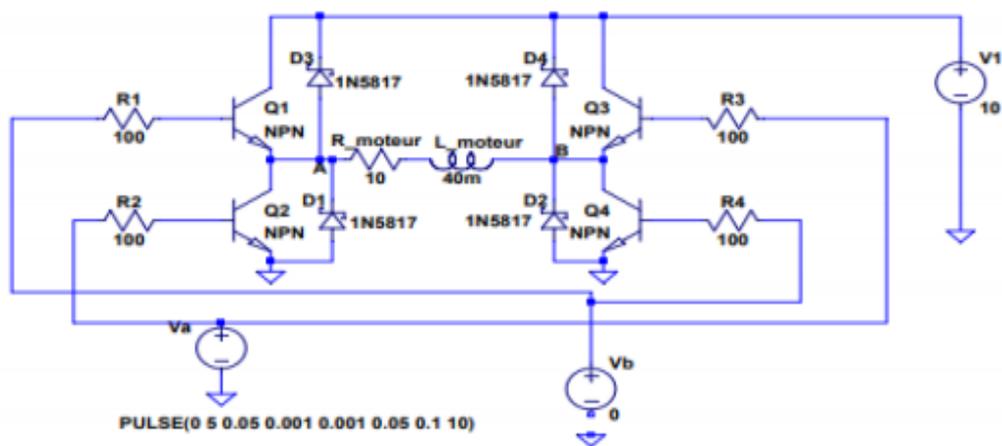


Figure III.12 : Schéma du pont en H.

Le bobinage est représenté par une résistance et une inductance. Lorsque $V_a = 5\text{ V}$ et $V_b = 0$, les transistors Q2 et Q3 sont passants et un courant positif circule dans le moteur de B vers A. Inversement, lorsque $V_a = 0$ et $V_b = 5\text{ V}$, un courant positif circule de A vers B. Lorsque les deux tensions de commande sont à zéro, le courant dans le bobinage est nul. Les diodes permettent au courant de ce réduire continûment lorsque les transistors sont bloqués (diodes de roue libre).

III.4.1. Contrôleurs de Moteurs Pas-à-Pas tb6600 :

III.4.1.1. Caractéristiques:

Alimentation CC 9V-42V, entraînement à flux constant bipolaire à pont en H. Le maximum de 4,0 A des huit types de courant de sortie.

Contrôleur de moteur: fonction de verrouillage de courant semi-automatique à économie d'énergie, protection de température intégrée et protection contre les surintensités.

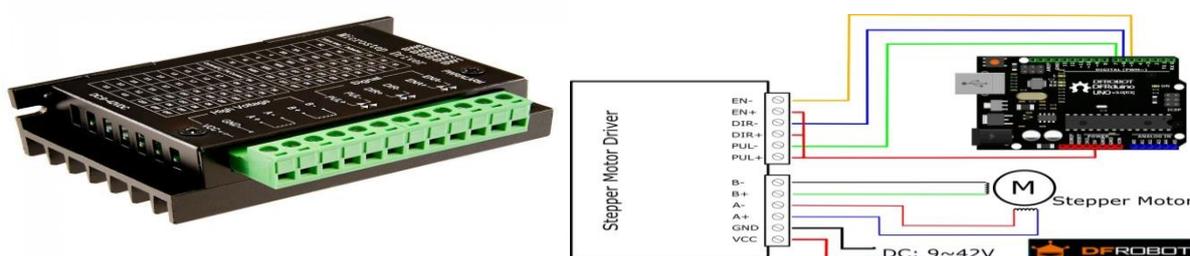


Figure III.13: câblage de contrôleur de moteur avec la carte arduino et le moteur.

III.5. L'alimentation :

Une source d'alimentation est un appareil pouvant fournir ou décharger de l'électricité. Cette électricité peut prendre la forme d'alternance, continue, polarité, polygonale, triangulaire, spécialisée, à impulsions ... ou même excentrique (harmoniques complexes). Dans notre projet, nous allons l'utiliser comme source d'alimentation pour Alimentée les Contrôleurs de Moteurs Pas-à-Pas (tb6600).

Qui incluent les caractéristiques suivantes :

AC entrée : 110 /220V , Fréquence : 50 /60HZ, DC sortie : +12V ,10A



Figure III.14 : Source d'alimentation: +12V ,10A.

III.6. la mini broyeur :

cet machine de marque (CROWN CT 13428) est utilisé pour le design et le coupage.



Figure III.15 :mini broyeur (CROWN CT 13428).

III .6.1.Caractéristiques:

Voltage/fréquence :Entrée: AC 220V/230V, 50Hz

Energie : 150w,Chuck: 0.8-3.2mm

Torsion: 530g / cm,Vitesse de rotation: 8000-35000rpm

Poids: 750g.,1× mini broyeur.,1× manuel.

Un outil universel, équipé d'un foret en acier remplaçable à haute vitesse de précision (0,8 à 3,2 mm).

Produits de haute qualité, appropriés pour le perçage, le meulage, le polissage, la coupe, etc.

Design exquis, prise confortable.

En plastique ABS, poids léger, convient aux travaux de longue durée.

Il peut changer de mandrin rapidement, sans aucun outil.

Peut changer le perçage et le mandrin différents, peut percer des trous de différentes ouvertures. Petit volume, poids léger, facile à transporter.

III .7.Interrupteur de fin de course :

En génie électrique, un interrupteur de fin de course est un interrupteur actionné par le mouvement d'une pièce de la machine ou la présence d'un objet.

Ils sont utilisés pour contrôler des machines dans le cadre d'un système de contrôle , d' inter verrouillage de sécurité ou pour compter des objets passant un point. Un interrupteur de fin de course est un dispositif électromécanique qui consiste en un actionneur relié

mécaniquement à un ensemble de contacts. Lorsqu'un objet entre en contact avec l'actionneur, l'appareil actionne les contacts pour établir ou rompre une connexion électrique. Les commutateurs de fin de course sont utilisés dans une variété d'applications et d'environnements en raison de leur robustesse, de leur facilité d'installation et de leur fiabilité. Ils peuvent déterminer la présence ou l'absence, le passage, le positionnement et la fin du voyage d'un objet. Ils ont d'abord été utilisés pour définir la limite de déplacement d'un objet; d'où le nom "Limit Switch".



Figure III.16 : Interrupteur de fin de course.

III .8.Armoire électrique :

Après avoir défini les différents matériels électroniques, nous assemblons maintenant la section de puissance et de contrôle dans une armoire, qui est composée par:

- L'alimentation 12V/10 A.
- Le microcontrôleur (la carte arduino).
- Les trois contrôleurs des moteurs.
- Les limites Switchs.

la figure en bas montre le câblage entre ces éléments



Figure III.17 : Armoire électrique

III.9. Conclusion :

Dans ce chapitre, nous avons présenté les différents appareils électroniques utilisés dans la machine à commande numérique, qu'il s'agisse de la partie commande ou de la partie opérative.

Chapitre IV :Les logiciels de gestion de la machine cnc :

IV.1.introduction :

Comme nous l'avons mentionné dans le premier chapitre, la machine à commande numérique comprend deux parties, la partie opérative et la partie commande, dans les deux chapitres précédents nous avons montré la partie pratique (PO) et nous passons maintenant à la partie contrôle(PC), qui est un ensemble de logiciels qui permettent de contrôler la machine CNC.

IV.2.arduino IDE :

IV.2.1.introduction :

L'environnement de développement intégré (IDE) Arduino est une application multiplateforme (pour Windows, macOS, Linux) écrite en langage de programmation Java. Il est utilisé pour écrire et télécharger des programmes sur des cartes compatibles Arduino, mais aussi, avec l'aide de cartes tiers, de cartes de développement d'autres fournisseurs.

Le code source de l'EDI est publié sous licence GNU General Public, version 2. L'IDE Arduino supporte les langages C et C++ en utilisant des règles spéciales de structuration du code. L'IDE Arduino fournit une bibliothèque de logiciels du projet Wiring, qui fournit de nombreuses procédures d'entrée et de sortie communes. Le code écrit par l'utilisateur nécessite seulement deux fonctions de base, pour le démarrage de l'esquisse et de la boucle du programme principal, qui sont compilées et liées à un talon de programme main() dans un programme exécutable cyclique avec la chaîne d'outils GNU, également incluse dans la distribution IDE. L'IDE Arduino utilise le programme avrdude pour convertir le code exécutable en un fichier texte au codage hexadécimal qui est chargé dans la carte Arduino par un programme de chargement dans le microprogramme de la carte.

IV.2.2.Lancement du logiciel :

Lançons le logiciel en double-cliquant sur l'icône avec le symbole "infinie" en vert. C'est l'exécutable du logiciel. Après un léger temps de réflexion, une image s'affiche :



Figure IV. 1 :L'écran de démarrage Arduino

Cette fois, après quelques secondes, le logiciel s'ouvre. Une fenêtre se présente à nous :

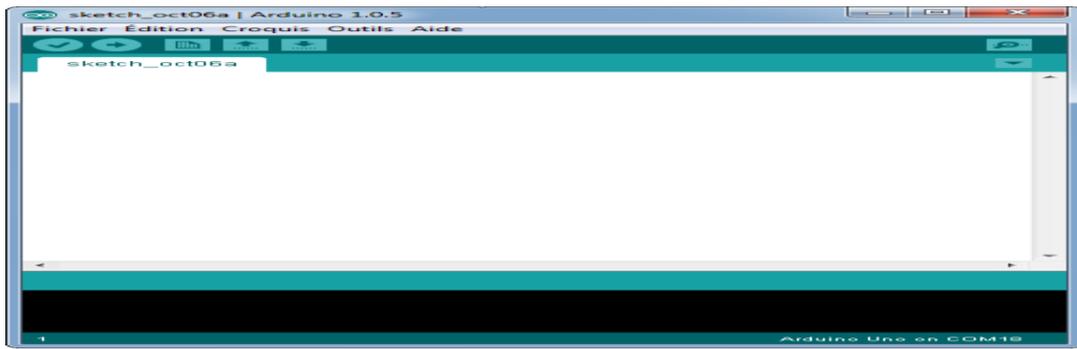


Figure IV.2 :L'interface de l'IDE Arduino

Ce qui saute aux yeux en premier, c'est la clarté de présentation du logiciel. On voit tout de suite son interface intuitive. Voyons comment se compose cette interface

IV.2.3.Présentation du logiciel:

