

**République Algérienne Démocratique et Populaire Ministère de
l'Enseignement Supérieur et de la Recherche Scientifique
Université A.MIRA-BEJAIA**

**Faculté de technologie
Département de Génie Électrique**



جامعة بجاية
Tasdawit n Bgayet
Université de Béjaïa



MEMOIRE DE FIN DE CYCLE

Présenté en vue de l'obtention du diplôme :

MASTER EN ELECTROTECHNIQUE

Spécialité : AUTOMATISMES INDUSTRIELS

Thème

**Réalisation et automatisation d'une machine à commande
numérique**

Présenté par :

- **MOUZAOUI** **Melissa.**
- **TAZAMOUCHT** **Yanis.**

Promoteur

- **Mr. TAIB** **Nabil** **M.C.A**

REMERCIEMENTS

Nous souhaitant adresser ici nos remerciements les plus sincères aux personnes qui nous ont apporté leur aide et qui ont contribué à l'élaboration de ce mémoire ainsi qu'à la réussite de cette formidable année universitaire.

Ces remerciements vont tout d'abord au corps professoral et administratif du département « Génie électrique », pour la richesse et la qualité de leur enseignement et qui déploient de grands efforts pour assurer à leurs étudiants une formation actualisée.

Nous tenant à remercier sincèrement Monsieur **TAIB Nabil**, maître de conférence à la faculté Technologie, qui, en tant que Directeur de mémoire, s'est toujours montré à l'écoute et très disponible tout au long de la réalisation de ce mémoire, ainsi pour l'inspiration, l'aide et le temps qu'il a bien voulu nous consacrer et sans qui ce mémoire n'aurait jamais vu le jour.

Nos remerciements s'adressent aussi à toute l'équipe de Denzer Technologies, en premier lieu, Monsieur **CHIKHI Mohand cherif** et **RACHIDIOU Lounis** pour leur aides pratiques, leurs conseils précieux et leurs encouragements

On remercie également le président de jury, de nous avoir honorés en acceptant de présider le jury.

Aussi, nous remercions L'examineur, pour avoir accepté d'examiner ce mémoire.

On n'oublie pas nos parents pour leur contribution, leur soutien et leur patience. Enfin, nous adressons nos plus sincères remerciements à tous nos proches et amis, qui nous ont toujours encouragés au cours de la réalisation de ce mémoire.

Merci à toutes et à tous.

DEDICACES

Je dédie ce modeste travail :

A mes très chers parents

*Pour leurs soutiens et leurs sacrifices. Que dieu leurs procure
bonne santé et longue vie.*

A ma sœur et mes frères

Yasmina, Kamal, Hakim

A ma belle-sœur, ma nièce et mon neveu

Fouzia, Anya, Aris

A la mémoire de ma très chère grand-mère

A tous mes amis

Imene, Fouzia, Lydia, Bariza

A toutes ma famille

A mon binôme TAZAMOUCHE Yanis

MOUZAOUI Melissa

DEDICACES

Je dédie ce modeste travail :

- *A Ma très chère Mère qui a toujours été à mes côtés ;*
- *A Mon Père qui m'a toujours guidé et soutenu ;*
- *A mon frère Arris et ma sœur Lydia pour leur soutien moral ;*
- *A tous mes amis avec lesquelles j'ai partagé les meilleurs moments de ma vie, plus particulièrement, RAFAA Imane, LASMI Idir, ZEBOUDJ Sofia, ISSAAD Imane ;*
- *Sans oublier ABID Nassim, KERKOUR Koussaïla,*
- *KATIR Yacine, MOUZAOUI Abdelkader, MENDIL Reda et BELHADAD Walid ;*
- *A la mémoire de ma tante et mes grands-parents ;*
- *A mes oncles et mes cousins et à toute la famille TAZAMOUCHT ;*
- *A tous ceux qui m'ont aidé de près ou de loin*
- *A ma très chère binôme MOUZAOUI Melissa, sans qui, ce travail n'aurait pas été possible.*

TAZAMOUCHT Yanis

RESUME

Les machines à commande numérique sont devenues des moyens de productions incontournables dans l'industrie. Elles permettent des cadences de production importantes et facilitent l'obtention de surface complexe. Pour cela, il est nécessaire d'utiliser des systèmes d'aide informatique spécialisé pour la conception et la fabrication assisté par ordinateur.

Dans notre travail, nous avons essayé de donner quelque notion fondamentale sur les machines à commande numérique, des moteurs pas à pas et une présentation de la CAO/FAO.

Une autre partie est consacrée au principe de fonctionnement de la machine DENZER-CNC. Ainsi, que les différents logiciels agissant sur la commande numérique.

Dans la dernière partie, nous présentons la simulation de la commande de trois moteurs pas à pas sur le logiciel ISIS PROTEUS. Ainsi, qu'une mise en marche de la machine DENZER-CNC.

Mots clés : Machine CNC, commande numérique, automatisme, électronique, grbl, g-code, moteur pas à pas.

ABSTRACT

Numerical Control Machines are today considered as an important means of production in all the industries. They not only increase the production rate but they also allow us to easily obtain complex surfaces. That is why, it is necessary to provide specialized information technologies in the design, analysis, and manufacture of products (Computer-aided technologies).

In our thesis we aimed at providing some basic notion about CNC machines, stepper motors as well as to present CAO/FAO. Our paper is divided into three parts, the first part deals with some generalities about the numerical control machines, whereas the second part is about

the operating principle of the DENZER-CNC machine as well as the different software that work on the numerical control. In the last part however, we present a simulation of the way the three stepper motors' are controlled on the ISIS PROTEUS software and we operate the machine DENZER-CNC.

Key words: Machine CNC, Numerical control, electronic, GRBL,G-code, Automatic, Stepper motors.

TABLE DES MATIERES

Remerciements

Dédicaces

Résumé

Table des matières

Liste des figures

Liste des tableaux

Introduction générale	1
Chapitre I : Généralité sur les machines à commande numérique.	
I.1 INTRODUCTION	3
I.2 HISTORIQUE DES MACHINES-OUTILS.....	3
I.3 LA COMMANDE NUMERIQUE	4
I.3.1 Origines de la commande numérique.....	4
I.3.2 Principe de la commande numérique	4
I.4 LA STRUCTURE D'UNE MACHINE A COMMANDE NUMERIQUE	5
I.4.1 Partie opérative.....	6
I.4.1.1 Tâches effectuées.....	6
I.4.1.2 Armoire électrique de relaying ou armoire de puissance	7
I.4.2 Partie commande	7
I.5 PRINCIPE DE FONCTIONNEMENT D'UNE MACHINE A COMMANDE NUMERIQUE.....	8
I.6 LES FAMILLES DES MACHINES A COMMANDE NUMERIQUE.....	8
I.6.1 Fraisage à commande numérique (FCN)	8
I.6.2 Tournage à commande numérique (TCN)	9
I.6.3 Rectification à commande numérique	9
I.6.4 Electro-érosion à commande numérique.....	10
I.7 CLASSIFICATION DES MOCN	10
I.7.1 Classification des MOCN selon le mode de fonctionnement	11
Les MOCN sont classifiées selon les modes de fonctionnement suivants :.....	11
I.7.1.1 Fonctionnement en boucle ouvert.....	11
I.7.1.2 Fonctionnement avec commande adaptative	11
I.7.1.3 Fonctionnement en boucle fermée.....	12
I.7.2 Classification des MOCN selon le nombre d'axe	13
I.7.3 Classification des MOCN selon le mode d'usinage.....	13
I.8 LA MACHINE A COMMANDE NUMERIQUE A TROIS AXES	13

I.9 LES MOTEURS PAS A PAS	14
I.9.1 Les différents types de moteur pas à pas	14
I.9.1.1 Les moteurs bipolaires	14
I.9.1.2 Les moteurs unipolaires	14
I.9.1.3 Les moteurs à réluctance variable.....	15
I.9.1.3.1 Le principe de fonctionnement	15
I.9.1.4 Les moteurs à aimant permanent	15
I.9.1.4.1 Le principe de fonctionnement	15
I.9.1.5 Les moteurs hybrides	16
I.9.1.5.1 Principe de fonctionnement	16
I.9.2 Comparaison des trois catégories des moteurs pas à pas	17
I.10 LA COMMANDE DES MOCN	17
I.10.1 La conception 3D	18
I.10.2 La génération des chemins de l'outil, création du G-code	18
I.10.2.1 Langue de contrôleurs numériques « G-code »	18
I.10.3 Interprétation du G-code et transformation en mouvement sur la machine	19
I.10.3.1 Description du micro-logiciel Grbl.....	20
I.11 LA CONCEPTION ASSISTEE PAR ORDINATEUR (CAO).....	20
I.11.1 Définition	20
I.11.2 Les outils de la CAO (logiciels)	20
I.12 FABRICATION ASSISTEE PAR ORDINATEUR.....	20
I.12.1 Définition	20
I.12.2 Les outils de la FAO (logiciels)	21
I.13 CONCLUSION.....	21

Chapitre II : Présentation de la machine à commande numérique (DENZER-CNC).

II.1 INTRODUCTION.....	22
II.2 LE FRAISAGE NUMERIQUE.....	22
II.2.1 La fraiseuse	22
II.2.2 Les différents types de fraiseuses	23
II.2.2.1 Fraiseuse manuelle	23
II.2.2.2 Fraiseuse par apprentissage	23
II.2.2.3 Fraiseuse à commande numérique	23
II.2.2.3.1 La fraiseuse à trois (3) axes	23
II.2.2.3.2 Fraiseuse à quatre (4) axes	24

II.2.2.3.3 Fraiseuse à cinq (5) axes	24
II.2.2.4 Autres types de fraiseuses	25
II.2.2.4.1 Fraiseuses horizontales ..	25
II.2.2.4.2 Fraiseuses verticales ..	25
II.2.2.4.3 Fraiseuses spéciales	25
II.2.2.4.4 Fraiseuses universelles	26
II.2.2.4.5 Fraiseuses automatiques à contrôle numérique	26
II.3 NECESSITE DU DEVELOPPEMENT DE LA MACHINE A COMMANDE NUMERIQUE A TROIS (3) AXES	26
II.4 DESCRIPTIF DU PRINCIPE DE FONCTIONNEMENT DE LA MACHINE DENZER- CNC.....	26
II.4.1 La partie opérative	27
II.4.2 La partie commande.....	28
II.5 LES MATIERES D'ŒUVRE DE LA MACHINE DENZER-CNC.....	28
II.6 LA MACHINE A COMMANDE NUMERIQUE « DENZER-CNC »	29
II.6.1 Les principaux modules de la machine DENZER-CNC.....	29
II.6.1.1 Les éléments de la partie commande.....	29
II.6.1.1.1 Carte Arduino	29
II.6.1.1.2 Carte Arduino Sensor Shield v5.0	30
II.6.1.1.3 Driver ST330-v3.....	31
II.6.1.1.4 Alimentation	32
II.6.1.2 Les éléments de la partie opérative	32
II.6.1.2.1 Moteurs pas à pas	33
II.6.1.2.2 La broche BOSCH GGS 28 C PROFESSIONAL.....	34
II.7 ENVIRONNEMENT DE TRAVAIL.....	35
II.8 LES CARACTERISTIQUES DE LA MACHINE DENZER-CNC	40
II.9 ARCHITECTURE MATERIEL DE LA MACHINE DANZER-CNC	41
II.10 ENVIRONNEMENT LOGICIEL DE LA MACHINE DENZER-CN	49
II.10.1 SolidWorks	49
II.10.2 Netfabb.....	51
II.10.3 Inkscape et l'extension G-codetools	53
II.10.3.1 Utilisation de G-codetools.....	54
II.10.4 Arduino IDE.....	57
II.10.5 Grbl	58
II.10.5.1 Grbl Contrôleur	58
II.11 CONCLUSION	59

Chapitre III : Automatisation de la machine et simulation de la commande numérique.

III.1 INTRODUCTION.....	60
III.2 SIMULATION DE LA COMMANDE NUMERIQUE.....	60
III.2.1 Le logiciel PROTEUS	61
III.2.1.1 PROTEUS ISIS.....	62
III.2.2 Commande des moteurs pas à pas	62
III.2.3 La simulation avec le couple L297-L298	69
III.2.3.1 liste des composants.....	70
III.2.3.2 Le programme de commande.....	73
III.2.3.3 Le logiciel de commande « Universel G-code Sender ».....	74
III.2.3.4 Virtual Serial Port Emulateur.....	75
III.2.3.5 Lancement de la simulation	76
III.3 MISE EN MARCHE DE LA MACHINE DENZER-CNC	77
III.3.1 Conception de la pièce à usiner	77
III.3.2 Génération du fichier G-code de la pièce à usiner.....	77
III.3.3 Configuration de l’interface de commande numérique	77
III.3.4 Lancement de L’usinage automatique de la pièce	79
III.3.5 Exemple d’usinage d’une pièce	79
III.4 FINALISATION DU PROJET (Fiche technique).....	84
III.5 CONCLUSION	85
Conclusion générale.....	86

Bibliographie

LISTE DES FIGURES

Chapitre I : Généralité sur les machines à commande numérique.

Figure -I.1- Structure d'une MOCN.....	5
Figure -I.2- Les éléments de la partie opérative.....	6
Figure -I.3- Fonction originale d'une commande numérique	7
Figure -I.4- Fraisage à commande numérique	8
Figure -I.5- Machine-outil à tour numérique	9
Figure -I.6- rectification cylindrique	9
Figure -I.7- rectification plane	9
Figure -I.8- Electro-érosion.....	10
Figure -I.9 - Fonctionnement en boucle ouverte	11
Figure -I.10- Fonctionnement avec commande adaptative	12
Figure -I.11- Fonctionnement en boucle fermée	12
Figure -I.12- Machine à commande numérique à trois axes.....	13
Figure -I.13- Moteur bipolaire.....	14
Figure -I.14- Moteur unipolaire.....	14
Figure -I.15- Moteur pas à pas à reluctance variable	15
Figure -I.16- Moteur pas à pas à aimant permanent.....	16
Figure -I.17- moteur pas à pas hybride	16

Chapitre II : Présentation de la machine à commande numérique (DENZER-CNC).