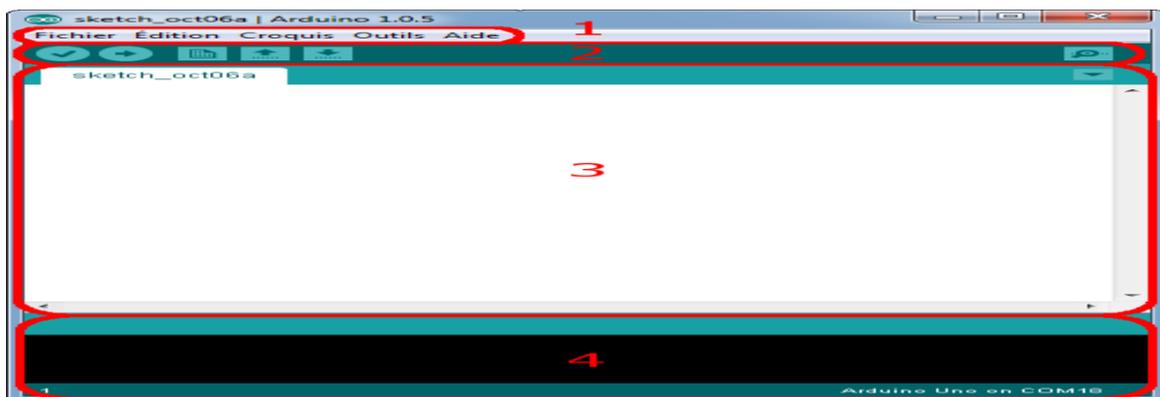


Figure IV.3 :L'interface de l'IDE Arduino en détail

Correspondance

Le cadre numéro 1 : ce sont les options de configuration du logiciel

Le cadre numéro 2 : il contient les boutons qui vont nous servir lorsque l'on va programmer nos cartes

Le cadre numéro 3 : ce bloc va contenir le programme que nous allons créer

Le cadre numéro 4 : celui-ci est important, car il va nous aider à corriger les fautes dans notre programme. C'est le débogueur

IV.2.4.Approche et utilisation du logiciel :

La barre des menus est entourée en rouge et numérotée par le chiffre 1

IV.2.4.1.Le menu File :

C'est principalement ce menu que l'on va utiliser le plus. Il dispose d'un certain nombre de choses qui vont nous être très utiles. Il a été traduit en français progressivement, nous allons donc voir les quelques options qui sortent de l'ordinaire :

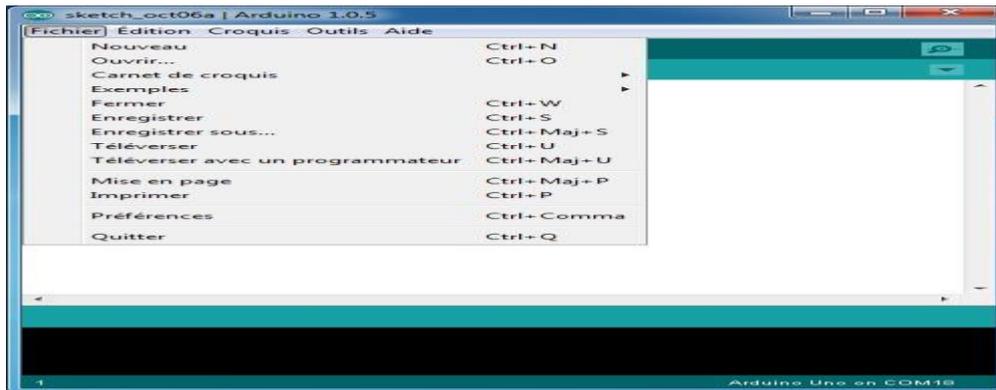


Figure IV.4 :Le menu Fichier

Carnet de croquis : Ce menu regroupe les fichiers que vous avez pu faire jusqu'à maintenant (et s'ils sont enregistrés dans le dossier par défaut du logiciel).

Exemples (exemples) : Ceci est important, toute une liste se déroule pour afficher les noms d'exemples de programmes existants ; avec ça, vous pourrez vous aider/inspirer pour créer vos propres programmes ou tester de nouveaux composants.

Télé verser : Permet d'envoyer le programme sur la carte Arduino. Nous y reviendrons ;).

Télé verser avec un programmeur : Idem que ci-dessus, mais avec l'utilisation d'un programmeur (vous n'en n'aurez que très rarement besoin).

Préférences : Vous pourrez régler ici quelques paramètres du logiciel. Le reste des menus n'est pas intéressant pour l'instant.

IV.2.4.2. Les boutons :

Voyons à présent à quoi servent les boutons, encadrés en rouge et numérotés par le chiffre 2.

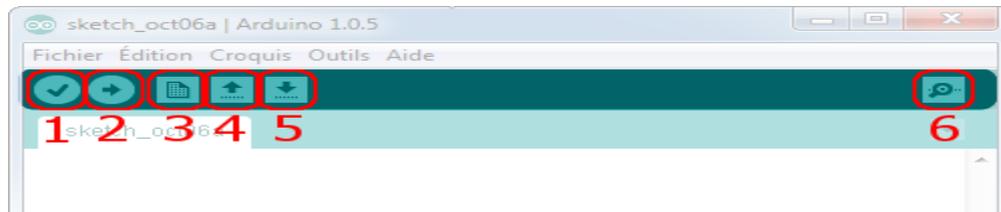


Figure IV.1.5 : La barre d'outils

Bouton 1 : Ce bouton permet de vérifier le programme, il actionne un module qui cherche les erreurs dans votre programme

Bouton 2 : Charge (téléverse) le programme dans la carte Arduino.

Bouton 3 : Crée un nouveau fichier.

Bouton 4 : Ouvre un fichier.

Bouton 5 : Enregistre le fichier.

Bouton 6 : Ouvre le moniteur série.

IV.3.G code sender :

Universel G-Code Sender est un expéditeur multiplateformes basé sur Java et compatible avec [GRBL](#) et [TinyG / g2core](#) . Utilisez ce programme pour exécuter (piloter) une machine CNC contrôlée par GRBL ou TinyG / g2core.

Il existe deux versions UGS :

Version	La description
<u>UGS Classic</u>	La version nocturne de la version classique avec une interface utilisateur propre et légère
<u>Plateforme UGS</u>	La version nocturne de la nouvelle génération, une version riche en fonctionnalités basée sur la plate-forme Netbeans

tableau IV..1 : les versions de UGS .

IV.3.1.Fonctionnement

Assurez-vous que Java 8 est installé.

Téléchargez UGS Classic ou UGS Platform et décompressez le fichier .zip.

- Pour UGS Classic, double-cliquez simplement sur le fichier jar. Sur certaines plates-formes, vous devrez peut-être exécuter le script de démarrage inclus.
- Pour UGS Platform, exécutez le script de démarrage:bin/ugsplatform

Dans notre projet nous allons utiliser le logiciel UGS classic.

IV.3.2.UGS Classic :



Figure IV.6 : Fenêtre principale UGS Classic

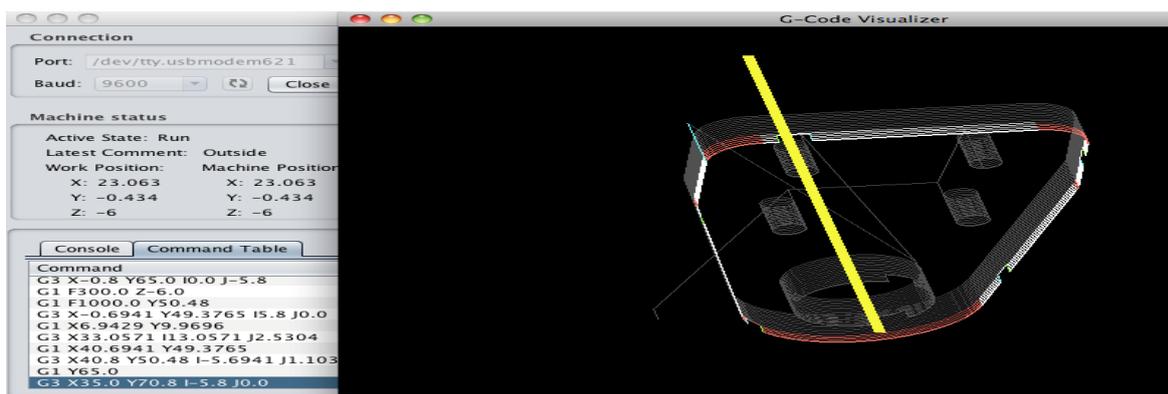


Figure IV.7 : UGS Classic avec visualiser

IV.3. 3.Des commandes permettent de causer avec le logiciel et la carte :

\$: Une aide sommaire.

\$\$: Voir les paramètres de la machines et permet de les modifier (par exemple je tape.

\$110=1490 : qui va changer la valeur \$110 (vitesse max en x).

\$H homing : la machine va chercher son zéro machine (capteur de fin de course X et Y).

Attention au bouton return to zéro !

Il va chercher le 0(travail) en X Y et Z en même temps... ce qui est dangereux pour le Z , la broche peut toucher le travail en cours.

Il faut préférer faire deux commandes Gcode G90 X0 Y0; puis G90 Z0;

Vous pouvez créer des macros avec ces deux commandes pour éviter de les taper à chaque fois !

IV.3.4.Utilisation basique :

Allumer la machine, ouvrir universel G code Sender

Commencer par un “homing” pour aller chercher le 0 machine (capteur fin de course)

Ensuite faites le 0 logiciel en plaçant la broche à l'endroit ou vous souhaitez commencer 0,0

Ouvrir un fichier G code.

Visualiser le fichier

Lancer (Send).

IV.4.Inkscape :

IV.4.1. introduction :

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.

IV.4.2. 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 lcanevas.
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.

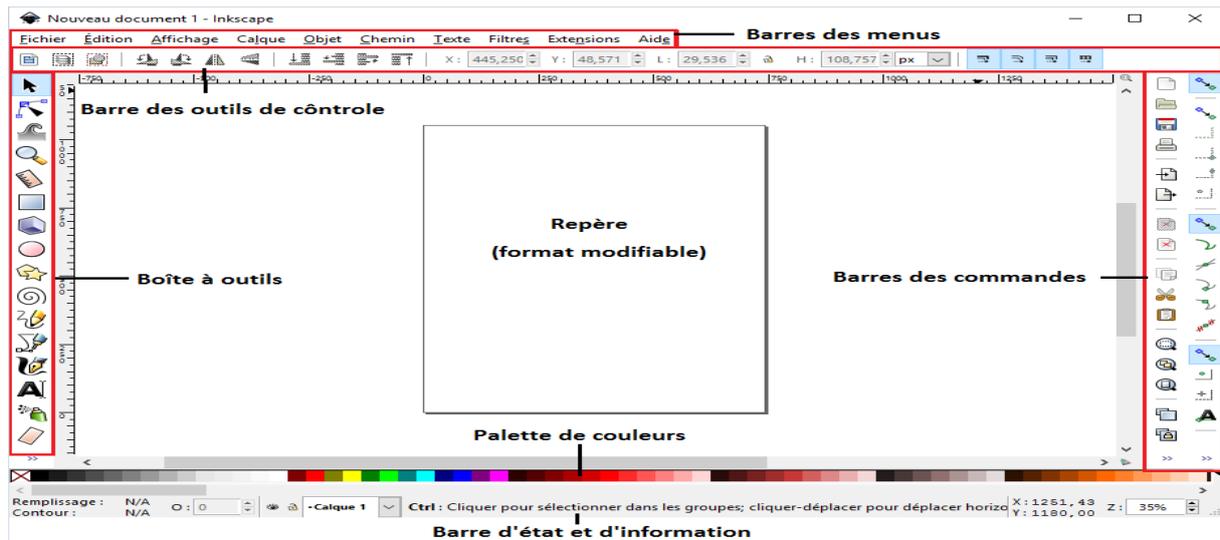
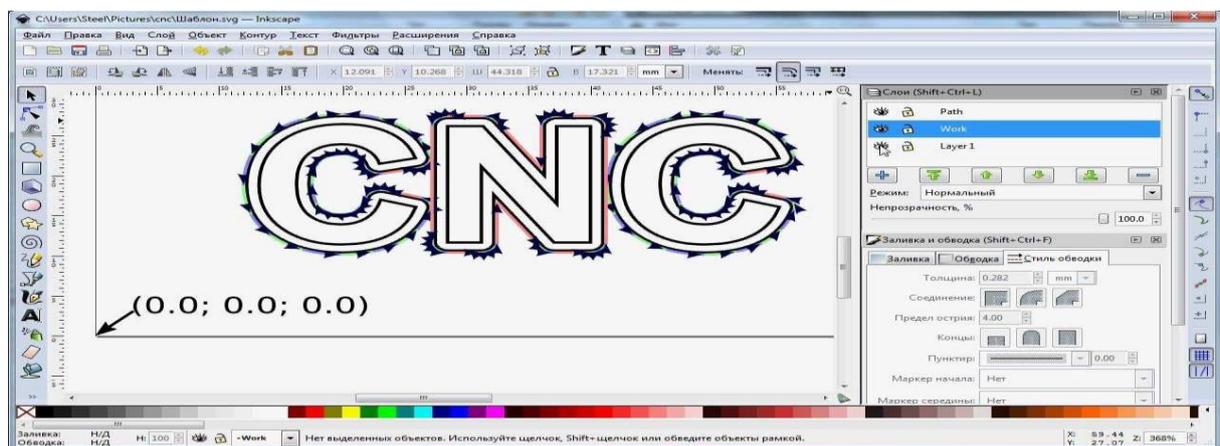


Figure IV.8 :Fenêtre principale(1) .

Dans notre projet, nous allons utiliser ce logiciel pour concevoir les formes que nous voulons dessiner avec une machine à commande numérique(CNC) .



FigureIV.9 :Fenêtre principale(2).

I.5.grbl :

Grbl est un logiciel “firmware”, exécuté sur un Arduino, qui analyse le G-Code reçu, construit un modèle mathématique des actions à effectuer, et exécute ces mouvements en envoyant un flot continu de pulsions haute fréquences aux moteurs pas à pas. Le seul petit détail, c’est que Grbl n’a aucune interface utilisateur, Il faut donc disposer d’un autre logiciel, exécuté sur une autre machine, en liaison avec Arduino, pour pouvoir interagir avec la machine. Il existe plusieurs de ces logiciels, par exempleGrblContrôleur ,X loader.

IV.6.Xloader :

XLoader est un chargeur de classes Java. Il est conçu pour chiffrer les classes Java, décoder les classes Java chiffrés au moment de l’exécution, charger les classes et rechiffrer les classes après l’exécution du programme.

Comment utiliser Xloader pour charger une nouvelle version de grbl dans le contrôleur Arduino / X.

La capture d'écran de gauche montre la fenêtre affichée par Xloader au démarrage.

Entrez le chemin du fichier ou naviguez jusqu'au fichier .hex que vous souhaitez charger dans le contrôleur Arduino / X.

Sélectionnez le type de périphérique dans lequel vous allez charger (Uno / ATmega328 pour le contrôleur Arduino / X).

Sélectionnez le port COM auquel votre contrôleur Arduino / X est connecté.

Le débit en bauds devrait être de 115200.

Appuyez sur le bouton "Upload".

Une fois le fichier .hex chargé dans le contrôleur Arduino / X, la capture d'écran de droite indique la taille du fichier chargé ou un message d'erreur en cas d'échec du téléchargement.

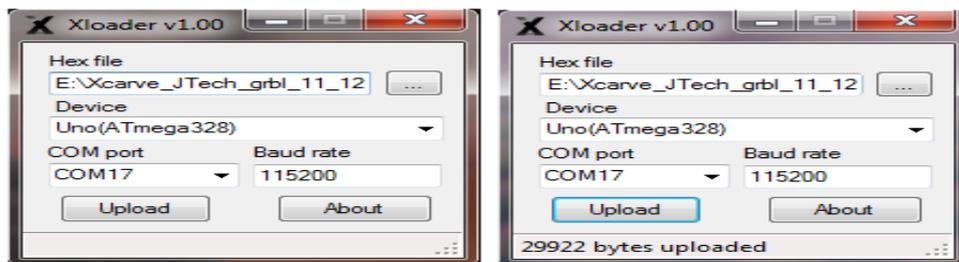


Figure IV.10 : Capture d'écran de logiciel Xloader .

IV.7.conclusion :

Dans ce chapitre, nous avons mentionné les différents logiciels utilisés par la machine numérique pour fonctionner correctement.

Chapitre V : Test de la machine:

V.1.Introduction :

Après avoir passé la définition de CNC et sa conception et la collection de tous les logiciels et fournitures électroniques, nous commençons maintenant à fonctionner, Au début nous lions tous les équipements et images ci-dessous montrant comment câblé la carte Arduino avec les contrôleurs du moteurs(drivers) et les moteurs des 3axes(X,Y,Z).

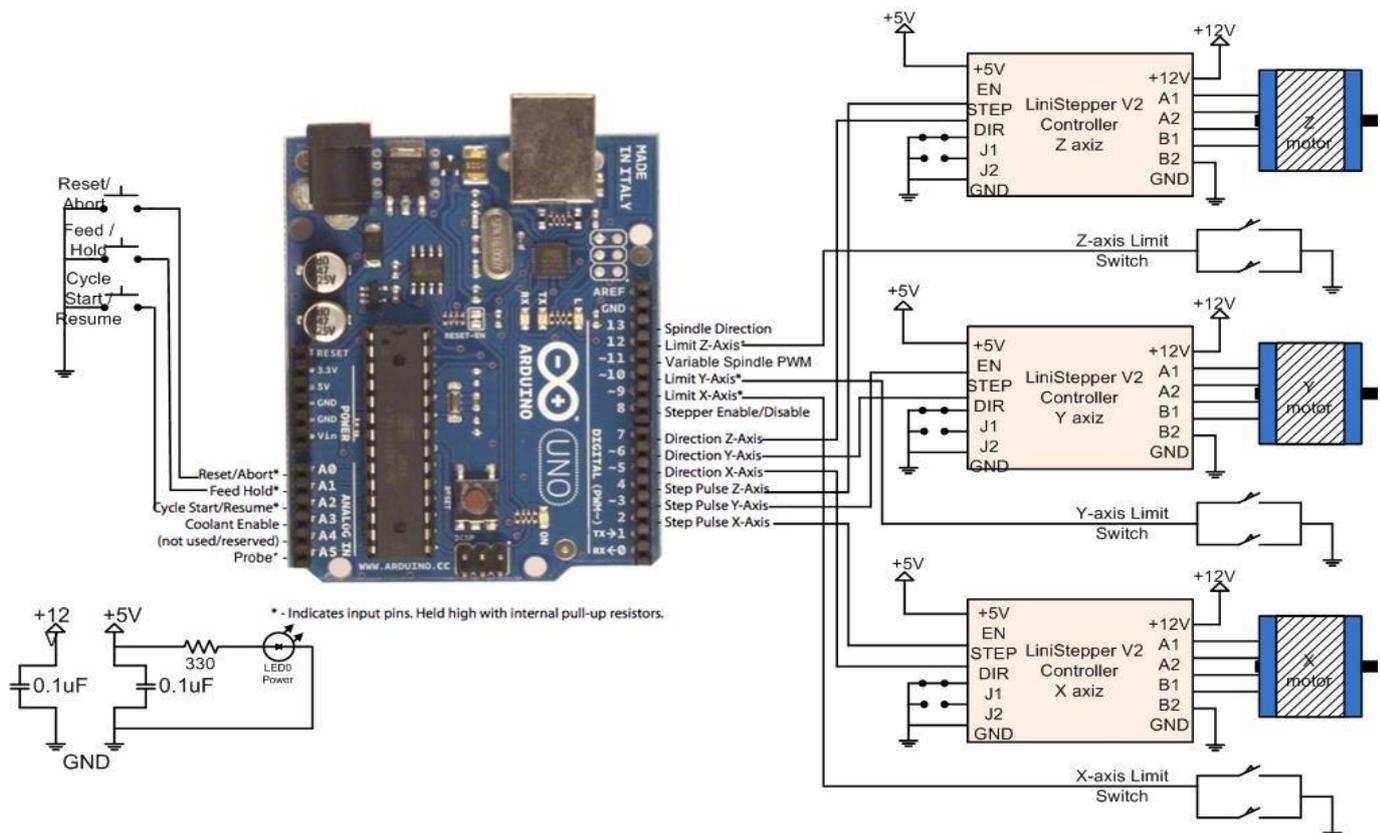


Figure V.1: les équipements de la machine CNC.

PIN02 : impulsion de pas de l'axe X.

PIN03: impulsion de pas de l'axe Y.

PIN04: impulsion de pas de l'axe Z.

PIN05: Direction de l'axe X.

PIN06: Direction de l'axe Y.

PIN07: Direction de l'axe Z.

PIN09 : limite de l'axe X.

PIN10 : limite de l'axe Y.

PIN12 : limite de l'axe Z .