Figure -II.1- Ce qui est possible de faire avec une fraiseuse trois (3) axes.	24
Figure -II.2- Fraiseuse horizontale.....	25
Figure -II.3- Fraiseuse verticale.	25
Figure -II.4- Principe du fraisage.....	27
Figure -II.5- Armoire de commande.	29
Figure -II.6- Carte Arduino Uno.....	30
Figure -II.7- Carte Arduino Sensor Shield v5.0.....	31
Figure -II.8- Driver ST330-v3.	31
Figure -II.9- Boite d'alimentation.....	32
Figure -II.10- Partie opérative.....	32
Figure -II.11- Moteur pas à pas NEMA 23.....	33

Figure -II.12- Broche BOSCH GGS 28 C PROFESSIONAL.....	34
Figure -II.13- Situation géographique de DENZER Technologies – Bejaia.	36
Figure -II.14- Situation géographique de DENZER Technologies – Akbou.....	36
Figure -II.15- Conception sites web.....	37
Figure -II.16- Robotisme et systèmes électroniques.....	37
Figure -II.17- Développement et maintenance.	38
Figure -II.18- Fenêtre au lancement de SolidWorks.....	49
Figure -II.19- Nouveau document SolidWorks.	50
Figure -II.20- L’interface de création de pièce.	51
Figure -II.21- Interface Netfabb.....	51
Figure -II.22- Réparations des erreurs.....	52
Figure -II.23- Réparation automatique.....	52
Figure -II.24- Validation de la réparation.	53
Figure -II.25- Fichier STL réparé.	53
Figure -II.26- Interface inkscape.....	54
Figure -II.27- Point d’orientation.....	54
Figure -II.28- Bibliothèque d’outils.	55
Figure -II.29- Panneau paramètre de modification.	55
Figure -II.30- Chemin vers G-code.....	56
Figure -II.31- Emplacement fichier G-code.....	56
Figure -II.32- Interface de l’ide Arduino.	57
Figure -II.33- Interface de Grbl Contrôleur.	58
Figure -II.34- Panneau des réglages Grbl.	59

Chapitre III : Automatisation de la machine et simulation de la commande numérique.

Figure -III.1- Home page de PROTEUS.	61
Figure -III.2- Nouvelle page ISIS... ..	62
Figure -III.3- Schéma du pont en H... ..	62
Figure -III.4- Circuit L298 connecté à un Arduino... ..	63
Figure -III.5- Déclaration des sorties... ..	65
Figure -III.6- Les différents modes.	66
Figure -III.7- Incrémentation ou décrémentation de la variable « etape ».	66
Figure -III.8- Application du signal PWM si les entrées ENABLE.....	67

Figure -III.9- Exemple.....	67
Figure -III.10- Simulation de l'exemple sur proteus.....	68
Figure -III.11- Montage électronique dans ISIS PROTEUS.....	69
Figure -III.12- Configuration du moteur pas à pas.....	70
Figure -III.13- Le couple L297-L298.....	71
Figure -III.14- Le régulateur de tension 7805.....	72
Figure -III.15- Configuration du port série.....	72
Figure -III.16- Lancement du GrblUpload.....	73
Figure -III.17- Emplacement du fichier « grblUpload.ino.elf ».....	73
Figure -III.18- Chargement du programme dans l'Arduino.....	74
Figure -III.19- Interface de UGS.....	74
Figure -III.20- commande en mode manuel.....	75
Figure -III.21- Interface de VSPE.....	75
Figure -III.22- Lancement de la simulation.....	76
Figure -III.23- commande en mode automatique.....	76
Figure -III.24- Configuration du fichier config.h.....	78
Figure -III.25- les paramètres de réglage du Grbl.....	78
Figure -III.26- réglage du nombre d'interrupteurs de butée pour un seul axe.....	80
Figure -III.27- Définition des états initiaux des fins de courses.....	80
Figure -III.28- Contrôle de la vitesse de la broche.....	80
Figure -III.29- Configuration Grbl contrôleur.....	81
Figure -III.30- Configuration avancé Grbl contrôleur.....	81
Figure -III.31- Importation du fichier d'usinage.....	82
Figure -III.32- La mise sous tension.....	82
Figure -III.33- lancement de l'usinage.....	83
Figure -III.34- Fiche technique de la machine DENZER-CNC.....	84

LISTE DES TABLEAUX

Chapitre I : Généralités sur les machines à commande numérique

Tableau -I.1- les différents types de moteurs pas à pas.	17
Tableau -I.2- Les principales fonctions du G-code.....	19

Chapitre II : Présentation de la machine à commande numérique (DENZER-CNC)

Tableau -II.1- Les caractéristiques de la machine DENZER-CNC.....	40
Tableau -II.2- Les composants matériels mécanique de la machine DENZER-CNC.....	41

INTRODUCTION GENERALE

L'industrie mécanique doit une grande partie de son développement très avancé aux technologies diversifiées de fabrication de pièces et composants qui constituent les ensembles et sous-ensembles mécanique.

Aujourd'hui, les machines-outils à commande numérique sont très utilisés dans le domaine de la fabrication mécanique permettant une production économique et rentable. Pour cause, les progrès significatifs qu'a subi la commande des machines électriques. 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. Ces commandes sont en majorité basées sur les microprocesseurs, les DSP (Digital Signal Processor) et les microcontrôleurs PIC. Ces processeurs sont équipés d'unité arithmétique et logique (UAL) dédiée à la réalisation des calculs arithmétiques. Ils intègrent également des périphériques tel que les convertisseurs analogiques/numériques et les « timers » bien adaptés aux besoins de commande de machines électriques.

Les moteurs pas à pas sont l'interface idéale entre l'électronique numérique et la mécanique, comment peut-on, donc, les commander numériquement et quelles sont les différentes étapes à suivre pour réaliser cette commande ?

Pour la finalisation de notre cursus Master en Automatismes Industriels notre projet fin d'étude s'est porté sur la commande et la réalisation de la carte de commande d'une machine à commande numérique. Projet proposé et réalisé au niveau de l'EURL Denzer-technologie.

Le projet se divise en deux parties :

- Fabrication et réalisation pratique de l'armoire de commande de la machine.
- Réalisation du projet de mémoire.

Le mémoire comporte, trois chapitres structurés comme suit :

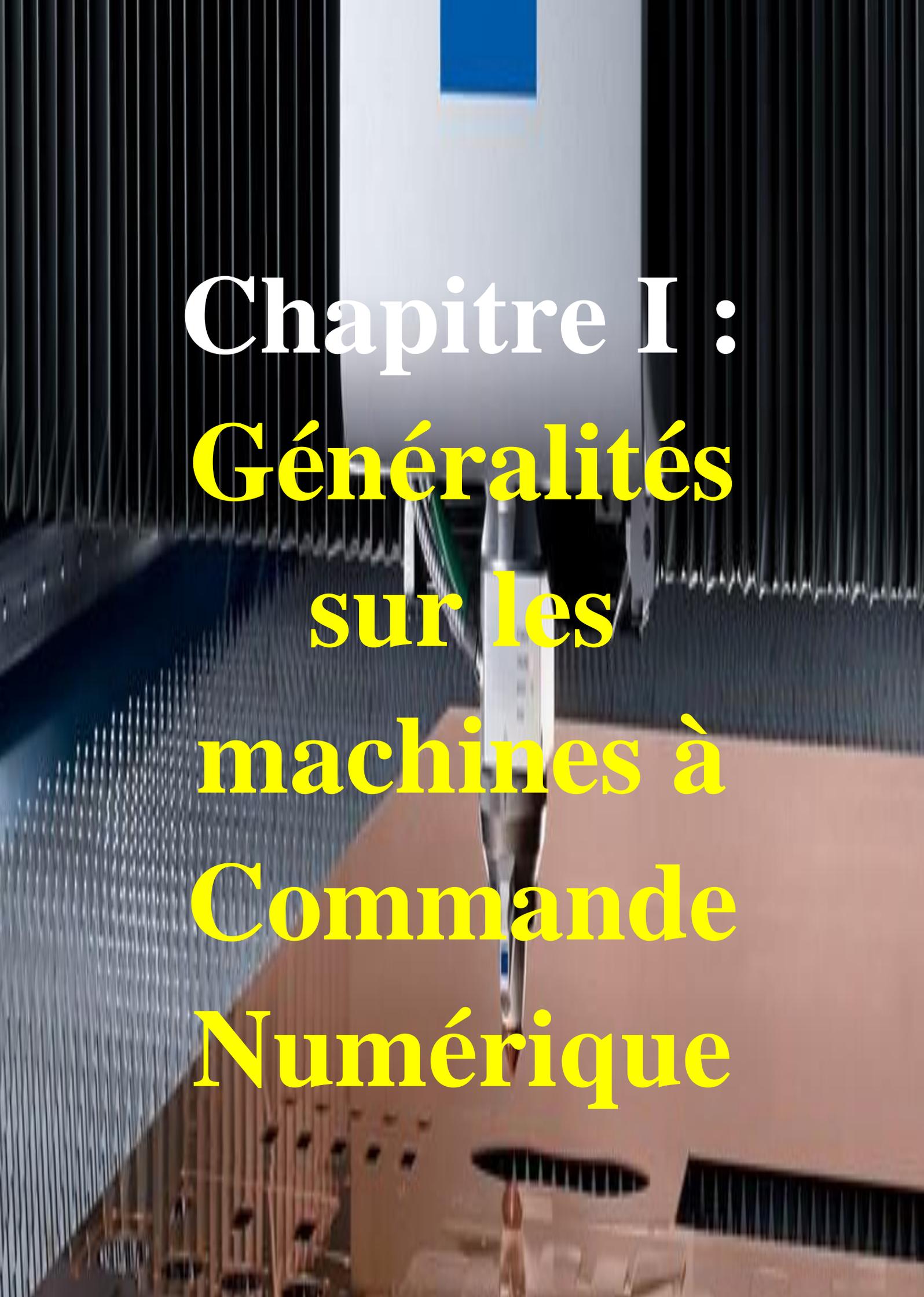
Le premier chapitre est consacré pour la présentation des machines-outils, de la commande numérique, des moteurs pas à pas et les différentes étapes avant l'acquisition d'une machine-outil à commande numérique.

INTRODUCTIO GENERALE

Le second chapitre concerne la description et le principe de fonctionnement de la machine DENZER-CNC et une présentation de l'environnement de travail.

Le dernier chapitre porte sur la finalisation du projet avec la mise en marche de la machine DENZER-CNC en passant par une simulation.

Ce mémoire comporte, en outre, une introduction générale et une conclusion générale.



Chapitre I :
Généralités
sur les
machines à
Commande
Numérique

I.1 INTRODUCTION

Ce chapitre apporte les connaissances fondamentales sur les machines-outils et de la commande Numérique. Ainsi, que les différentes étapes qu'il convient de respecter avant l'acquisition d'une machine-outil à commande numérique (MOCN).

I.2 HISTORIQUE DES MACHINES-OUTILS [1]

Selon le nouveau Larousse illustré de 1904 : « la machine-outil est une machine dont l'effet final est un outil mis en mouvement, et qui a pour but de remplacer la main de l'homme lors du façonnage des matières ». L'évolution des machines-outils commence il y a plus de cinq mille ans avec les premiers tours à potier.

A travers l'évolution des machines-outils, du tour du potier au tour numérique, apparaissent en filigrane de nombreuses tendances qui ont déterminé l'histoire de l'humanité.

D'abord, l'histoire des techniques qui a été influencée par la succession des types d'énergie utilisée : la force de l'homme (le pied sur la pédale), celle du vent ou de l'eau (au XVIe siècle, grâce à la mécanisation du système d'entraînement, le tour fut couplé à des moulins et autres roues à aubes), la combustion du charbon (la machine à vapeur), l'électricité (mais aussi l'énergie lumineuse, chimique, acoustique jusqu'au faisceau à particule) puis le développement de l'informatique.

Ensuite l'histoire des objets qui a suivi l'évolution des matériaux utilisés, la terre, le bois, le métal puis les matières les plus diverses et les alliages développés pour les besoins de la technologie moderne.

Enfin, l'histoire des machines-outils soulève des questions de société comme la condition du travailleur (de l'ouvrier à la chaîne, esclave de la machine-outil, jusqu'au technicien responsable des tours numériques...) et la condition de l'homme moderne. La production standardisée et les gains de productivité permis par les machines-outils ont en effet pu lui offrir la consommation de masse et le temps libre.

I.3 LA COMMANDE NUMERIQUE

I.3.1 Origines de la commande numérique

La Seconde Guerre Mondiale est à peine terminée que les États-Unis en amorcent une autre, la « guerre froide ». L'effort de recherche/développement en matière d'armement est à nouveau relancé. Dans ce contexte, Parsons Works, se voit confier par l'US Air Force la mission d'améliorer la productivité des chaînes de fabrication et plus particulièrement du long, complexe et fastidieux processus de fabrication et de contrôle des surfaces gauches des pales d'hélicoptère. John T. Parsons, aidé de son ingénieur en chef Frank Stulen mènent les études. Ils imaginent différents concepts dont l'un consiste à motoriser les axes des machines concernées. Se pose alors le problème du pilotage de ces moteurs. Ce point les amène à faire appel aux moyens et compétences tout d'abord d'une de leurs filiales, La Parson Corp, spécialisée dans les solutions usant de calculateurs (on appelait ainsi l'ancêtre de l'ordinateur) alimentés en données par des bandes perforées et, d'autre part, à consulter le M.I.T (Massachusetts Institute of Technology). Finalité : commander par un calculateur (IBM y participera) le déplacement d'une table sur glissières mue par une vis à billes couplée à un moteur électrique. En 1952, le Dr Richard KEGG (Milacron) et le MIT (concepteur des servo-contrôleurs) présentent au gratin de l'aéronautique le résultat prometteur de leurs travaux sur une fraiseuse Cincinnati Milacron Hydrotel. Une demande de brevet est aussitôt déposée par précaution au bureau des brevets américain (USPTO) par Parson et Stulen, qui l'accepte sous le n° 2,820,187.

Sur la base des succès précédents et après avoir travaillé pendant cinq ans à industrialiser le concept (affinage des tout premiers concepts de programmation de trajectoires avec interpolation axiale), Parson et Stulen confirment le brevet le 14 janvier 1958 par un brevet intitulé « Motor Controlled Apparatus for Positioning Machine Tool » qui sonne la naissance commerciale de cette technologie. General Electric sera historiquement le premier fabricant industriel de commandes numériques, rejoint en 1962 par son voisin japonais, FANUC. En 1964, les États-Unis comptaient 35 000 machines à commande numérique [2].