V .2.Mode de fonctionnement :

Il est existé deux modes de fonctionnement :

- Mode manuel
- Mode automatique

V .2.1.mode manuel:

En mode manuel, l'opérateur peut commander les moteurs directement par les boutons (X+,X-,Y+,Y-,Z+,Z-,...) au niveau de logiciel G-code Sender, l'organigramme en bas exprimer les étapes de travail du cet mode :

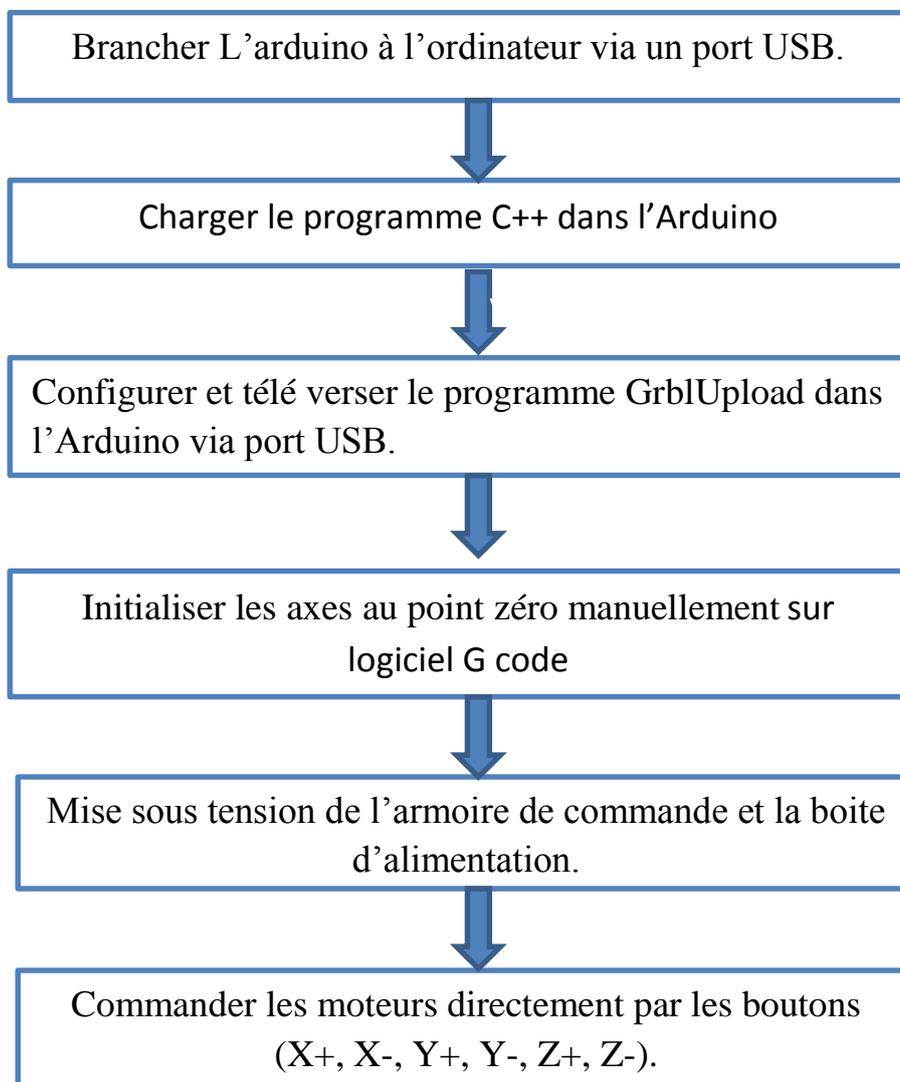


Figure V.2 :organigramme mode manuel.

V .2.1.1.Exemple de commande en mode Manuel :

Sans cette exemple nous avez désigné un carrée avec les dimension suivant (80*80)mm.

- **Brancher L'arduino à l'ordinateur via un port USB.**
- **Charger le programme C++ dans l'Arduino :**

Ce programme est une configuration de moteurs (configuration du directions, accélération, décélération ,limites, max Période, min Période.....etc).

```
Linear_stepper_motor_v2  easyLinearStepper.cpp  easyLinearStepper.h
#include "Arduino.h"

class linearStepper

private:
    // arduino pins for controlling
    unsigned short direction;
    unsigned short enable;
    unsigned short step;
    unsigned short limit1;
    unsigned short limit2;
    |
    // variables
    double travelPerStep;
    unsigned int period;           // period between step in microseconds
    unsigned int acceleration;
    unsigned int minPeriod;
    unsigned int maxPeriod;

    inline void oneStep();
    void accel();
    void decel();
    unsigned int spd2period(double);
```

Figure V.3:programme arduino(1).

```

Linear_stepper_motor_v2  easyLinearStepper.cpp
#include "easyLinearStepper.h"

inline void linearStepper::oneStep()
{
    digitalWrite(step, HIGH);
    delayMicroseconds(5);
    digitalWrite(step, LOW);
    delayMicroseconds(period);
}

void linearStepper::accel()
{
    if(period > minPeriod)
        period -= acceleration;
}

void linearStepper::decel()
{
    if(period < maxPeriod)
        period += acceleration;
}

```

Figure V.4:programme arduino(2).

```

Linear_stepper_motor_v2  easyLinearStepper.cpp  easyLinearStepper.h
digitalWrite(enable, HIGH); // disable board
}

/**
 * move with constant slow speed
 *
 * dir: {0, 1}
 * steps: quantity of microsteps to move, 3200 steps = 1 revolution
 */
void linearStepper::moveConst(const bool dir, double dist)
{
    uint32_t steps = dist/travelPerStep;

    digitalWrite(direction, dir); // set direction
    period = (minPeriod+maxPeriod)/2; // set speed

    digitalWrite(enable, LOW); // enable board
    //~ Serial.println(steps);
    for(uint32_t i=steps; i > 0; i--)
    {
        oneStep();
        //~ Serial.println("step");
    }
    digitalWrite(enable, HIGH); // disable board
}

```

Figure V.5:programme arduino(3).

- **Configurer et télé verser le programme GrblUpload dans l'Arduino via port USB :**

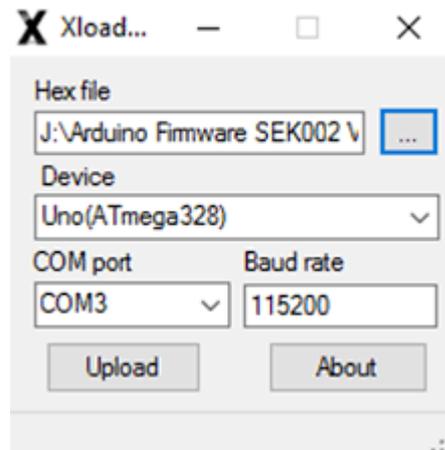
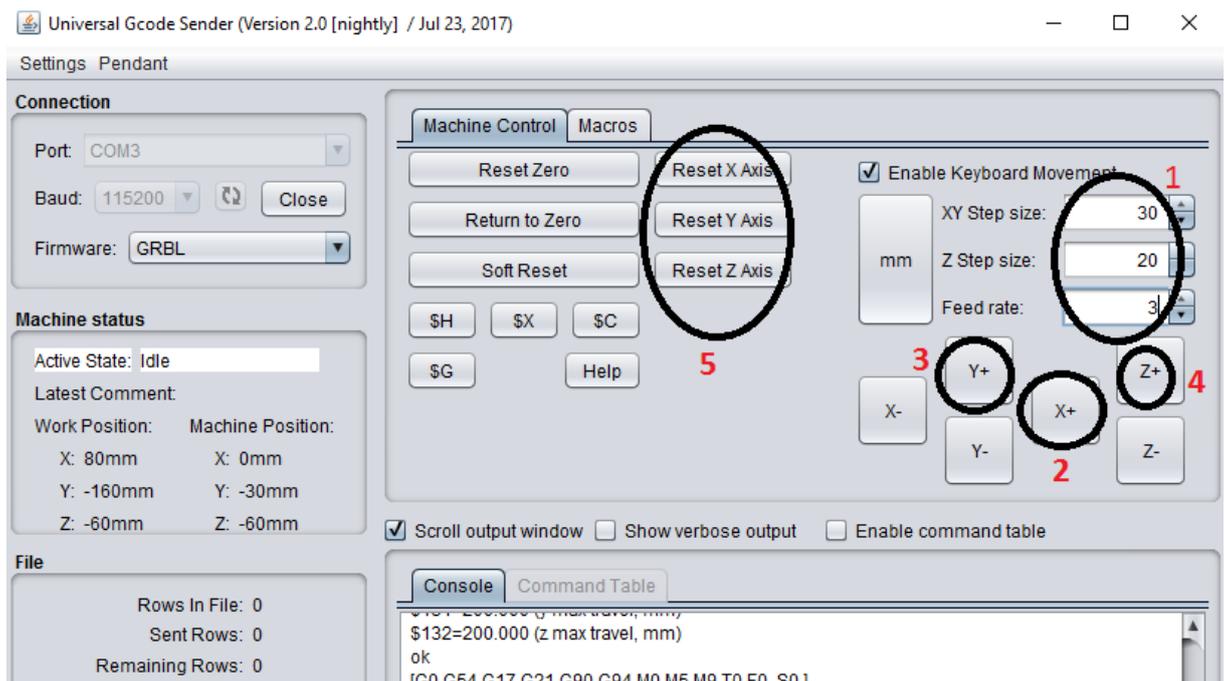


Figure V.6: charger GRBL vers la carte arduino.

- **Initialiser les axes au point zéro manuellement sur logiciel G code :**
 Pour initialisé les 3 axes(X,Y,Z) nous devrions réglé l'origine de système et ont utilisé les boutons (X+, X-, Y+, Y-, Z+, Z-)
 Puis cliquez sur les boutons de reset de chaque axe.



- **Figure V.7 :** Initialiser les axes au point zéro manuellement sur logiciel G code

- **Mise sous tension de l'armoire de commande et la boîte d'alimentation.**
- **Commander les moteurs directement par les boutons (X+, X-, Y+, Y-, Z+, Z-) :**
 D'après terminé les étapes de préparation nous commençons maintenant a désigné notre forme :

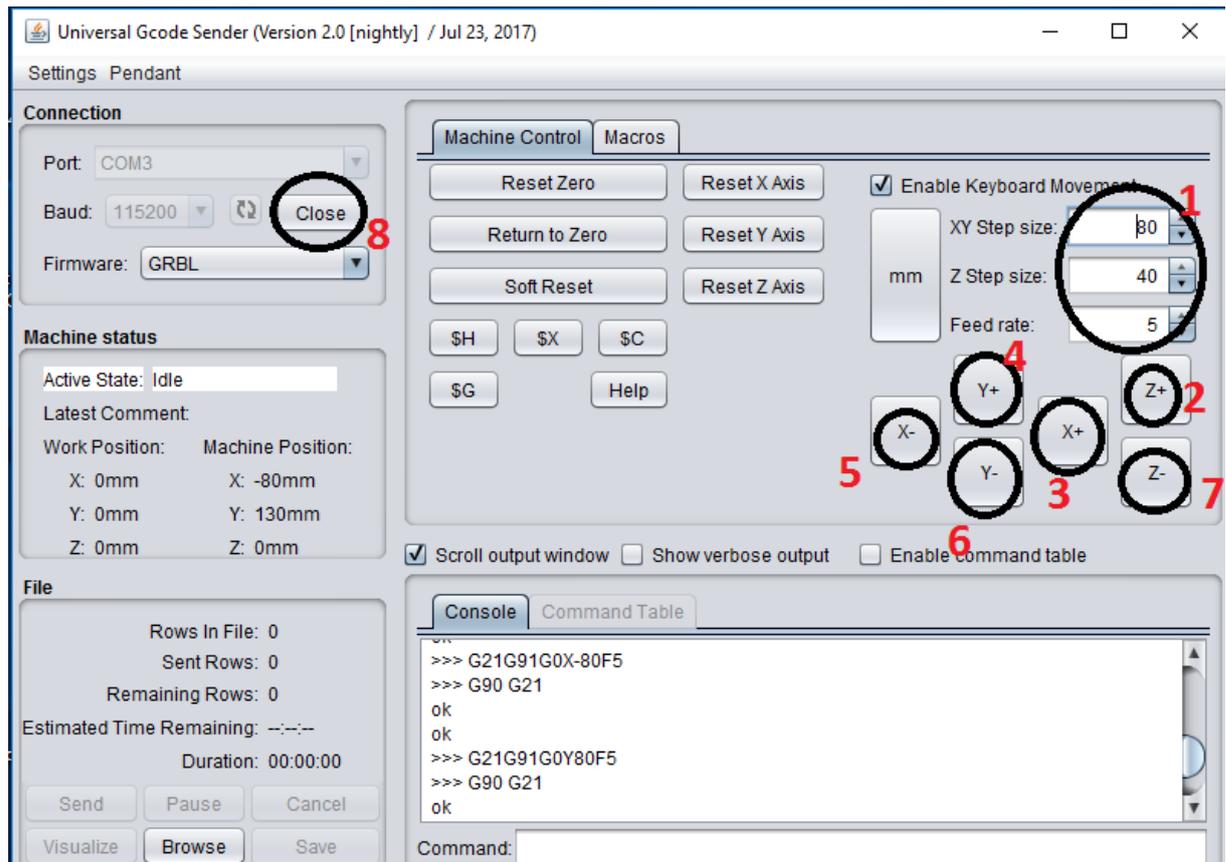


Figure V.8 : Commander les moteurs directement.

1 : entrée les dimension de carrée.

2,3,4,5,6,7 : les boutons pour commander les directions.

8 : une fois terminé la presseuse cliqué sur close pour la sortie.

V .2.2.Mode automatique :

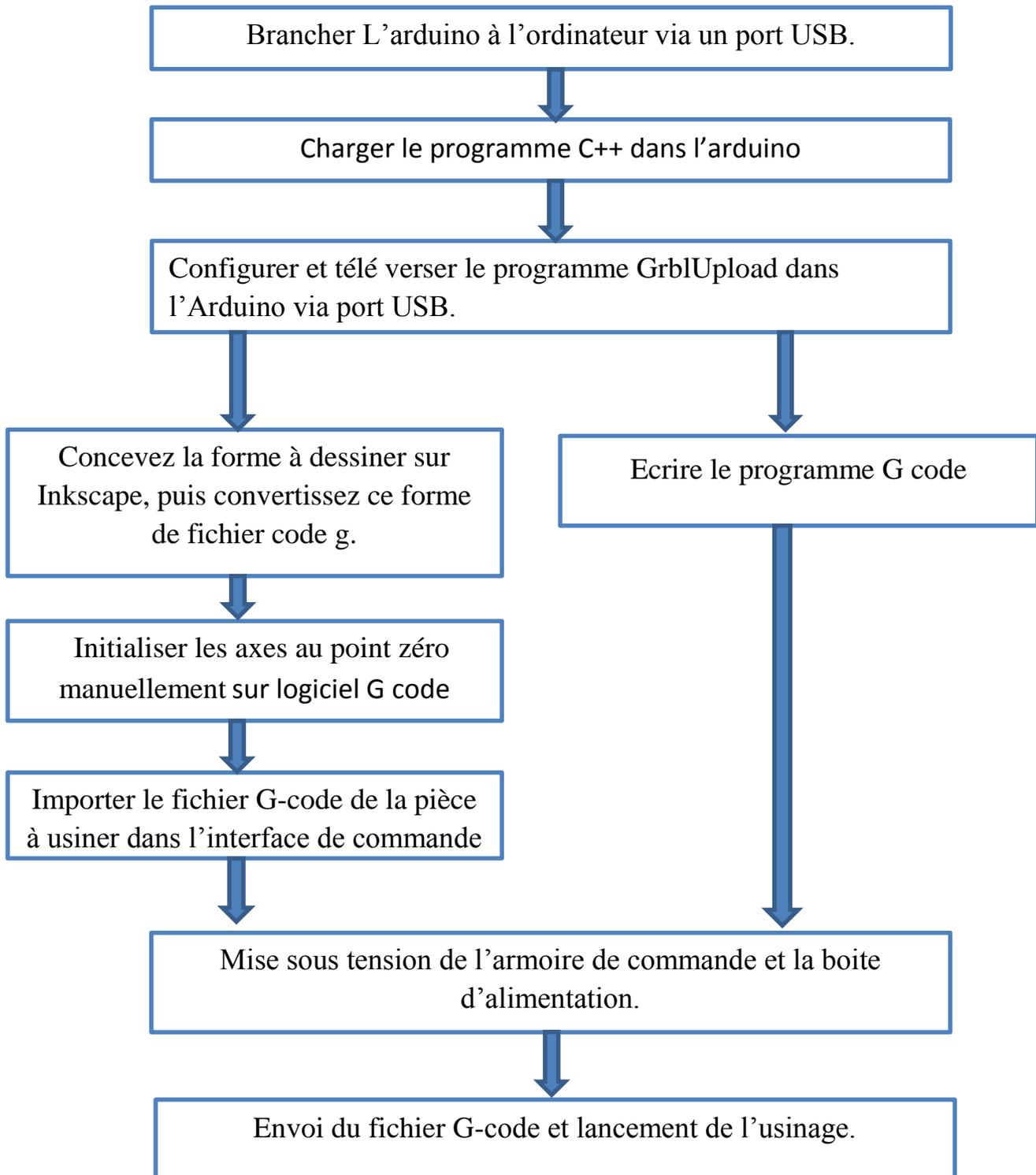


Figure V.9 :organigramme mode automatique.

1) Brancher L'arduino à l'ordinateur via un port USB.

2) Charger le programme C++ dans l'arduino :

On a utilisé le même programme de mode manuel.

3) Configurer et télé verser le programme GrblUpload dans l'Arduino via port USB .

4) Concevez la forme à dessiner sur Inkscape, puis convertissez ce forme de fichier G_code :

Cette extension permet de convertir des chemins obtenus dans Inkscape en langage G-code.

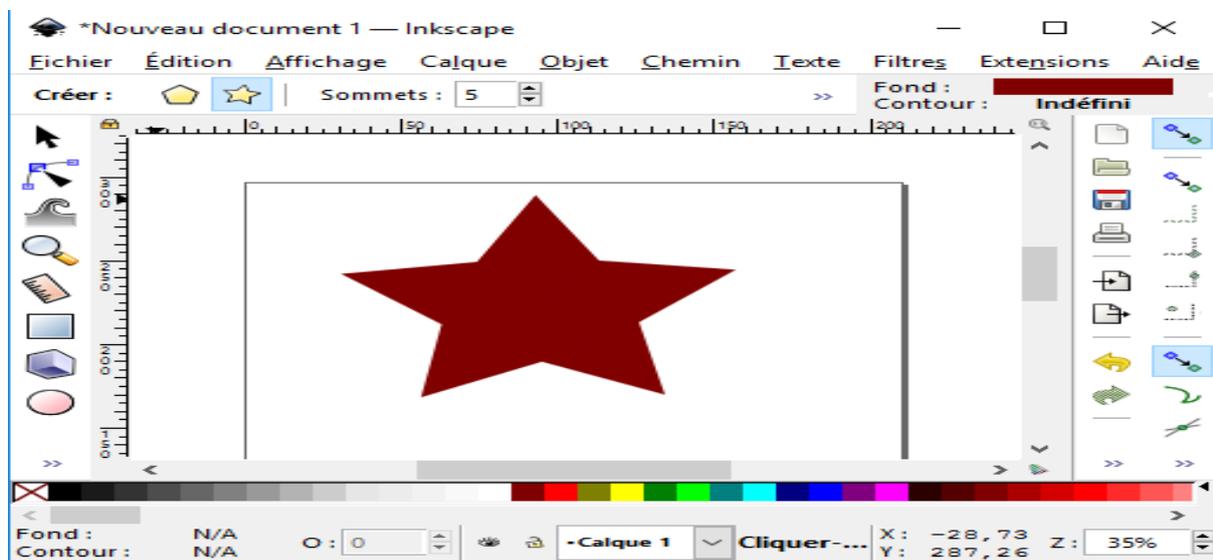


Figure V.10 : Concevez la forme à dessiner sur Inkscape,

Une fois que le fichier est importé ou bien la forme à découper est tracée, on va dans le menu " Extensions → GcodeTools→chemin vers G_code.

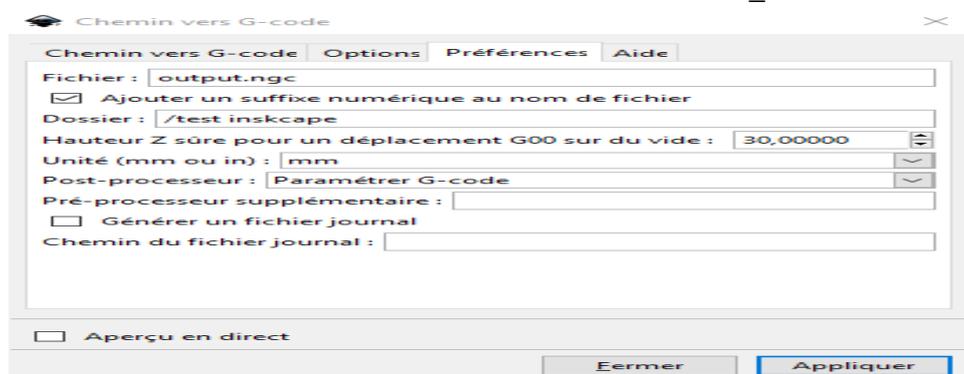


Figure V.11 : convertissez ce forme de fichier G_code :

5) Initialiser les axes au point zéro manuellement sur logiciel G code :



Figure V.12 : Initialiser les axes au point zéro manuellement sur logiciel G code

6) Importer le fichier G-code de la pièce à usiner dans l'interface de commande :