I.3.2 Principe de la commande numérique

La commande numérique a pour tâche de générer des mouvements d'axes. Ces instructions qui viennent du programme pièce ou de l'opérateur machine consisteront en une position et une vitesse de déplacement. Le processeur de la commande numérique va alors générer une consigne afin qu'on puisse commander les moteurs d'axes. Elle aura également

pour tâche, lors du déplacement des axes, de vérifier la position de ces derniers et dans certain cas, la vitesse de déplacement. Les commandes numériques actuelles sont capables d'effectuer des mouvements en combinant simultanément les positions sur les axes X, Y et Z. Elles s'appellent commande numérique par ordinateur. De plus, elles sont capables de gérer plusieurs axes simultanément. On parle alors d'interpolation. Cette dernière peut être linéaire ou circulaire selon la consigne générée par le microcontrôleur de la partie commande. Le rôle de ce processeur est d'interpréter un code machine que l'on appelle un code G, puis de générer les signaux de commande des axes et de la broche [3].

I.4 LA STRUCTURE D'UNE MACHINE A COMMANDE NUMERIQUE

La machine-outil est équipée d'une commande numérique capable de réaliser les calculs des coordonnées des points définissant une trajectoire (interpolation), on dit qu'elle est à ordinateur. Elle est appelée CNC (Commande Numérique par Calculateur). La plupart des MOCN sont des CNC [4].

La machine-outil à commande numérique forme un ensemble comprenant : une partie commande et une partie opérative, tel qu'elle est représentée dans la figure -I.5-.

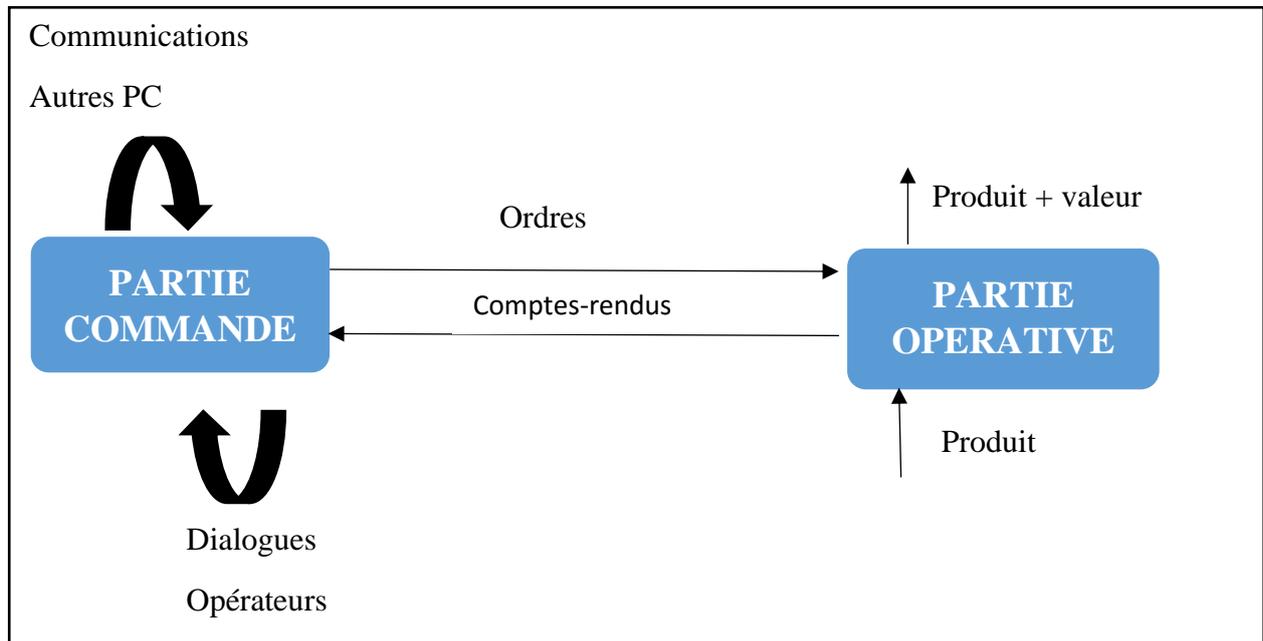


Figure -I.1- Structure d'une MOCN.

I.4.1 Partie opérative

Les mouvements sont commandés par des moteurs ; presque comparable à une machine-outil classique, et elle comprend :

- Un socle, très souvent en béton hydraulique vibré, assurant l'indépendance de la machine au sol,
- Un bâti, un banc, dont les larges glissières sont en acier traité,
- Un support outil (broche, torche, laser, jet d'eau ...),
- Une table support pièce, mobile selon 2 ou 3 axes, équipée de système de commande à vis et écrou à bille. Le granit, ou le granit reconstitué, est utilisé pour la fabrication des tables et des bâtis des machines à mesurer tridimensionnelles des rectifieuses et de certains tours,
- Des moteurs chargés de l'entraînement de la table,
- Un élément de mesure ou capteur de position renseignant à tout moment sur la position du mobile sur chaque axe,
- Une dynamo tachymétrique assurant la mesure de la vitesse de rotation.

Les éléments de la partie opérative sont représentés dans la figure –I.6-.

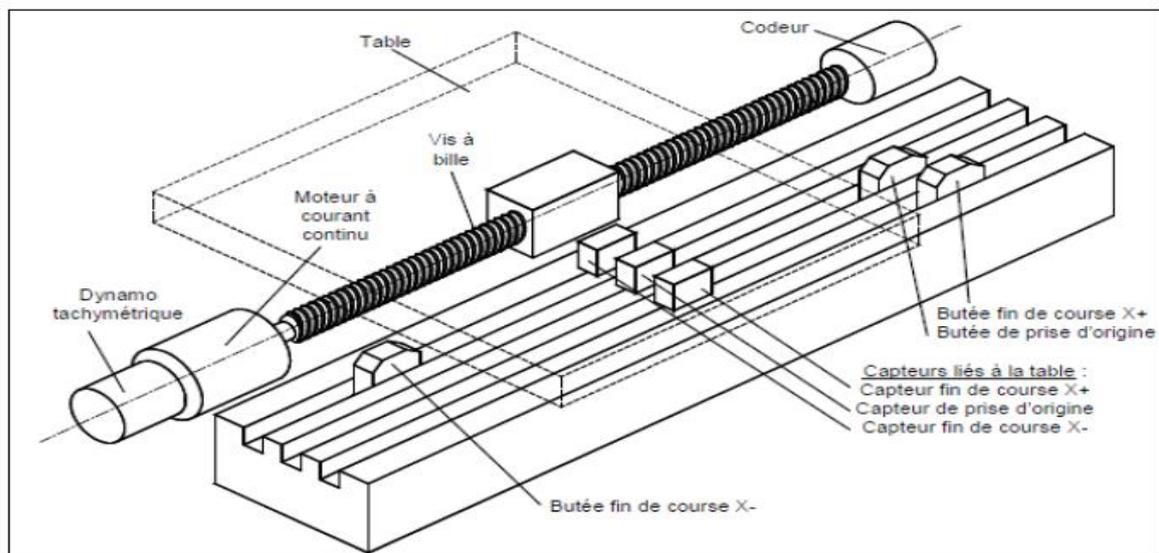


Figure-I.2- Les éléments de la partie opérative. [4]

I.4.1.1 Tâches effectuées

Les tâches effectuées sur le site de la partie opérative sont :

- ✓ Chargement et déchargement (pièce port pièce).
- ✓ Chargement et déchargement (outils port outils).

- ✓ Intervention manuelles nécessitées par l’usinage et l’entretien.
- ✓ Surveillance de commande.

I.4.1.2 Armoire électrique de relaying ou armoire de puissance

Elle est composée :

- D’un automate programmable gérant toutes les entrées - sorties,
- D’un relais,
- D’électrovannes,
- De cartes variateurs d'axes (une par axe),
- De contacteurs (1 par élément de machine : axes, broche)
- D’un interrupteur général avec sécurité.

I.4.2 Partie commande

Différente d'une machine conventionnelle et constituée d'une armoire dans laquelle on trouve :

- ✓ Le pupitre permettant de rentrer les commandes à l'aide d'un clavier,
- ✓ Le lecteur de données (ce lecteur peut être une option lors de l'achat de la machine),
- ✓ La sortie RS 232 pour les liaisons avec les Périphériques externes,
- ✓ L’écran de visualisation de toutes les données enregistrées,
- ✓ Le calculateur,
- ✓ les cartes électroniques (commandes d'axes, mémoire ...).

La partie commande est alimentée par un courant faible et ne peut donc pas alimenter directement les moteurs de la machine. La fonction originale de cette dernière est représentée dans la figure –I.7-.

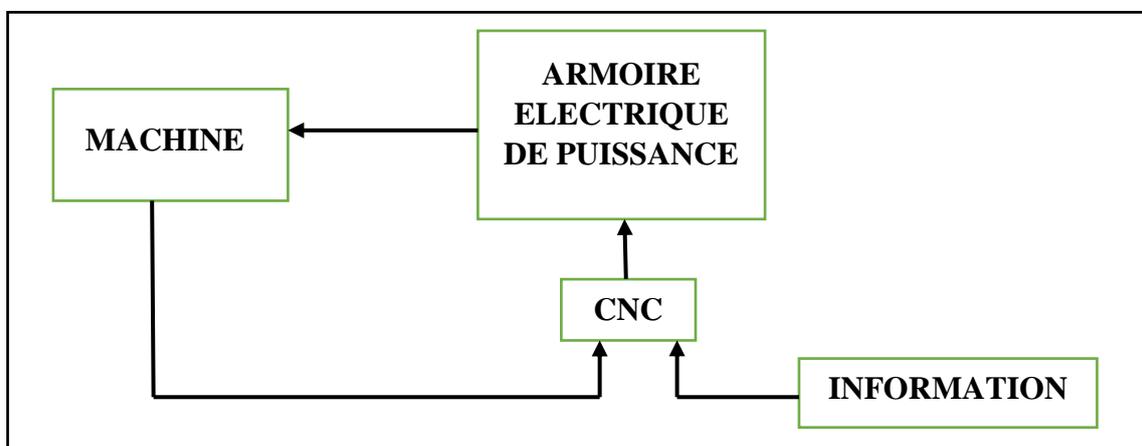


Figure-I.3- Fonction originale d’une commande numérique [4]

I.5 PRINCIPE DE FONCTIONNEMENT D'UNE MACHINE A COMMANDE NUMERIQUE

Les programmes d'usinage sont réalisés à partir d'une origine appelée « origine programme » (OP) positionnée par le programmeur. Le programme commande les déplacements relatifs entre le brut et les outils dans le but de réaliser l'usinage de la pièce finale. Ces déplacements sont réalisés dans un repère orthonormé normalisé (O, X, Y, Z) basé sur la structure de la machine.

L'axe (Z) de ce repère est un axe confondu avec celui de la broche de la machine — axe de rotation de la fraise en fraisage, axe de rotation de la pièce en tournage. Le sens positif de cet axe est donné par le sens d'éloignement de l'outil par rapport à la pièce. La détermination de l'axe (X) entre les 2 axes restants se fait en identifiant celui qui permet le plus grand déplacement. Le sens positif de (X) est déterminé par le sens logique d'éloignement de l'outil par rapport à la pièce.

L'axe(Y) est déterminé à partir de (X) et (Z) grâce à la règle du trièdre direct [5].

I.6 LES FAMILLES DES MACHINES A COMMANDE NUMERIQUE

On peut aussi découper la famille des commandes numériques en quatre sous familles de machines :

I.6.1 Fraisage à commande numérique (FCN)

Le fraisage est un procédé de fabrication où l'enlèvement de matière sous forme de copeaux résulte de la combinaison de deux mouvements : la rotation de l'outil de coupe, d'une part, et l'avancée de la pièce à usiner d'autre part. Le fraisage est habituellement réalisé par une machine-outil, la fraiseuse qui est particulièrement adaptée à l'usinage de pièces prismatiques et permet également, si la machine est équipée de commande numérique, de réaliser tous types de formes même complexes. L'outil classiquement utilisé est la fraise [6]. La figure –I.8- présente un exemple de fraisage numérique.

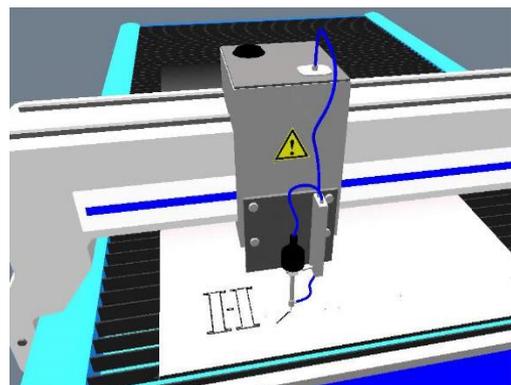


Figure-I.4- Fraisage à commande numérique. [7]

I.6.2 Tournage à commande numérique (TCN)

Le tournage est une technique d'usinage qui consiste à enlever, à l'aide d'outils coupants, de la matière sur une pièce initiale cylindrique pour obtenir une pièce finale (figure-I.9). La matière est enlevée par la combinaison de la rotation de la pièce usinée et du mouvement de l'outil coupant [8].

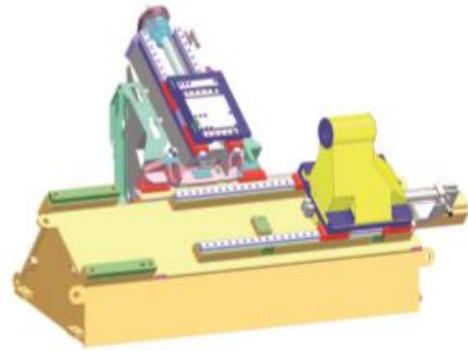


Figure -I.5- Machine-outil à tour numérique. [9]

I.6.3 Rectification à commande numérique

La rectification d'une pièce mécanique est une opération destinée à améliorer son état de surface. Les deux techniques principales sont la rectification plane (figure -I.11-) et la rectification cylindrique (figure -I.10-). La rectification s'effectue sur une machine-outil conçue à cet effet : la rectifieuse. Il s'agit de rectifier, c'est-à-dire d'approcher une surface d'une forme parfaite (en général : plan, cylindre de révolution ou cône). La rectification est souvent utilisée dans le but de préparer des surfaces de frottement, par exemple, la portée d'un arbre qui tournera dans un palier lisse ou dans un joint d'étanchéité. Elle peut également être utilisée pour donner un profil particulier à la pièce lorsque la meule a été au préalable usinée au profil complémentaire [10].