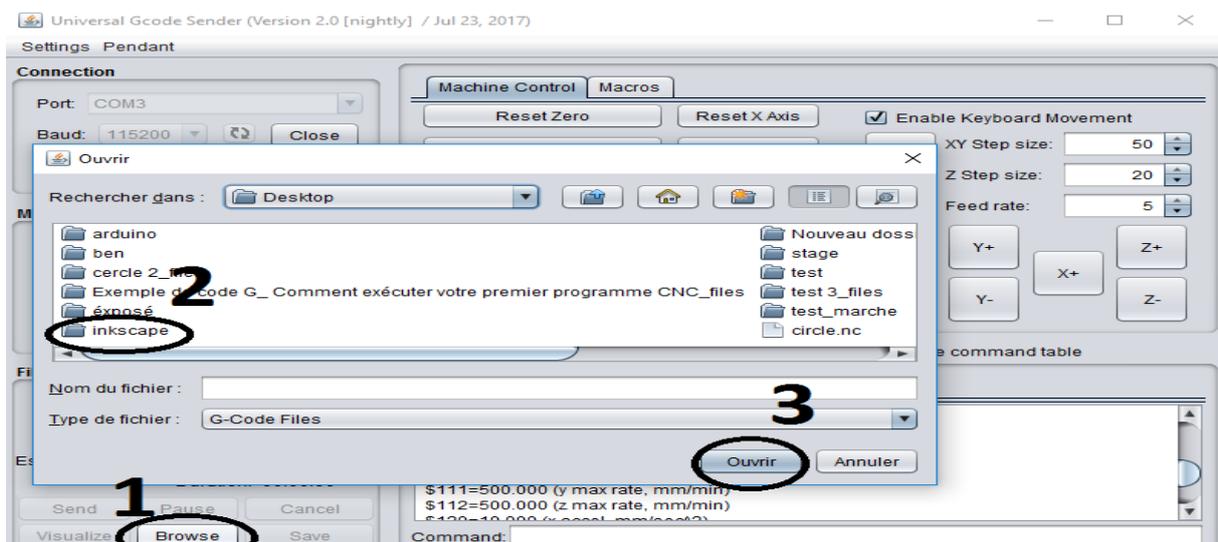


Figure V.13 : Importer le fichier G-code de la pièce à usiner dans l'interface de commande

7) Mise sous tension de l'armoire de commande et la boîte d'alimentation.

8) Envoi du fichier G-code et lancement de l'usinage .

Comme nous l'avons mentionné sur l'organigramme, il existe deux méthodes de contrôle: la méthode indirecte qui est effectuée par le logiciel insckape ou bien la méthode directe consiste à écrire le programme de contrôle directement dans le G-code, et voila un exemple sur cet méthode :

Conception d'un Forme géométrique (carrée) avec les dimensions suivantes (4cm*4 cm) :

```

% // début de programme
G90 G17 G 21 //déplacement en coordonnée absolue, plan( x, y ),programmation en
(mm).
M03 // déplacement en sens horaire
F500 // vitesse de déplacement
G00 Y40 // déplacement rapide (40mm sur l'axe y)
G00 X40 // déplacement rapide (40mm sur l'axe x)
G00 Y-40 // déplacement rapide (40mm sur l'axe y dans le sens inverse)
G00 X-40 // déplacement rapide (40mm sur l'axe x dans le sens inverse))
G00 G43 H1 Z-30 //
G91 G28 Z0 // } déplacement relatives(incrémental) vers les points d'origine
G91 G28 X0 // }
G91 G28 Y0 // }
G90 // déplacement en coordonnée absolue
M30 // Fin du programme, réinitialisation, rembobinage
% // fin de programme
  
```

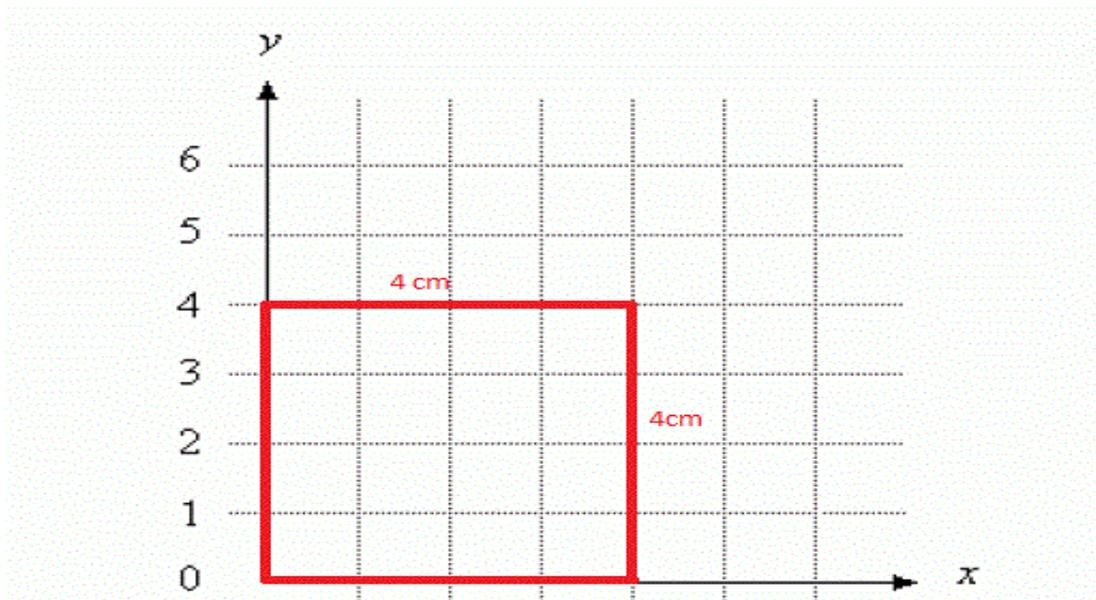


Figure V.14 :exemple de G-code.



Figure V.15 :test réel.

V.3 . Conclusion :

Dans ce chapitre nous avons testé notre machine CNC qui passe a plusieurs étapes et ensuite obtenir le résultat

Conclusion générale :

Ce travail a été développé dans le cadre du projet de fin d'étude, qui nous a permis d'acquérir des connaissances précieuses dans plusieurs domaines (domaine électronique en fabriquant nos propres circuits, en étudiant divers matériels électroniques, et le câblage entre eux , ainsi qu'en domaine de la mécanique et des mesures en étudiant les moteurs pas à pas, et en concevant la machine. Le contrôle numérique, la zone de programmation à travers laquelle nous contrôlons la machine.

Ce projet a également été une occasion précieuse de relier le domaine théorique que nous avons étudié tout au long de la période universitaire au domaine appliqué qui nous mène à lui, et d'acquérir une bonne expérience au niveau de la perception pratique.

Après tout ce que nous avons fait, cette machine doit encore être développée soit dans la partie contrôle, soit dans la partie pratique

Je vais donc continuer dans ce domaine pour atteindre un meilleur résultat obtenu.

Bibliographié :

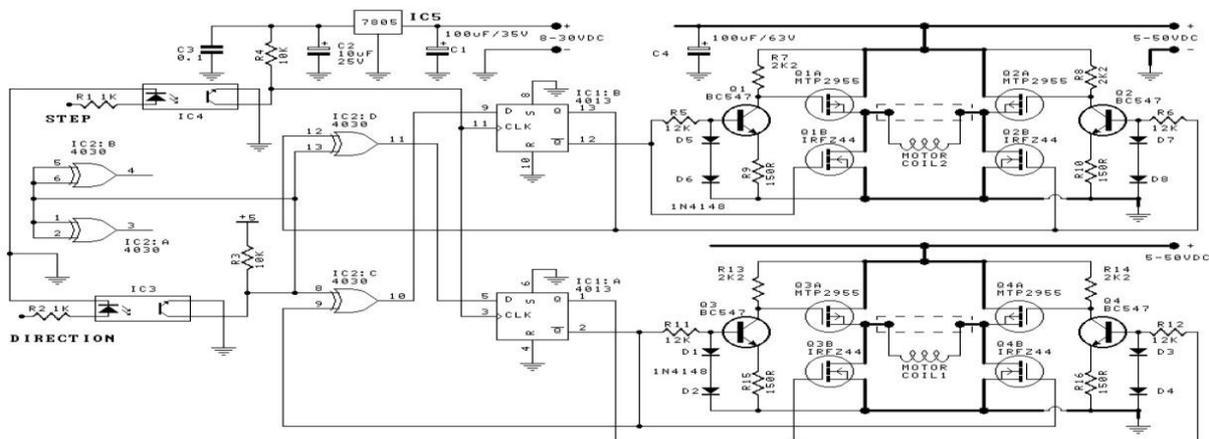
- 1/SINUTRAIN 828D ShopMill) Rapport de Projet de Fin d'Études*
- 2/Build Your Own CNC Machine(livre)*
- 3/Fundamentals of CNC Machining livre)*
- 4/James Floyd Kelly, Patrick Hood-Daniel - Build Your Own CNC Machine (Technology in Action)-Apress (2009)*
- 5/Make - Design for CNC - Furniture Projects and Fabrication Technique(livre).*
- 6/Mr. MAHDJOURI Abdelfettah (Etude et réalisation d'une machine outil à commande*
- 7/numérique (fraiseuse 3 axes)). Rapport de Projet de Fin d'Études*
- 8/www ,technologuepro ,com*
- 9/http://www.evxonline.com*
- 10/ https://forum.pololu.com*
- 11/www .robotshop .com*
- 12/https://www.dfrobot.com*
- 13/https://www.amazon.fr*
- 14/http://www.arduino.cc*
- 15http://arduino.blaisepascal.fr*
- 16/https://fr.wikipedia.org*
- 17/https://docplayer.fr*
- 18/https://wiki.mdl29.net*
- 19/https://zestedesavoir.com*
- 20/https://github.com*
- 21/https://docplayer.fr*
- 22/http://www.qhunt.com*
- 23/https://www.quora.com*
- 24/Bentalebmokhtargasmiehladi (Réalisation et commande d'une machine cnc à base*
- 25/des moteurs pas à pas) Rapport de Projet de Fin d'Études*
- 26/BOUANIK FOUAD (Simulation de l'usinage d'un guidage longitudinal sous le*
- logiciel*
- 27/MOUZAOUI Melissa. TAZAMOUCHE Yanis.(Réalisation et automatisation d'une*
- 28/machine à comma Rapport de Projet de Fin d'Études IMA5nde numérique) Rapport de*
- 29/Projet de Fin d'Études*

Annexes :

Annexe1 :réalisation dunotre circuit imprimé (contrôleur de moteur):

1 : les étapes de réalisation du circuit imprimé :

1ere étape : Apporter le circuit électronique qui composé par un séquenceur (la partie qui communiqué avec le microcontrôleur et deux Pont H (A, B).