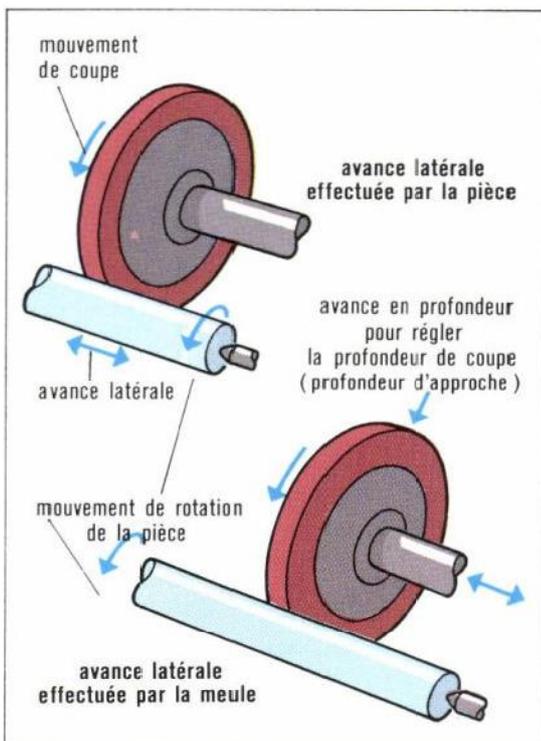


Figure -I.6- rectification cylindrique. [11]

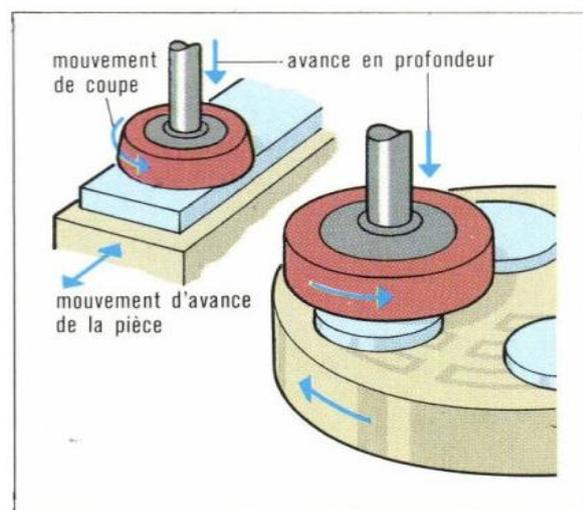


Figure-I.7- rectification plane. [11]

I.6.4 Electro-érosion à commande numérique

L'électroérosion, appelée aussi EDM (*electrical discharge machining*), est un procédé d'usinage qui consiste à enlever de la matière dans une pièce en utilisant des décharges électriques.

On parle aussi d'usinage par étincelage. Cette technique se caractérise par son aptitude à usiner tous les matériaux conducteurs de l'électricité (métaux, alliages, carbures, graphites, etc.) quelle que soit leur dureté. La figure –I.12- représente un dessin d'une machine d'électro-érosion à commande numérique [12].

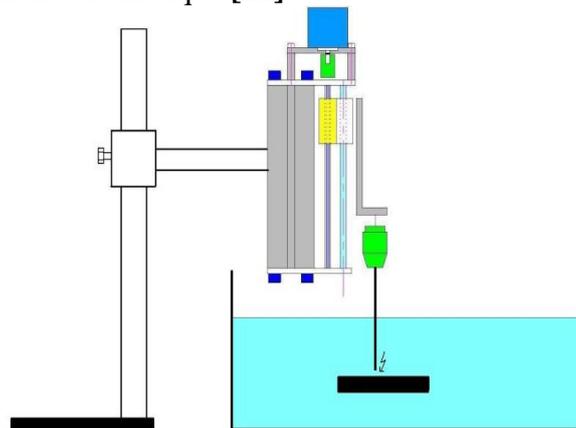


Figure-I.8- Electro-érosion. [13]

I.7 CLASSIFICATION DES MOCN [14]

Les machines-outils à commande numérique (MOCN) sont classées selon :

- le mode de fonctionnement de la machine
- le nombre d'axes de la machine
- le mode d'usinage
- le mode de fonctionnement du système de mesure
- le mode d'entrée des informations
- Les machines-outils à commande numérique (MOCN) peuvent être assistées d'une programmation extérieure et de mécanismes tendant à les rendre encore plus performantes, tels que :
 - ordinateur et ses périphériques.
 - commande adaptative.
 - pré réglage des outils.
 - codage des outils.
 - chargeur d'outils et magasin.

- chargeur et convoyeur de pièces.
- combinaison de type d'usinages (centre de tournage, centre d'usinage).
- table de montage.
- évacuateur de copeaux.
- dispositifs de contrôle de pièces.

I.7.1 Classification des MOCN selon le mode de fonctionnement

Les MOCN sont classifiées selon les modes de fonctionnement suivants :

I.7.1.1 Fonctionnement en boucle ouverte

En boucle ouverte (figure –I.13-), le système assure le déplacement du chariot mais ne le contrôle pas.

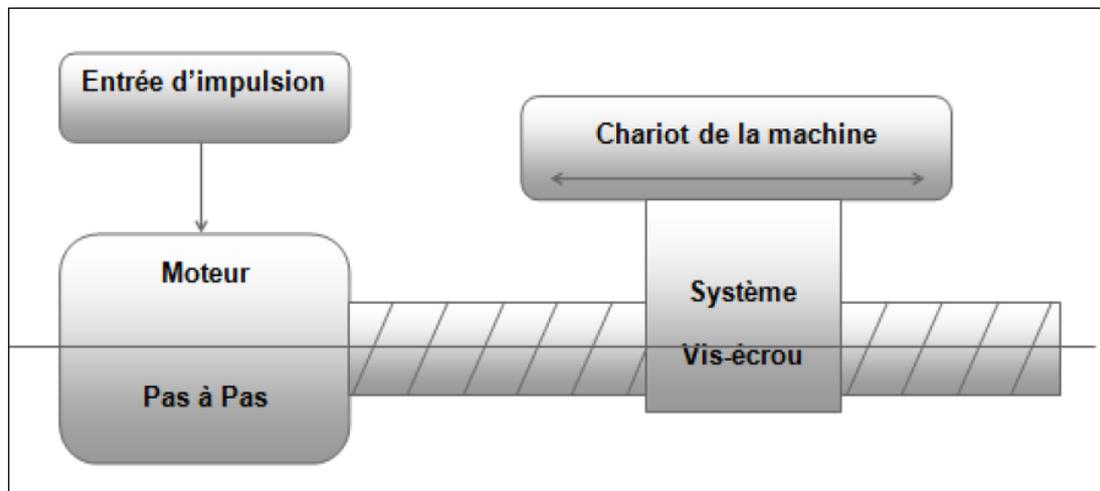


Figure-I.9- Fonctionnement en boucle ouverte [15]

I.7.1.2 Fonctionnement avec commande adaptative

La commande adaptative (figure –I.14-) réalise d'une façon continue et automatique l'adaptation des conditions de coupe. Des capteurs relèvent les valeurs de couple de la broche, l'amplitude de vibration de la broche, la température au point de coupe. Ces informations sont transmises à une unité spéciale qui les envoie vers le directeur de commande numérique qui agit selon l'analyse des informations sur les conditions de coupe pour permettre une meilleure qualité de travail, une meilleure productivité et une plus grande sécurité.

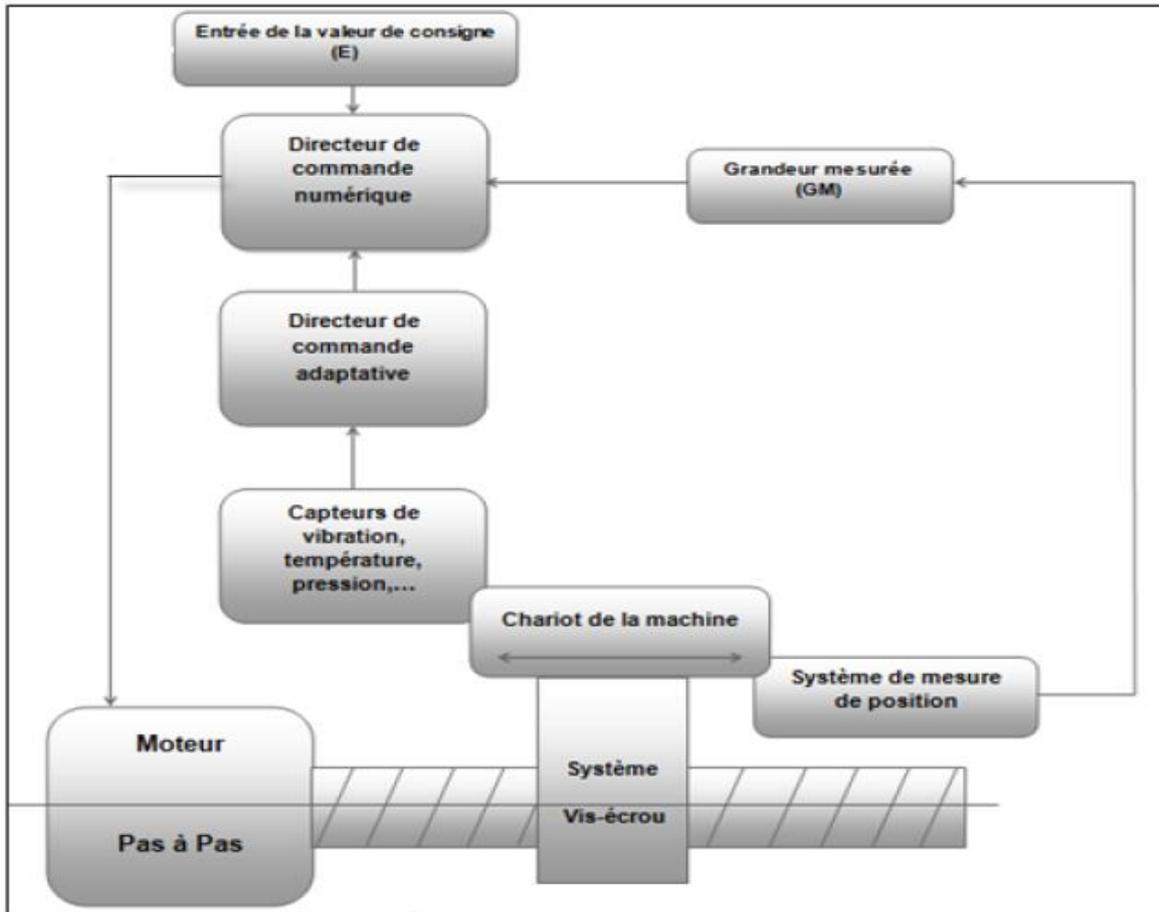


Figure -I.10- Fonctionnement avec commande adaptative [15]

I.7.1.3 Fonctionnement en boucle fermée

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

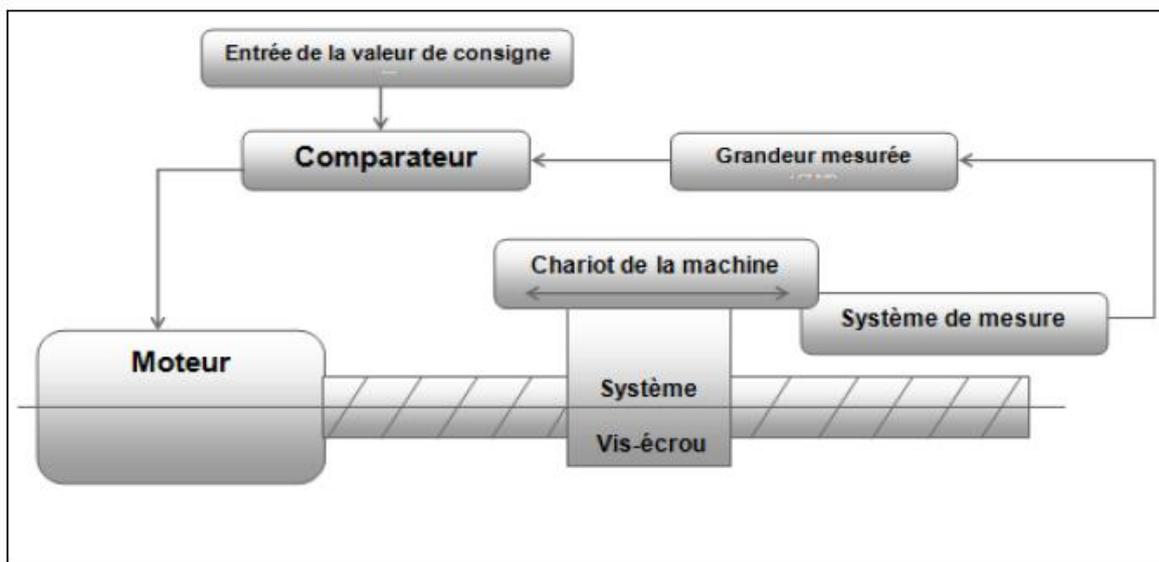


Figure -I.11- Fonctionnement en boucle fermée [15]

I.7.2 Classification des MOCN selon le nombre d'axe

Les possibilités de travail des MOCN s'expriment en nombre d'axes de travail.

Un axe définit toute direction principale suivant laquelle le mouvement relatif de l'outil et de la pièce a lieu lorsqu'un seul des moteurs de déplacement fonctionne avec contrôle numérique continue.

Un demi-axe définit la direction dans laquelle l'avance n'est pas contrôlable numériquement mais contrôle par pistes, cames ou plateaux diviseurs.

I.7.3 Classification des MOCN selon le mode d'usinage

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

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

I.8 LA MACHINE A COMMANDE NUMERIQUE A TROIS AXES

L'usinage trois (3) axes est l'une des techniques les plus employées dans la fabrication de pièces mécaniques. Tel que les machines-outils classiques comme la fraiseuse qui permet un travail de la matière sur trois (3) axes (X, Y et Z) (figure -I.19-). La machine-outil procède alors à l'enlèvement des copeaux suivant trois directions de base correspondant aux axes d'une surface plane. Tout à fait adaptée aux pièces peu profondes. Cependant cette technique a de grosses limites quand il s'agit de traiter une pièce profonde avec des cavités étroites. Pour cela de nouvelles technologies d'usinage ont été développées tel que 4 axes puis axes [16].

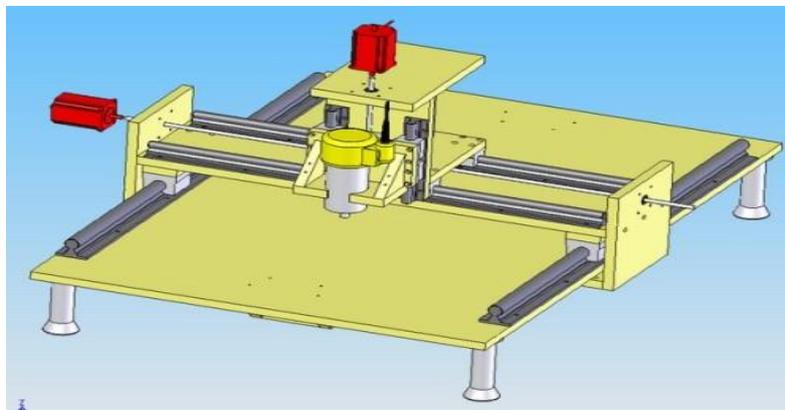


Figure -I.12- Machine à commande numérique à trois axes [17]

I.9 LES MOTEURS PAS A PAS

Le moteur pas à pas est un convertisseur électromécanique qui assure la transformation d'un signal électrique impulsionnel en un déplacement mécanique (angulaire ou linéaire). Sa structure de base se présente sous la forme de deux pièces séparées mécaniquement, le Stator et le Rotor. L'interaction électromagnétique entre ces deux parties assure la rotation.

I.9.1 Les différents types de moteur pas à pas

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

Il existe 3 types de moteurs pas à pas, à aimants permanents, à réluctance variable ou hybrides.

On peut les classer aussi en deux types selon l'alimentation de bobinage, les moteurs bipolaires, les moteurs unipolaires.

I.9.1.1 Les moteurs bipolaires

Ils ont en général quatre (4) fils et se commandent en inversant le sens du courant dans une des deux bobines, tel que c'est représenté dans la figure -I.20-, qui sont alimentés une fois dans un sens, une fois dans l'autre sens. Ils créent une fois un pôle nord, une fois un pôle sud d'où le nom de bipolaire.

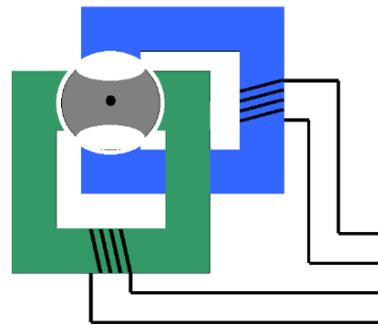


Figure -I.13- Moteur bipolaire [18]

I.9.1.2 Les moteurs unipolaires

Ils ont en général six (6) fils, dont deux sont reliés au milieu des bobines, tel que c'est représenté dans la figure -I.21-. Ils se commandent en les alimentant tour à tour, mais les bobinages sont alimentés toujours dans le même sens par une tension unique d'où le nom d'unipolaire.

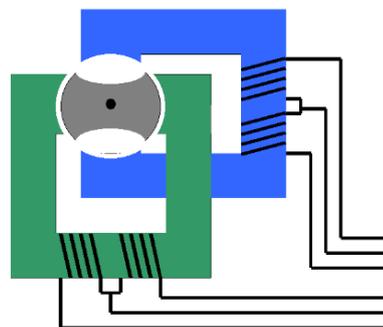


Figure-I.14 - Moteur unipolaire [18]

I.9.1.3 Les moteurs à réluctance variable

Les moteurs à réluctance variable fonctionnent selon un principe différent de celui des moteurs à aimant permanent. Ils possèdent bien évidemment un stator, mais ce dernier est fabriqué en acier doux non magnétique. Il n'est pas lisse et possède plusieurs dents. Le stator est composé de 8 plots sur lesquels sont enroulés les bobinages, ce qui donne 4 phases. Le rotor, quant à lui, ne comporte que 6 dents [19].

I.9.1.3.1 Le principe de fonctionnement

Le principe de fonctionnement est très simple : l'un des plots opposés du stator est traversé par un courant afin d'aligner deux des dents du rotor avec ces plots (figure I.22-). Le mode de commande peut dans ce cas, de la même façon que pour les autres moteurs, être monophasé, biphasé ou demi-pas.

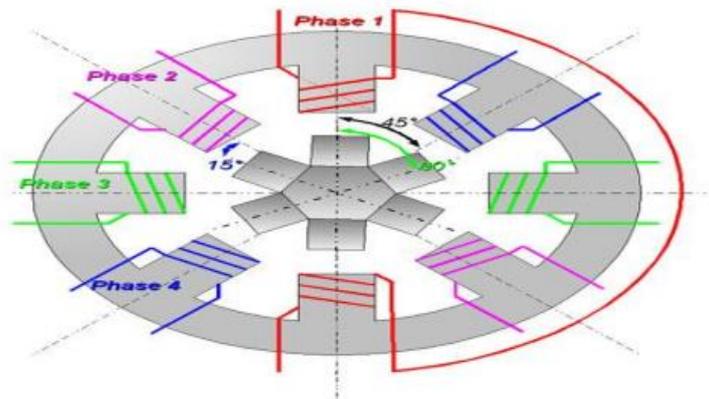


Figure-I.15- Moteur pas à pas à réluctance variable [19]

I.9.1.4 Les moteurs à aimant permanent

Les moteurs à aimants permanents sont semblables aux moteurs à réluctance variable, sauf que le rotor possède des pôles NORD et SUD. À cause des aimants permanents, le rotor reste freiné à sa dernière position lorsque le bloc d'alimentation cesse de fournir des impulsions.

I.9.1.4.1 Le principe de fonctionnement

C'est le modèle dont le fonctionnement est le plus simple. Le rotor est constitué d'un aimant permanent, et le stator comporte deux paires de bobines (figure I.23-). En agissant sur les bobines alimentées, et le sens des courants, on fait varier le champ créé par le stator. À chaque pas, la direction du champ induit par le stator tourne de 90°. L'aimant permanent suit le déplacement du champ magnétique créé par les bobines et s'oriente selon une de ses quatre positions stables. Comme le rotor est aimanté, lorsque le moteur n'est pas alimenté le

flux magnétique dû à l'aimant permanent crée un couple résiduel en se plaçant dans l'axe de l'une des bobines [20].

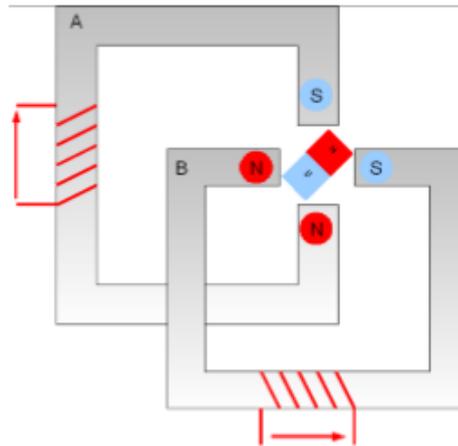


Figure-I.16- Moteur pas à pas à aimant permanent [20]

I.9.1.5 Les moteurs hybrides

C'est des moteurs qui superpose le principe de fonctionnement des moteurs a aimant permanent et a reluctance variable et combine leurs avantages. Le rotor est constitué de deux disques dentés décalés mécaniquement. Entre ces deux disques est inséré un aimant permanent.

Leur intérêt réside dans un meilleur couple, une vitesse plus élevée, une résolution de 100 à 400 pas/tour, et sont plus chers [21].

I.9.1.5.1 Principe de fonctionnement

En mode pas entier, les bobines sont alimentées une par une alternativement, dans un sens puis dans l'autre. Dans la figure -I.24-, on voit bien l'intérêt d'avoir un rotor polarisé : quand on coupe l'alimentation de la bobine verte et qu'on alimente la bobine rose, en l'absence de polarisation du rotor on n'aurait aucun contrôle sur le sens de rotation, les deux dents bleues et rouges étant attirés de la même façon par la bobine. Avec un rotor polarisé, on peut choisir une de ces deux dents en agissant sur le sens du courant dans la bobine [22].

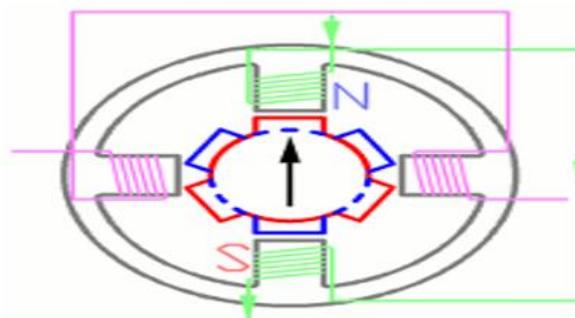


Figure -I.17- moteur pas à pas hybride [22]

I.9.2 Comparaison des trois catégories des moteurs pas à pas

Le tableau –I.1- représente les différents types de moteurs pas à pas et leurs avantages et inconvénients.

Tableau -I.1- les différents types de moteurs pas à pas :

Type de moteur	Moteur à aimant permanent	Moteur hybride	Moteur à reluctance variable
Résolution (Nombre de pas)	Moyenne	Elevée	Bonne
Couple moteur	Elevé	Elevé	Faible
Vitesse	Faible	Elevé	Elevé
Sens de rotation	Dépend : <ul style="list-style-type: none"> • De l'ordre d'alimentation des phases. • Du sens de courant dans les bobines. 	Dépend : <ul style="list-style-type: none"> • De l'ordre d'alimentation des phases. • Du sens de courant dans les bobines. 	Dépend : <ul style="list-style-type: none"> • De l'ordre d'alimentation des phases.
Fréquence de travail	Faible	Grande	Grande
Coût	Bas	Haut	Moyen/haut

I.10 LA COMMANDE DES MOCN

De l'idée de départ jusqu'à la fabrication de la pièce sur la machine CNC, tout va se passer sur l'ordinateur, utilisant différents logiciels selon l'étape du processus à effectuer, constituant ainsi une chaîne logicielle.

Les étapes de création :

- La conception 3D.
- La génération des chemins de l'outil, création du G-code.
- Interprétation du G-code et transformation en mouvement sur la machine.

I.10.1 La conception 3D

La conception permet d'élaborer, dessiner, visualiser un objet en trois (3) dimensions. La modélisation va plus loin, permettant de modéliser des mouvements des objets en relation les uns avec les autres, de simuler des efforts, etc. Autrement dit, tous les outils de conception 3D ne sont pas nécessairement aussi des outils de modélisation.

Il existe plusieurs logiciels de conception et modélisation 3D, tel que le logiciel Solid Works qui est le plus populaire.

I.10.2 La génération des chemins de l'outil, création du G-code

Après la conception en 3D de la pièce, par exemple avec le logiciel SolidWorks, on obtient un fichier de format STL, OBJ, Image ou autre. Pour la génération d'un fichier G-code il existe des logiciels spécialement conçu pour cela, tel que le logiciel Slic3r, Inkscape, Charlygraal et bien d'autre. Aussi, Il est possible de créer un fichier d'usinage directement à partir d'un site spécialement conçu pour cela, tel que Makercam.com.

I.10.2.1 Langue de contrôleurs numériques « G-code »

Le G-code est le langage de programmation pour contrôler une machine à commande numérique. Il est basé sur des lignes de code, plusieurs de ces lignes peuvent être regroupées dans un fichier pour créer un programme G-code, qui sert à programmer les mouvements que la machine va effectuer (droite, arc de cercle, ect). Il est à noter que le G-code est utilisé pour des fraiseuses à commande numérique, des tours, des imprimantes 3D et des lasers de découpe.

a- Les principales fonctions du G-code

Les principales fonctions du G-code sont résumées dans le tableau suivant (Tableau-I-2-).

Tableau -I.2- Les principales fonctions du 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
M00/M01	Arrêt du programme, arrêt optionnel ou avec condition
M06	Changement d'outil
M30	Fin du programme, réinitialisation, rembobinage

I.10.3 Interprétation du G-code et transformation en mouvement sur la machine

Après la génération d'un fichier G-code, il reste à l'interpréter de façon à pouvoir contrôler les moteurs avec un microcontrôleur plus précisément avec une carte Arduino. Pour cela il existe des logiciels et des micro-logiciels spécialement conçu pour cela, tel que le Grbl.

I.10.3.1 Description du micro-logiciel Grbl

Le Grbl est un micro-logiciel libre développé sur Arduino pour contrôler des graveuses CNC (Computer Numerical Control), des fraiseuses munis d'une tête mobile contrôlée en X, Y et Z par un ordinateur. Grbl interprète du G-code et déplace en conséquence un outil sur trois (3) axes (X, Y et Z). Il comprend de multiples optimisations sur l'usage et le déplacement des moteurs afin de gérer correctement les accélérations, les trajectoires, etc [23].

I.11 LA CONCEPTION ASSISTEE PAR ORDINATEUR (CAO)

I.11.1 Définition

La CAO (Conception Assistée par Ordinateur) est un ensemble d'outils et de programmes informatiques permettant d'assister l'ingénieur dans la conception et la mise au point d'un produit. Un système de CAO permet de représenter et d'étudier le fonctionnement d'un objet sans l'avoir fabriquer réellement, c'est-à-dire en virtuel [24].

I.11.2 Les outils de la CAO (logiciels)

Un grand nombre de logiciels de CAO sont disponibles en open-source :

- ✓ Art of Illusion.
- ✓ Blender.
- ✓ FreeCAD.
- ✓ ImplicitCAD.
- ✓ QCAD.
- ✓ pythonOCC.
- ✓ OpenCASCADE.
- ✓ Solidworks.

I.12 FABRICATION ASSISTEE PAR ORDINATEUR

I.12.1 Définition

La fabrication assistée par ordinateur ou FAO est d'écrire le fichier contenant le programme de pilotage d'une machine-outil à commande numérique. Ce fichier va décrire

précisément les mouvements que doit exécuter la machine-outil pour réaliser la pièce demandée. On appelle également ce type de fichiers : programme ISO ou blocs ISO.

Les logiciels de FAO utilisent les modèles et les ensembles créés dans les logiciels de CAO pour générer des trajectoires d'usinage sur lesquelles s'appuient les machines qui convertissent les conceptions en pièces physiques [25].