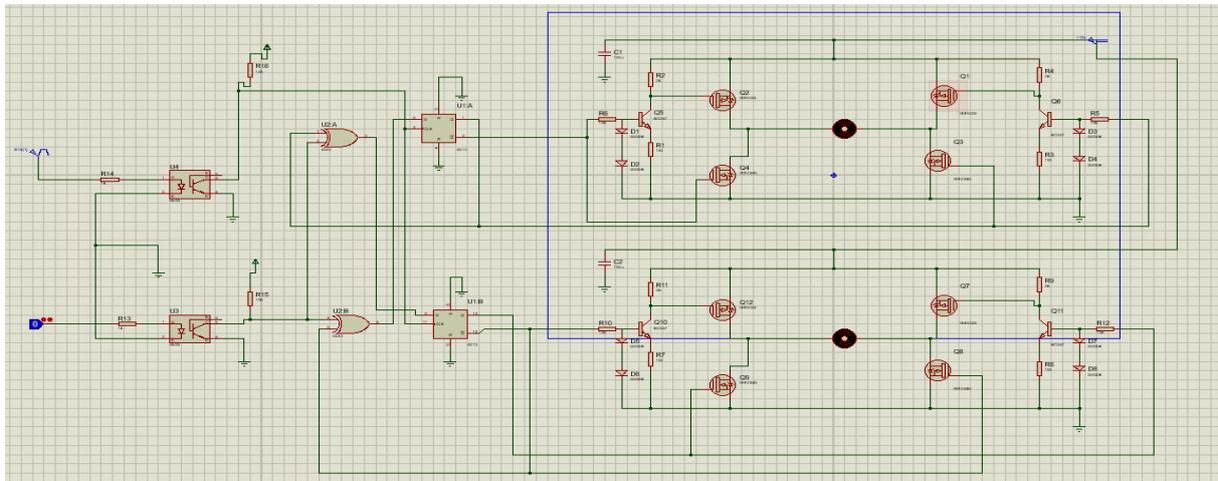


circuit électronique de contrôleur de moteur .

2ème étape : simulation du circuit par logiciel proteus (logiciel de conception de circuits imprimés et outils de simulation intégrés pour le développement EDA moderne. Proteus Design Suite associe facilité d'utilisation et fonctionnalités puissantes pour permettre la conception, le test et la mise en page rapides de circuits imprimés professionnels)



interface de logiciel proteus .

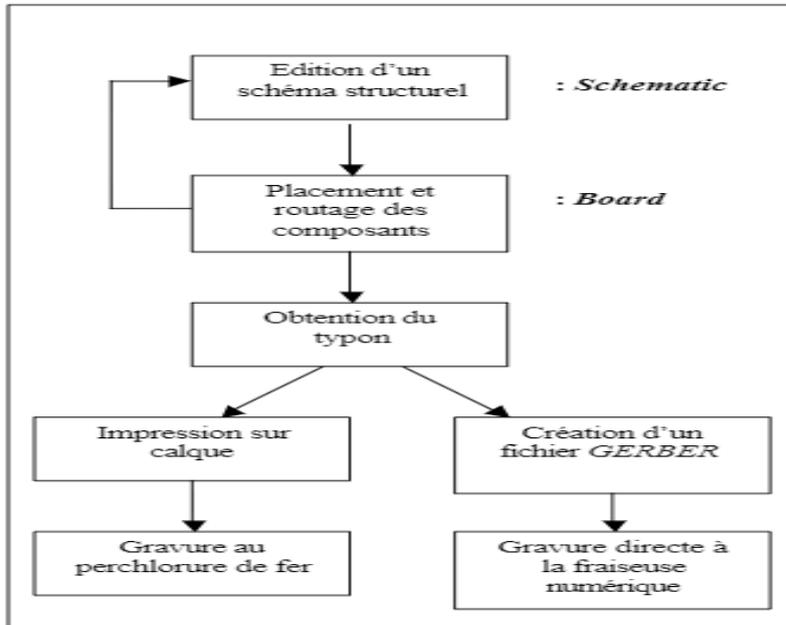


simulation du circuit par logiciel proteus.

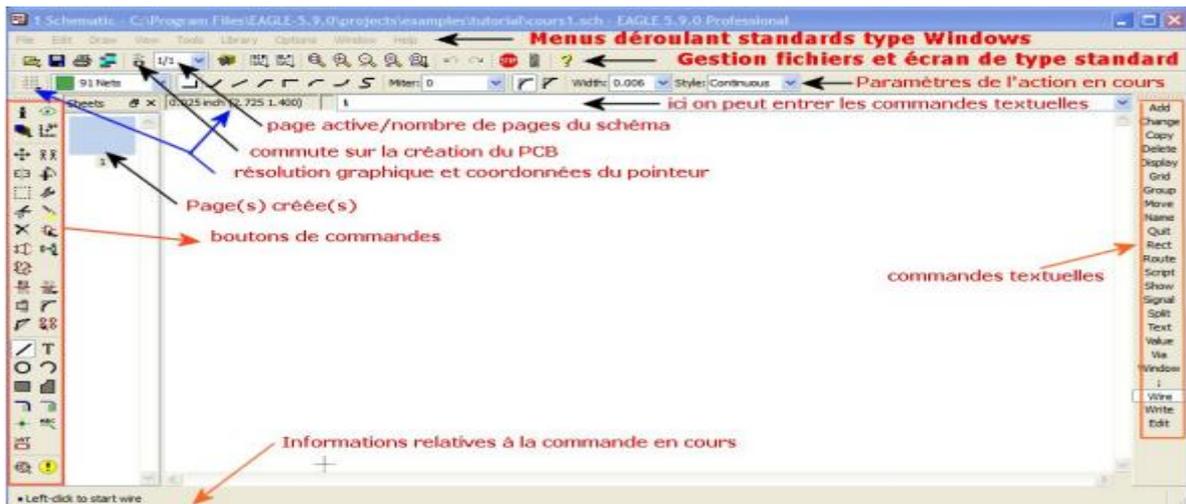
Rq :

- 1) Nous avons divisé le circuit en trois parties, partie(1) le séquenceur, partie(2) le pont H(A), partie(3) le pont H(B).
- 2) Le principe de fonctionnement de pont H(A) lui-même de pont H(B), Nous allons donc créer deux versions du partie pont H (A).

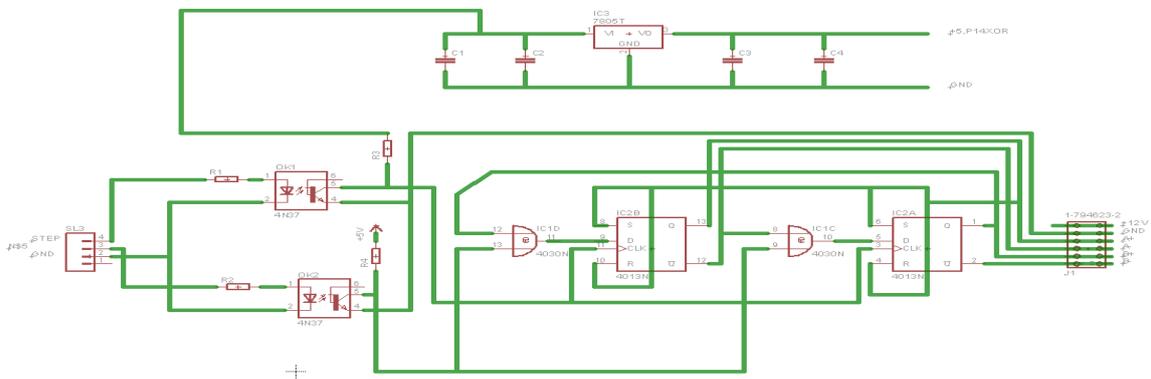
3^{ème} étape : Faire le circuit par logiciel eagle (Easily Applicable GraphicsLayout Editor) est un logiciel fourni par la société CADSOFT, qui permet la réalisation de typons en vue de construire une carte électronique.



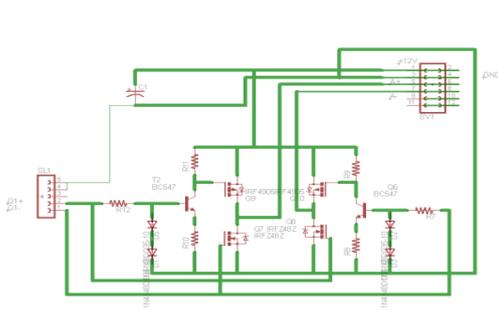
organigramme montre les étapes de circuit par logiciel eagle.



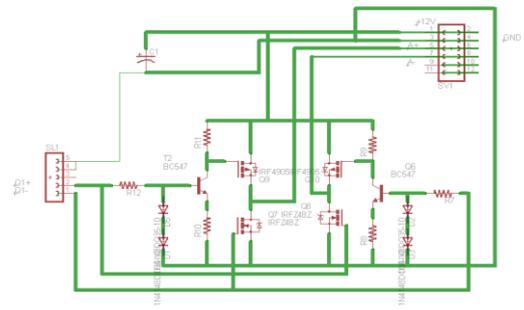
Fenêtre de logiciel eagle .



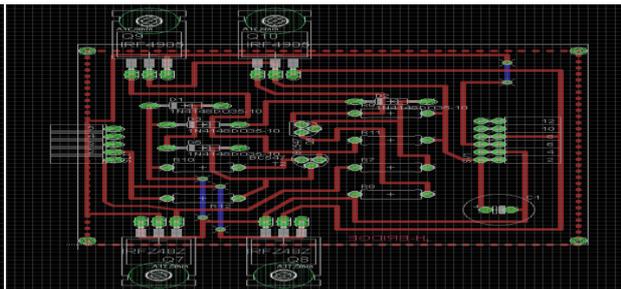
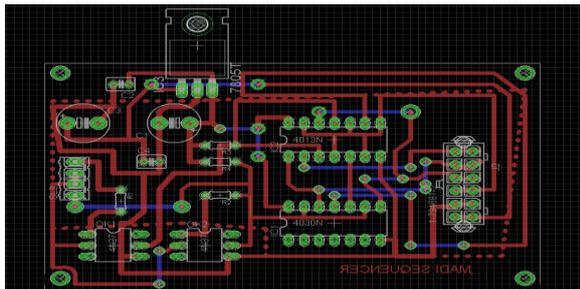
Le schématique du séquenceur.



.Le schematique duPont H(A)

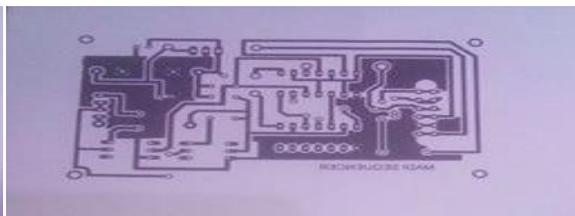
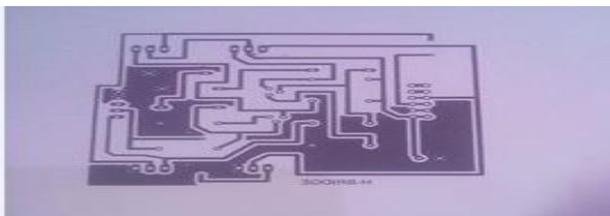


Le schematiquePont H(B) .