I.12.2 Les outils de la FAO (logiciels)

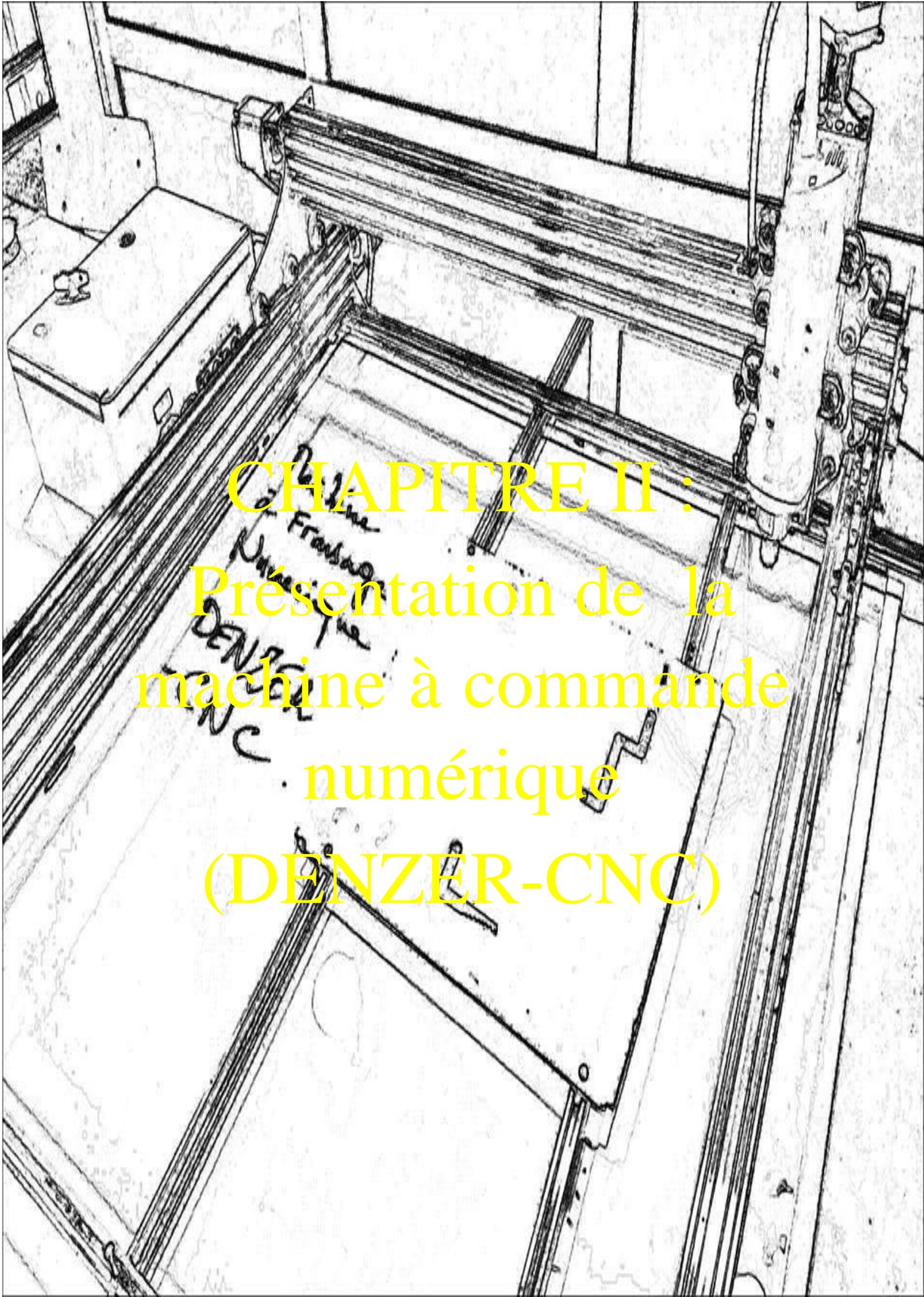
Les outils de la FAO sont des modeleurs, nous citons :

- ✓ Catia.
- ✓ RhinosCAM.
- ✓ SolidCAM.
- ✓ Power MILL.
- ✓ Hyper MILL.
- ✓ Camworks.

I.13 CONCLUSION

Ce premier chapitre s'appuie essentiellement sur quelques notions de la machine numérique, des moteurs pas à pas et de la commande numérique, elles sont nécessaires pour l'étude, l'automatisation et la réalisation d'une machine à commande numérique. Ces fondements sont importants pour tenir compte des contraintes que toute réalisation pourrait avoir.

Aujourd'hui, l'outil informatisé, appelé commande numérique, remplace de plus en plus l'équipement manuel et permet de programmer à l'avance les opérations à réaliser. Les pièces sont parfois fabriquées jusqu'au millième de millimètre.

A technical drawing of a CNC machine structure, showing a complex arrangement of beams, rails, and mechanical components. The drawing is rendered in black lines on a white background. Overlaid on the drawing is yellow text. In the background, there are handwritten notes in French: 'Machine à commande numérique', 'N° de fabrication', and 'DENZER-CNC'.

CHAPITRE II :
Présentation de la
machine à commande
numérique
(DENZER-CNC)

II.1 INTRODUCTION

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 une partie des fonctions de commande.

La fraiseuse verticale « DENZER-CNC » est une machine de haute précision, cette fraiseuse est destinée au fraisage des pièces très variées d'acier, de fonte, de bois et de métaux non ferreux. C'est une machine pilotée par des moteurs pas à pas. Ces derniers permettent de convertir directement un signal électrique numérique en un positionnement angulaire de caractère incrémental.

Ce chapitre présente une description de la machine DENZER-CNC, son principe de fonctionnement, son architecture et environnement logiciels et matériels et ses principales caractéristiques.

II.2 LE FRAISAGE NUMERIQUE

Le fraisage est l'un des plus importants développements de la technologie d'usinage.

II.2.1 La fraiseuse

Autrefois, toutes les machines pouvaient être simplement divisées en quatre catégories : horizontales et verticales, par tournage ou fraisage. Aujourd'hui, comme nous venons de le voir, les machines sont plus complexes et peuvent être de plusieurs types à la fois.

Les fraiseuses sont des machines-outils par enlèvement de copeaux, caractérisées par l'utilisation d'outils tournants, appelés forets de précision. Ils sont conçus pour usiner des surfaces planes, cylindriques, coniques et / ou hélicoïdales.

Les fraiseuses à une seule broche peuvent également être équipées d'arbres de fraisage spéciaux, permettant ainsi le travail simultané de plusieurs forets utilisés sur différentes surfaces.

II.2.2 Les différents types de fraiseuses

Nous pouvons classer les fraiseuses de la manière suivante : fraiseuses manuelles, fraiseuses par apprentissage et fraiseuses à commande numérique. Voyons ensemble les différents types un par un :

II.2.2.1 Fraiseuse manuelle

Les différents mouvements de la machine sont commandés par l'opérateur. Il effectue les différents réglages.

II.2.2.2 Fraiseuse par apprentissage

Elle permet de mémoriser les mouvements exécutés par l'opérateur et de les reproduire ensuite automatiquement.

II.2.2.3 Fraiseuse à commande numérique

Elle dispose d'un organe de contrôle informatique (automate programmable ou base PC) où sont enregistrés les différents ordres nécessaires à la fabrication d'une pièce. Lorsque le programme est lancé, la machine effectue automatiquement toutes les opérations.

II.2.2.3.1 La fraiseuse à trois (3) axes

La plupart des fraiseuses à commande numérique sont des machines trois (3) axes ; de ce fait, le mouvement de l'outil de coupe est limité, ce qui rend impossible d'usiner des dépouilles négatives ou de réaliser des formes sur plusieurs plans orthogonaux.

a. Ce qu'il est possible de faire avec une fraiseuse trois (3) axes

- Des découpes sur des surfaces 2D ou 3D.
- Des gravures sur des surfaces 2D ou 3D.
- Créer des formes 3D sans dépouilles négatives.

b. Ce qu'il est impossible de faire avec une fraiseuse trois (3) axes

- Des usinages nécessitant un axe d'outil perpendiculaire à plusieurs plans.
- Des usinages de surfaces avec dépouilles négatives.

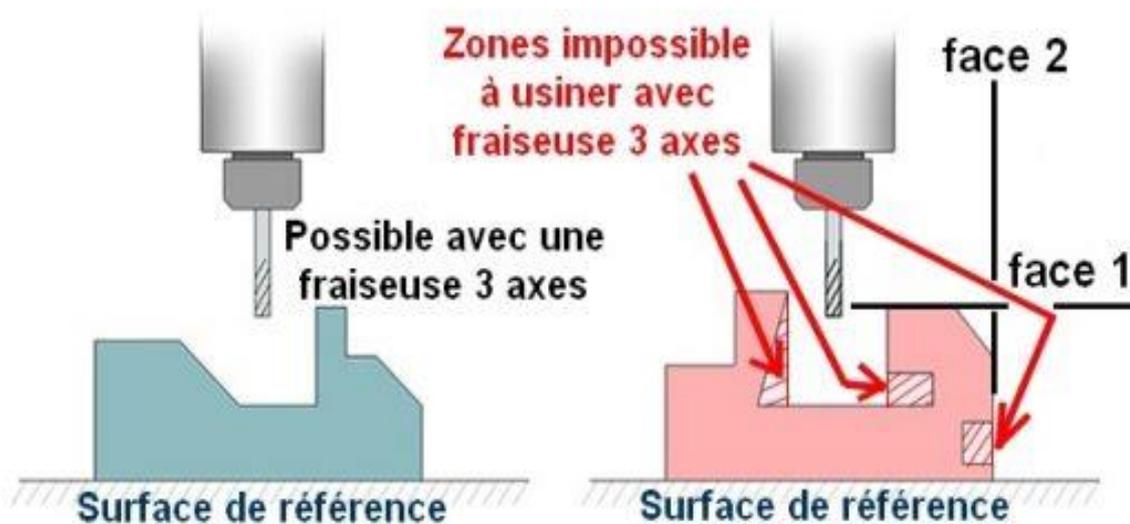


Figure-II.1- Ce qui est possible de faire avec une fraiseuse trois (3) axes.

II.2.2.3.2 Fraiseuse à quatre (4) axes

Une fraiseuse quatre (4) axes est tout simplement une fraiseuse trois (3) axes dotée d'un plateau tournant. C'est sur ce plateau tournant que va être fixée la matière à usiner, l'outil de coupe peut ainsi travailler sur toutes les faces sans avoir à intervenir pour retourner la matière, le 4ème axe se charge de cela au fur et à mesure selon les commandes du programme.

Le 4ème axe est souvent utilisé soit en mode « indexation », soit en mode « continu ». En mode indexation, la broche est désactivée et aucune coupe n'est faite jusqu'à ce que l'axe quatre (4) achève son mouvement de rotation, en mode continu par contre, l'outil de coupe travaille en même temps que l'axe 4 tourne.

II.2.2.3.3 Fraiseuse à cinq (5) axes

Une fraiseuse à cinq (5) axes comporte toujours trois axes linéaires (X, Y, Z), mais aussi deux axes rotatif qui peuvent varier selon le modèle choisi. Il existe principalement deux modèles qui sont les deux axes sur la tête ou les deux axes sur la table.

II.2.2.4 Autres types de fraiseuses

Il existe d'autres types de fraiseuse, telles que :

II.2.2.4.1 Fraiseuses horizontales

Elles sont appelées ainsi car l'axe de la broche est horizontal (figure-II-2). Ce qui permet aux coupeaux de tomber et de ne pas rester sur l'élément à usiner, de manière plus générale, avec cette technique, on obtient une pièce de meilleure qualité.

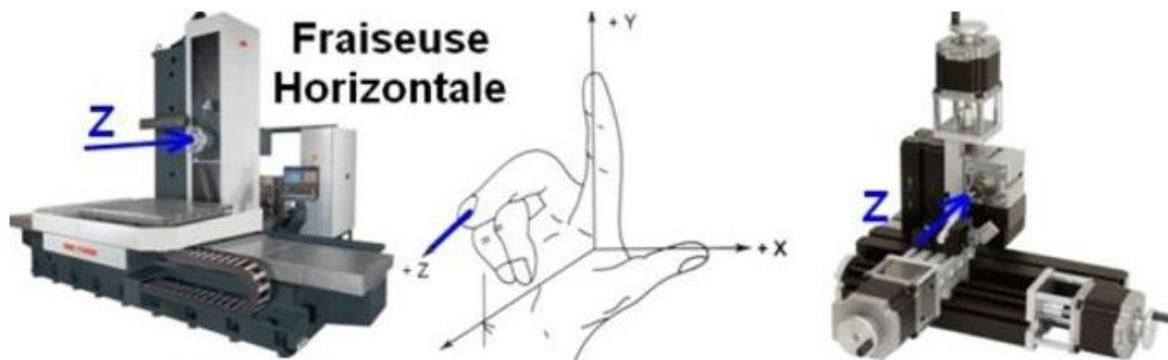


Figure –II.2- Fraiseuse horizontale.

II.2.2.4.2 Fraiseuses verticales

Ces fraiseuses quant à elles, ont une broche positionnée verticalement (figure-II-3). Le porte-outil est équipé d'un mouvement d'avance et se déplace sur deux glissières perpendiculaires.

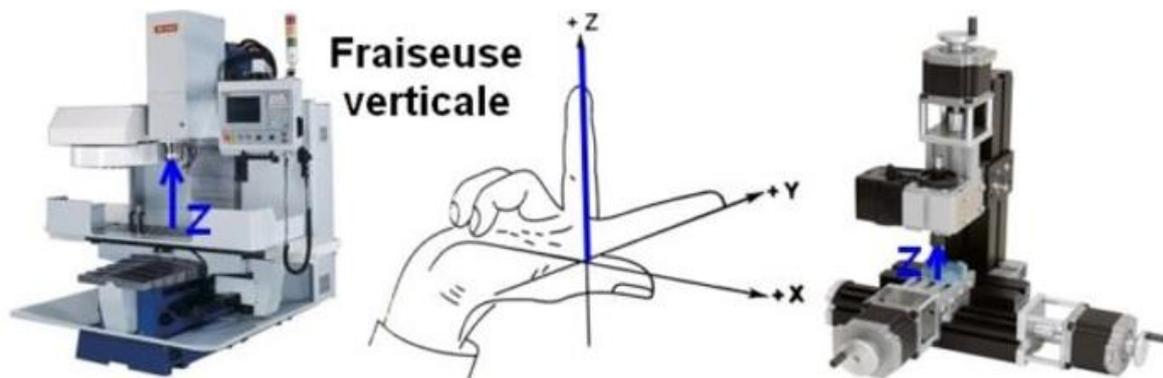


Figure –II.3- Fraiseuse verticale.

II.2.2.4.3 Fraiseuses spéciales

Comme son nom l'indique, ces fraiseuses sont utilisées pour des processus particuliers, tels que la réalisation de clés, de sculptures ou de pièces avec rainures.

II.2.2.4.4 Fraiseuses universelles

Les fraiseuses universelles sont celles qui combinent les fonctions des fraiseuses horizontales et verticales. Dans les fraiseuses universelles, la table porte-pièces peut également être rotative.

II.2.2.4.5 Fraiseuses automatiques à contrôle numérique

Ces machines sont utilisées pour le traitement de profils ou de surfaces très complexes. Les fraiseuses automatiques sont équipées d'une pointe de sonde qui reste toujours adhérente au modèle (fraiseuses à copier), guidant le couteau dans ses mouvements.

II.3 NECESSITE DU DEVELOPPEMENT DE LA MACHINE A COMMANDE NUMERIQUE A TROIS (3) AXES

Pour notre projet de fin d'étude, l'entreprise Denzer Technologies nous a demandé de commander et de réaliser la carte de commande d'un centre d'usinage en respectant le cahier de charge suivant:

- ✓ Centre d'usinage vertical de production à architecture ouverte de trois (3) axes numérisés avec usinage en continue avec des moteurs pas à pas de type NEMA 23.

Le centre d'usinage à commande numérique à trois (3) axes est une machine CNC pour la découpe et la gravure des matériaux 2D et 3D.

Pourquoi, alors, développer la machine à commande numérique a trois (3) axes ?

La machine convient à de nombreuses applications. Ingénieur, fabricant, et même les chercheurs peuvent augmenter leur productivité et leur service avec la possibilité de fabriquer des produit, par découpe CNC, perçage de précision et gravure de métaux tendre, de bois dur, de plastique et de mousse haute densité.

II.4 DESCRIPTIF DU PRINCIPE DE FONCTIONNEMENT DE LA MACHINE DENZER-CNC

La fraiseuse à commande numérique (Fraiseuse DENZER-CNC) est une machine d'usinage à cycle automatique programmable. Le terme "commande numérique" est générique et a été retenu parce que la machine est commandée par des "consignes" numériques fournies par un calculateur. En d'autres termes, on peut dire que les organes mobiles de la machine sont motorisés et qu'un automatisme assure la commande et dans la plupart des cas le contrôle de la position et/ou de la vitesse.

Le fonctionnement de la fraiseuse DENZER-CNC peut se diviser en deux parties :

- La partie opérative (la fraiseuse agit directement sur le produit à réaliser).
- La partie commande (La commande numérique permet d'élaborer des ordres en fonction des consignes et des comptes rendus).

II.4.1 La partie opérative

Ce type de fraiseuse se compose ainsi deux mouvements de travail, rotation de la broche et translation des axes. Ces derniers sont complémentaires.

La rotation de la broche provient d'un moteur électrique.

Le mouvement de translation comprend trois mouvements : l'avance transversale, longitudinale et verticale.

- L'avance transversale du chariot s'effectue par le système vis infinie, cette dernière est fixée dans le support à un moteur électrique.
- Le mouvement longitudinal se fait par le système de deux vis infinies, ces derniers sont fixés dans le support à deux moteurs électriques.
- Le mouvement vertical s'effectue de la même façon que celui du mouvement longitudinal.

La pièce à usiner est serrée sur la table de la machine-outil et le mouvement de coupe générant l'enlèvement de copeaux est produit par la rotation de l'outil de coupe que l'on appelle une fraise par rapport à la pièce à usiner. Le mouvement de la fraise tout autour de la pièce à usiner est appelé mouvement d'avancement. Le mouvement d'avancement donne la forme voulue de la pièce à usiner (figure-II-4-).

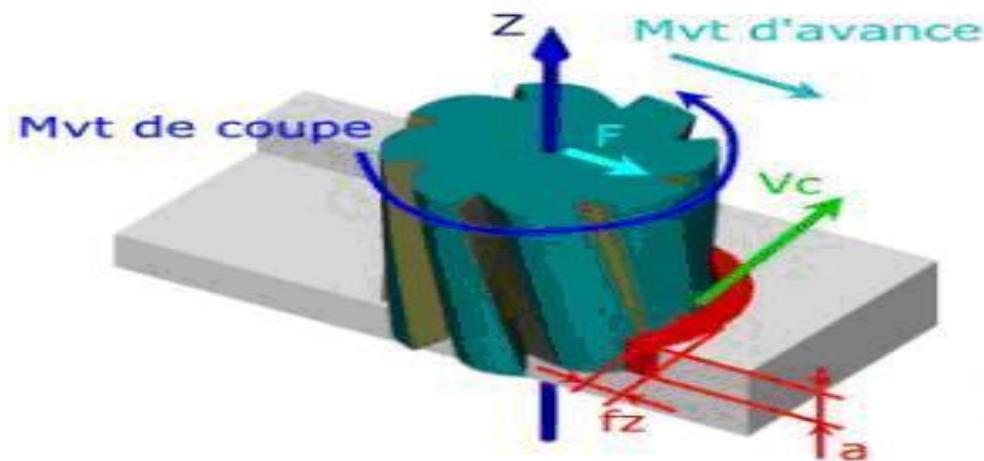


Figure-II.4- Principe du fraisage.

II.4.2 La partie commande

DENZER-CNC est une machine à commande numérique, ce qui veut dire que la machine est commandée par des consignes numériques fournies par l'armoire de commande. Cette dernière est constituée d'une carte Arduino, d'une carte Shield, trois (3) Drivers moteurs et d'une alimentation.

Depuis le développement des ordinateurs personnels, des programmes de FAO (fabrication assistée par ordinateur, ou en anglais CAM, Computer Aided Manufacturing) ont été développés afin d'écrire les séquences de code ISO automatiquement à partir de fichiers de dessin et des paramètres des outils finaux. Les interventions directement dans le code sont devenues marginales.

Pour vérifier le bon fonctionnement des séquences d'usinage, il existe des programmes de simulation d'usinage permettant une visualisation de la pièce telle qu'elle sera conçue lors de l'usinage.

La séquence de fabrication complète est donc :

CAO (conception assistée par ordinateur) → FAO (développement des séquences) → Simulation de coupe → Exécution de la fabrication.

II.5 LES MATIERES D'ŒUVRE DE LA MACHINE DENZER-CNC

Comme toutes les autres fraiseuses, DENZER-CNC est capable de traiter différentes surfaces et matières, telles que :

- Aluminium.
- Bois.
- Verre en plexi.
- Delrin (polyoxymethylene).
- HDPE (High Density Polyethylene).
- Mousse.

II.6 LA MACHINE A COMMANDE NUMERIQUE « DENZER-CNC »

DENZER-CNC est une fraiseuse à commande numérique, qui consiste à enlever de la matière sur une pièce initiale pour obtenir une pièce finale, à l'aide d'un outil coupant accroché à une broche, se déplaçant dans l'espace suivant trois (3) axes.

II.6.1 Les principaux modules de la machine DENZER-CNC

DENZER-CNC est formée par un ensemble de composants électriques et mécaniques de deux (2) parties complémentaires, une partie commande et une partie opérative.

II.6.1.1 Les éléments de la partie commande

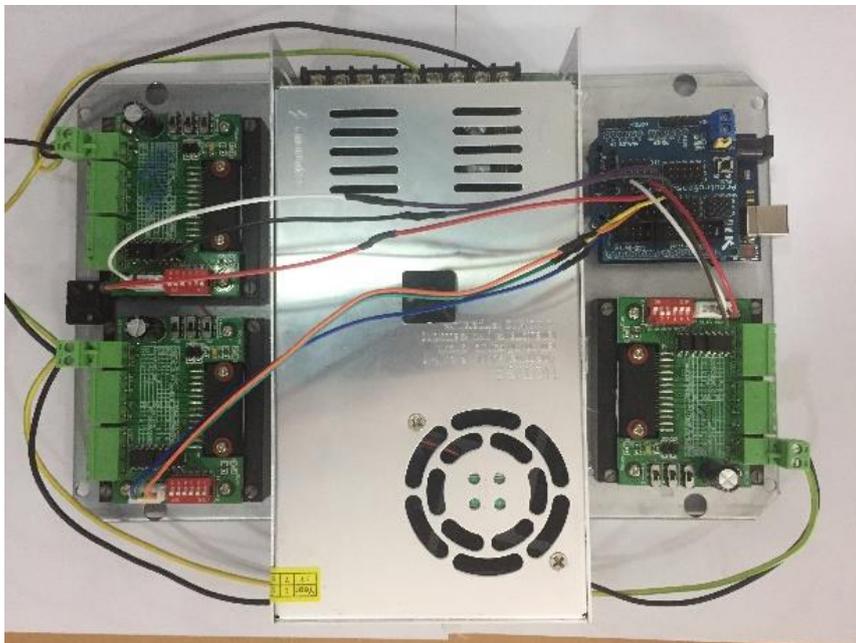


Figure-II.5- Armoire de commande.

II.6.1.1.1 Carte Arduino

La carte Arduino est une carte électronique basée autour d'un microcontrôleur et de composants minimum pour réaliser des fonctions plus ou moins évoluées à bas coût. Elle possède une interface USB pour la programmer.

Arduino peut être utilisé pour développer des applications matérielles industrielles légères ou des objets interactifs, et il peut recevoir en entrées une très grande variété de capteurs. Les projets Arduino peuvent être autonomes, ou communiquer avec des logiciels sur un ordinateur.

Il existe différent type et modèle d'Arduino, allant du Nano au Mega, avec différent caractéristique, comme le nombre de pins et les fonctionnalités prédéfini.

Dans notre cas l'Arduino Uno est le plus appropriier à la machine DENZER CNC (figure-II-6).

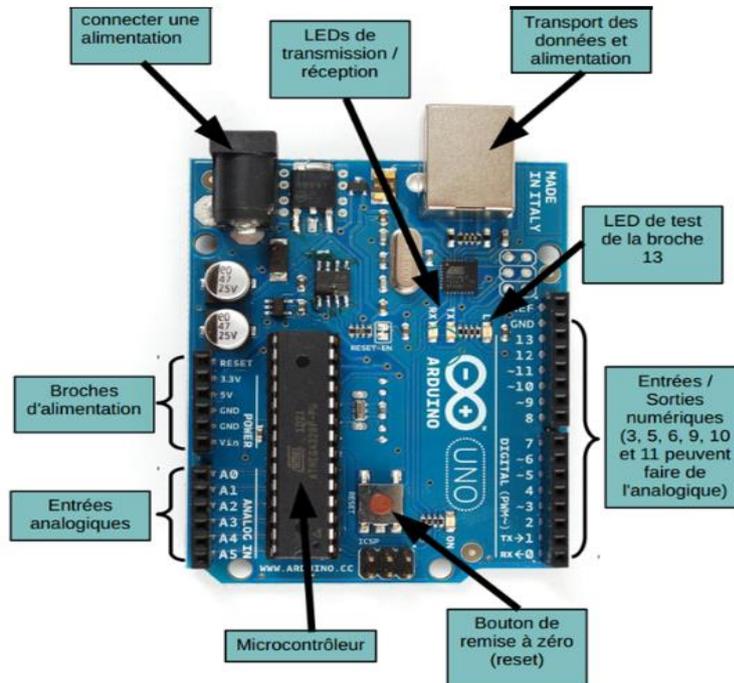


Figure-II.6- Carte Arduino Uno.

II.6.1.1.2 Carte Arduino Sensor Shield v5.0

L'Arduino Sensor Shield V5.0 (figure-II-7) est le nouveau blindage empilable qui répartit les différentes broches d'E/S de l'Arduino sur des connecteurs plus robustes que les broches de la carte Arduino, ce qui permet de connecter divers modules comme des capteurs, des servos, des relais, des boutons, des potentiomètres et beaucoup plus de façon beaucoup plus facile. Les ports d'E/S numériques comprennent les broches numériques 0 à 13. Les ports d'E/S analogiques comprennent les broches analogiques A0 à A5. Chacune de ces rangées d'E/S possède une broche de mise à la terre, de tension et de signal dédiée. Un commutateur de réinitialisation est également fourni. Il prend en charge différents types d'interfaces : IIC, port de servocommande, port Bluetooth, emplacement de carte SD, port de module APC220, port de capteur à ultrasons.

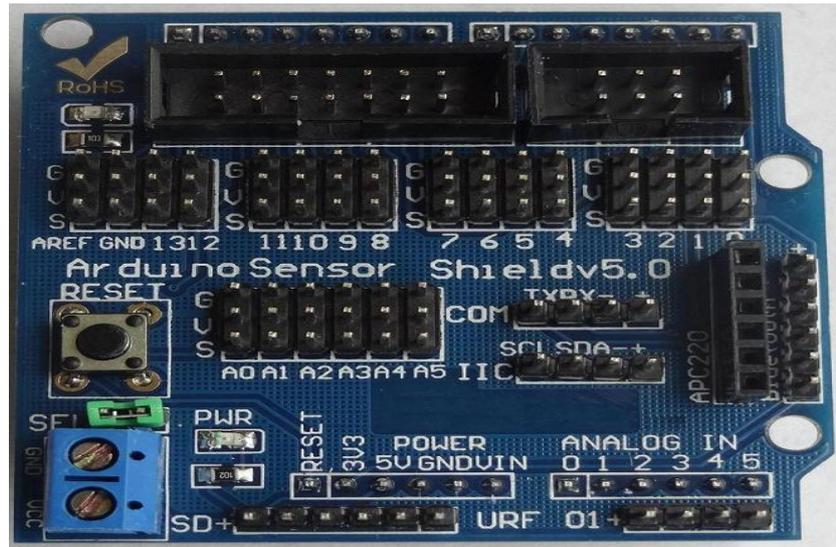


Figure-II.7- Carte Arduino Sensor Shield v5.0.

II.6.1.1.3 Driver ST330-v3

Le ST330-v3 (figure-II-8) est un pilote monophasé bipolaire compact à moteur pas à pas biphasé 3.0A par phase avec des entrées de pas et de direction, commutateur DIP réglable en mode pas à pas (plein, demi, huitième et seizième S3 / S4), couple (100%, 75%, 50 %, 20% S5 / S6), fonctionnent en mode courant (0-3A SW1 / SW2 / SW3) et décroissent (0%, 25%, 50% et 100% S1 / S2).

Convient pour toute sorte d'équipement et d'appareils d'automatisation de petite et moyenne taille.

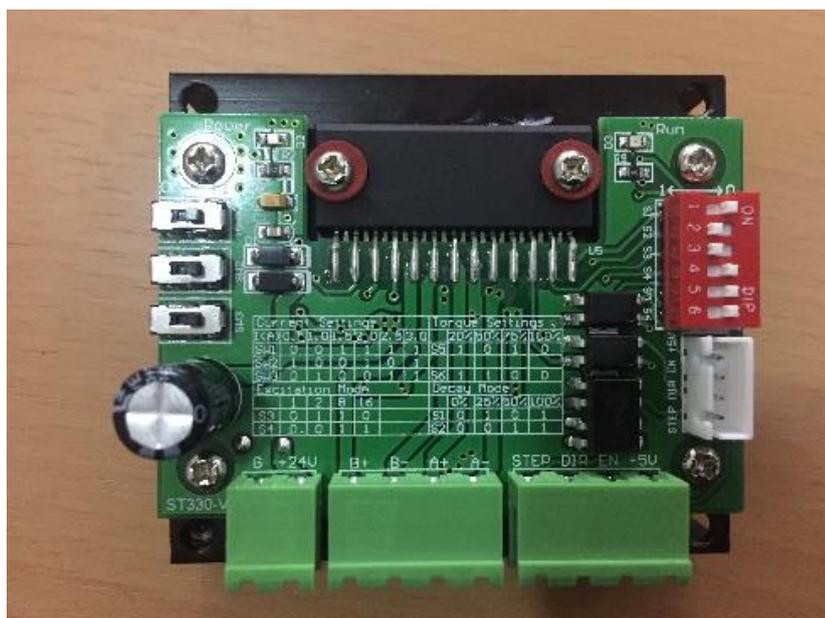


Figure-II.8- Driver ST330-v3.