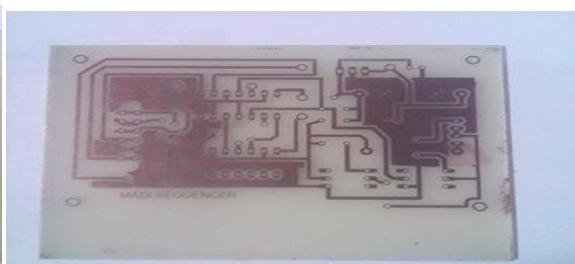
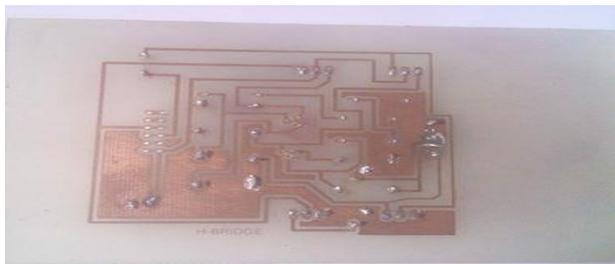
PCB du séquenceur PCB pont H (A ,B).

4ème étape : imprimé les circuits sur une faille transparente.



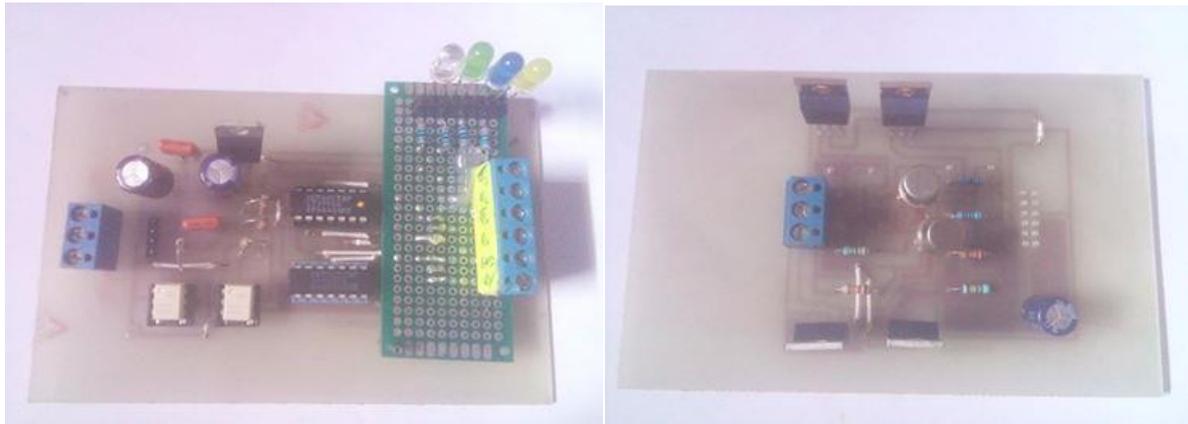
circuits sur une faille transparente.

5 ème étape : Le circuit est imprimé sur la carte de cuivre



circuits imprimés sur les cartes de cuivre .

6 ème étape : Soudage des différents composants



Placements des différents composants

les composants utilisés : le tableau en bas au montrée les composants qui on utilisé

Noms des composants	Quantité	Valeurs
Transistors Mosfet (irf 540 n)	4ps	100V/30A
Transistors Mosfet (irf9z34n)	4ps	55V/19A
Transistors 8007	4	45V /0.1A
Diode schottky	4	
Condensateurs	1	100uf / 63V
Condensateurs	1	0.1 uf
Condensateurs	1	10uf /25V
Condensateurs	1	100uf/ 35V
Resistances	4	12k Ω
Resistances	4	2k2 Ω

Resistances	4	150Ω
Resistances	4	330Ω
Resistances	2	10KΩ
Resistances	2	1KΩ
Port logique OU EXCL (4030)	1	14 pates
Bascule IC (4013)	1	14 pates
Optocoupler (4N35)	2	30V
Régulateur de tension(7805)	1	In 12v /out 5V
Borné 3 pates	3	/
Borné 6pates	3	/
Ledscoeurs	4	3V

Tableau des composants qui onutilisé.

Note : Nous n'avons pas utilisé ce circuit Par manque de quelques composants.

Annexe 2 :Tableau des fonctions préparatoires G code :

G00	Déplacement rapide
G01	Interpolation linéaire
G02	Interpolation circulaire (sens horaire)
G03	Interpolation circulaire (sens anti-horaire)
G04	Arrêt programme et ouverture carter (pour nettoyer) (temporisation - suivi de l'argument F ou X en secondes)
G10/G11	Écriture de données/Effacement de données
G17	Sélection du plan X-Y
G18	Sélection du plan X-Z
G19	Sélection du plan Y-Z
G20	Programmation en pouces
G21	Programmation en mm
G28	Retour à la position d'origine
G31	Saute la fonction (utilisé pour les capteurs et les mesures de longueur d'outil)
G33	Filetage à pas constant
G34	Filetage à pas variable
G40	Pas de compensation de rayon d'outil
G41	Compensation de rayon d'outil à gauche
G42	Compensation de rayon d'outil à droite
G90	Déplacements en coordonnées absolues
G91	Déplacements en coordonnées relatives
G94/G95	Déplacement en Pouces par minute/Pouce par tour
G96/G97	Vitesse de coupe constante (vitesse de surface constante)/Vitesse de rotation constante ou annulation de G96

tableau IV.2 : fonctions préparatoires G code .

Codes M:

M03, M04, M05 Broche sens horaire, sens antihoraire, arrêt de broche

M21 Activation pression broche /M121 à M127 Palier pression selon outils

M07,M08, M09 Mise en route arrosage/ arrêt arrosage

M02 Arrêt du programme

M30 Fin du programme, réinitialisation,rembobinage

M99 Fin du sous-programme

M00, M01 Arrêt du programme, arrêt optionnel ou avec condition

M06 Changement d'outil.

autre code :

Les codes

- Fonctions préparatoires (G), fonctions d'appel de mode d'interpolation (G 0), cycles machine
- Coordonnées de points (X, Y, Z, I, J, K)
- Vitesses, avances... (S, F)
- Fonctions auxiliaires (M) qui permettent d'enclencher la lubrification, de changer d'outil, ou de déclencher des accessoires.

X : position absolue

Y : position absolue

Z : position absolue

A : position (rotation autour de l'axe X)

B : position (rotation autour de l'axe Y)

Programmation de commande numérique 2

C : position (rotation autour de l'axe Z)

U : position Relative OU axe secondaire

V : position Relative OU axe secondaire

W : position Relative OU axe secondaire

M : code Fonction "Machine" ou "Modale"(un autre type d'action ou de code machine(*))(parfois référencé

comme fonction 'diverse' ("Miscellaneous" en anglais)

Permet aussi de contrôler des entités externes à la machine proprement dit (magasin auxiliaire, refroidisseur,

compteur ETC)

D : association d'un correcteur de jauge outil

F : vitesse de déplacement

S : vitesse de rotation

C : s'il est intégré dans un bloc de cycle fixe ébauche ou finition, C désigne un chanfrein

N :numéro de ligne

Le numéro de bloc n'est pas obligatoire à chaque ligne avec les commandes ISO Fanuc, il désigne juste un endroit du

programme où l'on peut ordonner un saut d'opération ou un départ d'opération, le plus souvent avant un changement d'outil

V : permet de contrôler une vitesse de rotation dans une machine avec diverses configurations

R :Rayon d'arc ou option passée à un sous programme

P :Temps de pause ou option passée à un sous programme ou appel de sous programme (Précédé de M98 par ex)

T Sélection d'outil

I :Axe X des données d'un arc

J : Axe Y des données d'un arc

K : Axe Z des données d'un arc

D :diamètre de coupe/décalage pour l'épaisseur d'outil

H : décalage pour la hauteur de l'outil

Origines :

- **Origine programme (OP) :** c'est le point origine du programme à partir duquel les mouvements de la machine sont programmés.

- **Origine Machine (OM):** Cette expression existe dans des manuels de constructeurs de machines-outils sans pour autant apporter de réponse, voire sans fondement. L'origine machine est une coordonnée mesure particulière.

Lors de l'initialisation de l'axe (dans le cas de capteurs relatifs) au passage du top zéro de la règle, le processus d'initialisation permet de forcer la valeur du registre du point courant par la valeur de l'axe dans le registre P16 (cas d'un DCN NUM). Cette valeur est souvent non nulle qui remet en cause la notion d' « origine » d'une part, et

« machine » d'autre part puis que c'est une coordonnée mesure particulière. Cette expression est sans fondement

dans le cas de technologie de règle absolue, et sans fondement tout court.

• **Origine mesure (Om)** : L'origine mesure est propre à chaque axe asservi. Dans le cas d'une structure articulaire

de type RRPPP (rotoïde, rotoïde, prismatique, prismatique, prismatique) il existe 5 origines mesures. Chaque axe

mesure est constitué d'une origine et d'une dimension. La dimension de l'espace vectoriel de cette structure est de

dimension 5. Suivant le type de technologie des capteurs permettant l'asservissement des axes, il est nécessaire de

procéder à l'initialisation de la partie opérative avec la partie commande. Dans le cas de capteur relatif, les POM

(Prise d'origines mesure) servent à établir les références de la mesure sur chaque axes des machines outils ne

possédant pas de règles avec des capteurs absolus (détection du zéro du capteur de mesure).

Les origines mesures appartiennent à l'espace articulaire (espace de la structure cinématique de la machine en

robotique). L'espace travail, ou de la tâche est celui où se trouve les autres éléments de la cellule élémentaire de

production. La dimension vectorielle est 3. Il y a donc une endomorphisme d'espaces vectoriels entre l'espace

articulaire de la machine et l'espace travail. Il est donc totalement inutile de représenter le zéro mesure comme un

point concourant de tous les axes dans l'espace travail. C'est un sophisme.

• **Origine porte pièce (Opp)** : C'est le point caractéristique de la liaison encastrement supposée parfaite entre la

machine et le porte-pièce. En tournage on le place souvent à l'intersection de la face avant du mandrin et de l'axe

de la broche (axe Z, pour les mandrins qui ne sont pas changés régulièrement). En fraisage pour des raisons de

standardisation on alèse des centreurs sur les tables des machines outils pour le situer plus facilement.

Programmation de commande numérique 6

• **Origine pièce (Op)** : (appelé G... 54 par exemple, G55,etc.) C'est le point d'intersection de l'isostatisme. Ce point

situe la pièce par rapport au porte-pièce.

• Distance origine programme (OP) - origine machine (OM) : c'est la distance que la machine doit additionner pour

passer de son origine (OM) à l'origine du programme (OP).

• "DECALAGE" (dec) : distance vectorielle de l'origine porte-pièce (Opp) à l'origine programme (OP).

• "PREF" (pref) : distance vectorielle de l'intersection des origines mesure de chaque axe qu'on appelle souvent

Origine mesure pour simplifier (Om) à l'origine porte-pièce.

$(POM \text{ ou } OM) + PREF = OPP$ $OPP + DECALAGE = OP$