II.6.1.1.4 Alimentation

- ❖ Commutateur d'alimentation, convertir AC 110V/220V à DC 24V.
- ❖ Puissance : 480W.
- ❖ Tension de sortie : 24V.
- ❖ Courant de sortie : 20A.
- ❖ Rendement > 85%.
- ❖ Matière : aluminium.
- ❖ Protections : court-circuit/surcharge/température.



Figure-II.9- Boite d'alimentation.

II.6.1.2 Les éléments de la partie opérative

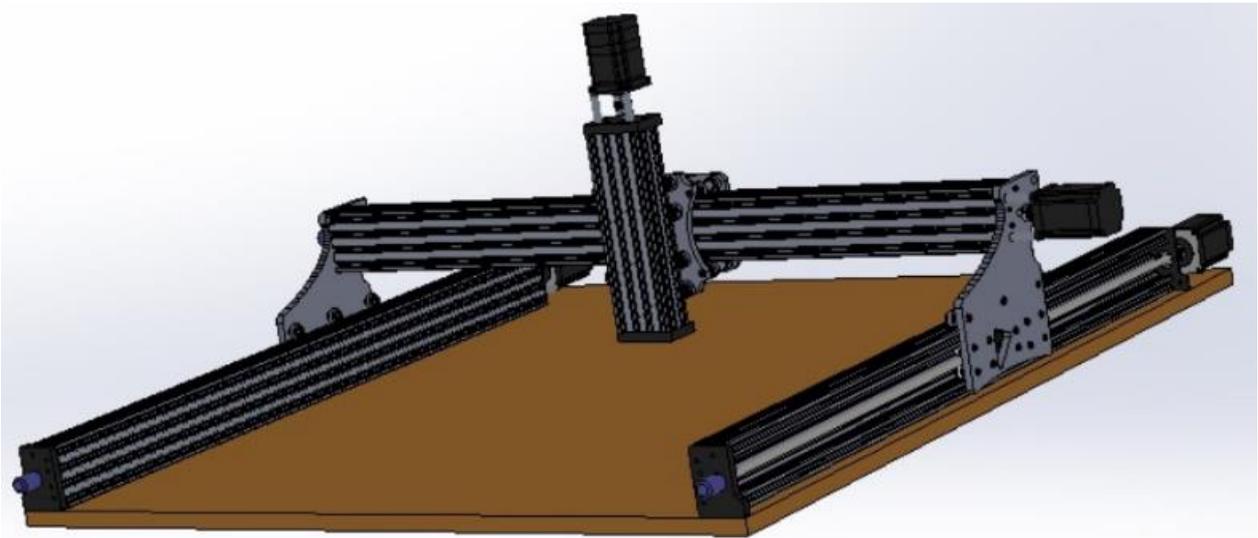


Figure-II.10- Partie opérative.

II.6.1.2.1 Moteurs pas à pas

Les moteurs pas à pas NEMA 23 sont les moteurs les plus appropriés à la machine DENZER-CNC, c'est un moteur hybride bipolaire (figure-II.11) avec un couple beaucoup plus élevé que les moteurs pas à pas conventionnels.



Figure-II.11- Moteur pas à pas NEMA 23.

a. Caractéristique du moteur NEMA 23

- Modèle : 57BYGH627
- Haute qualité
- Couple de maintien : 190Ncm
- Angle de pas : 1.8 ° (200 pas par tour)
- Voltage : 3V
- 2 phases
- Courant par phase : 3A
- Résistance de phase : 0.65ohm
- Inductance de phase : 1.5mH
- Longueur : 76mm
- Largeur : 56.4mm

II.6.1.2.2 La broche BOSCH GGS 28 C PROFESSIONAL

On utilise généralement différents types de broche pour l'usinage, les Dremel, les défonceuses, les broches spéciales comme les Kress et les broches BOSCH (figure-II.12).



Figure-II.12- Broche BOSCH GGS 28 C PROFESSIONAL.

a. Caractéristiques techniques de la broche

- Vitesse de rotation nominale : 28000 tr/min
- Puissance absorbée nominale : 600 W
- Puissance débitée : 350 W
- Porte-outils : Pincettes de serrage jusqu'à \varnothing 8 mm
- Diamètre collet de broche : 43 mm
- Diamètre maxi. pince de serrage : 8 mm
- Diamètre de meule maxi : 50 mm
- Méplat écrou de serrage : 19 mm²
- Méplat broche : 19 mm²
- Poids : 1,4 kg
- Interrupteur : 2 voies.

II.7 ENVIRONNEMENT DE TRAVAIL

La machine DENZER-CNC est le fruit du projet fin d'étude « Réalisation et Automatisation d'une machine à commande numérique », projet proposé et réalisé au niveau de l'entreprise DENZER Technologies.



CHAPITRE II : PRESENTATION DE LA MACHINE A COMMANDE NUMERIQUE (DENZER-CNC)

L'entreprise DENZER Technologies est une jeune entreprise innovante activant dans le développement informatique et électronique, créée en 2016. Elle est subdivisée en deux unités :

- ❖ Unité Recherche et Développement (Laboratoire et atelier) : Cité Aouchiche, Route de Boukhiana, Béjaia.



Figure-II.13- Situation géographique de DENZER Technologies – Bejaia.

- ❖ Siège social (Service administratif et commercial) : Sis au Boulevard Aissat Idir, Akbou, Béjaia.



Figure-II.14- Situation géographique de DENZER Technologies – Akbou.

L'entreprise DENZER Technologies conçoit des sites web, des logiciels, des applications pour montres intelligentes ainsi que des applications fonctionnant sur des Smartphones.



Figure -II.15- Conception sites web.

L'entreprise réalise et produit des systèmes électroniques complets visant à mieux commander des machines, des appareils et des objets domestiques, professionnels et dans les usines industrielles, fermes agricoles et établissements administratifs, à titre d'exemple : des armoires électriques pour la commande des ascenseurs, un système de gestion de file d'attente, des bracelets électroniques pour le suivi des malades Alzheimer.



Figure -II.16- Robotisme et systèmes électroniques.

L'entreprise innove dans l'impression 3D, elle a fabriqué une imprimante pour le prototypage rapide, qui porte une option en plus de celles existantes sur le marché.

Plusieurs systèmes sont en cours de développement, il s'agit de dongles pour la transmission automatique des SMS, un système de sécurité et d'alarme, un système de géolocalisation des engins et des véhicules, des chargeurs de batteries de véhicules ainsi que celles des PC portables et des téléphones mobiles.

DENZER Technologies fournit aux entreprises et aux particuliers différents services à savoir :

- ❖ Installation, maintenance, entretien et rénovation des ascenseurs.
- ❖ Impression 3D des prototypes avec des imprimantes 3D fabriquées par les ingénieurs de l'entreprise.
- ❖ Installation des réseaux informatiques et des standards téléphoniques.
- ❖ Formations pratiques en informatique et en électronique.



Figure-II.17- Développement et maintenance.

Données sur l'entreprise

- Raison sociale : EURL DENZER TECHNOLOGIES.
- Adresse du siège social : Boulevard Aissat Idir, Akbou, Wilaya de Béjaia.
- Adresse de l'Unité Recherche et Développement : Cité Aouchiche, Route de Boukhiana, Béjaia.
- Forme juridique : Entreprise Unipersonnelle à Responsabilité Limitée (EURL).
- Capital social : 1.000.000,00 DA soit un million de dinar algérien.
- Banque de domiciliation : BDL Akbou.
- Fondateur : Dr LADDI Samir.
- Gérant : Dr LADDI Samir.

II.8 LES CARACTERISTIQUES DE LA MACHINE DENZER-CNC

Le tableau suivant (tableau-II.1) représente les caractéristiques de la machine DENZER-CNC

Tableau -II.1- Les caractéristiques de la machine DENZER-CNC.

Dimensions de la surface de la table	
<ul style="list-style-type: none"> • Longueur (cm) • Largeur (cm) 	<p style="text-align: center;">97</p> <p style="text-align: center;">97</p>
Encombrement de la machine	
<ul style="list-style-type: none"> • Longueur (cm) • Largeur (cm) • Hauteur (cm) 	<p style="text-align: center;">110</p> <p style="text-align: center;">110</p> <p style="text-align: center;">128</p>
Nombre d'étaux	2
Taille des étaux (cm)	96
Nombre de vitesse de la broche	1
La vitesse de la broche	
<ul style="list-style-type: none"> • A vide (tr/mn) • Avec fraise (tr/mn) 	<p style="text-align: center;">30000</p> <p style="text-align: center;">28000</p>
Distance entre le nez de la broche et la surface de la table	
<ul style="list-style-type: none"> • Minimale (mm) • Maximale (mm) 	<p style="text-align: center;">10</p> <p style="text-align: center;">260</p>
Déplacement maximale longitudinale de la table (cm)	90
Déplacement maximale transversale de la table (cm)	90
Déplacement maximale verticale de la broche (cm)	25
Poids (kg)	35

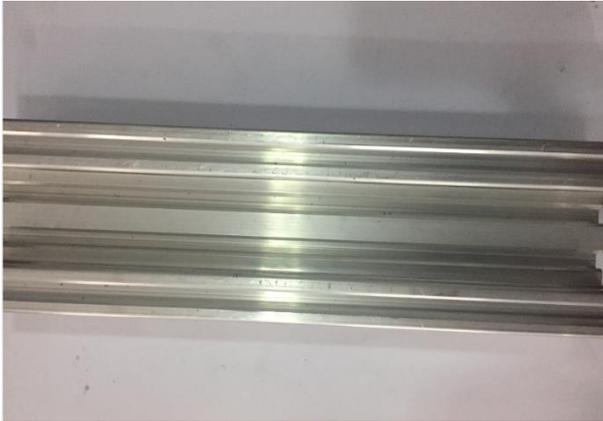
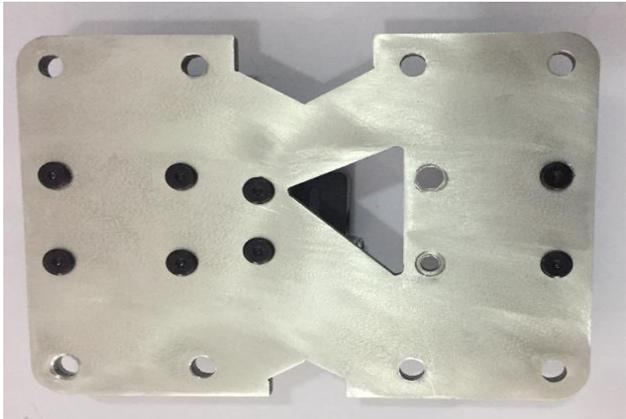
Poids maximale de la pièce usiné (kg)	20
Moteur du mouvement : <ul style="list-style-type: none"> • type • Puissance (W) • Nombre de pas par tour 	Nema 23 « 57BYGH627 » 80 200 (1.8° par pas)
La broche de fraisage : <ul style="list-style-type: none"> • type • Puissance absorbée (W) • Puissance débitée (W) • Nombre de tours par minute 	BOSCH GGS 28 C Professionnal 600 350 28000
Puissance sommaire des moteurs (W)	670

II.9 ARCHITECTURE MATERIEL DE LA MACHINE DANZER-CNC

Les composants matériels et mécaniques de la machine DENZER-CNC sont résumés dans le tableau suivant :

Tableau -II.2- Les composants matériels mécanique de la machine DENZER-CNC.

Composants	Description	Nombre
Vis sans fin-écrou 	<p>La vis sans fin-écrou est une tige filetée qui permet à l'aide d'un écrou de transformer la vitesse de rotation de moteur en un mouvement de translation, et aussi elle assure des déplacements de grande précision.</p>	<p>Quatre, un pour chaque uns des axes (X et Z) et deux pour l'axe (Y).</p>

<p>Accouplement</p> 	<p>En mécanique, un accouplement ou joint de transmission est un dispositif de liaison entre deux arbres en rotation, permettant la transmission du couple.</p>	<p>Quatre (4), un pour chaque moteur.</p>
<p>Rail de guidage</p> 	<p>Elle permet de guider le mouvement de translation des roulements selon les trois axes de la machine. L'axe Y est constitué de deux rails parallèle tandis que l'axe X et Z contiennent un seul rail chaque un.</p>	<p>Quatre (4), une pour l'axe (X), deux pour l'axe (Y) et une dernière pour l'axe (Z).</p>
<p>Plaques d'axes X, Y, Z</p> 	<p>Profile en aluminium sert à porter les roues de déplacement et la vis sans fin-écrou, elle assure aussi la liaison entre les rails de guidage X, Y et Z.</p>	

		
<p>Plaques de fixation</p> 	<p>Considérer comme un support qui sert à fixer le moteur sur le rail.</p>	<p>Deux (2), un à chaque extrémité de l'axe (Z).</p> <p>Quatre (4), un à chaque extrémité des deux rails de mouvement de l'axe (Y).</p>

<p>Espaceur excentriques</p> 	<p>Cet écrou est particulier, car il sert à faire une connexion très forte et scrute entre la rail et la roue.</p>	<p>Cinquante (50).</p>
<p>Ecrou</p> 	<p>Toute pièce ayant un trou taraude fait fonction d'écrou. Par l'intermédiaire d'une tige filetée, un écrou peut servir pour assemblage.</p>	<p>Indéfini.</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, elle se place :</p> <ul style="list-style-type: none"> ➤ entre l'écrou et la pièce à serrer. ➤ entre la tête de la vis et la pièce à serrer. 	<p>Indéfini.</p>

Collier verrouillage



Idéal pour une utilisation sur une tige filetée ou arbre lisse.

Huit (8), un à chaque extrémité d'une tige filetée.

Écrou pour vis métrique



C'est un dispositif indispensable pour fixer directement sur les plaques de mouvement des axes, il sert à maintenir et canaliser la tige filetée pour crée un mouvement linéaire.

Quatre (4), un pour chaque tige filetée.

Vis de fixation



Ils servent à relier et fixer les pièces entre elles, on trouve différents vis suivant leurs tailles, sections et leurs utilités dans la machine. les caractéristiques des vis sont les suivantes :

Indéfini.

- Alliage d'acier
- Finition noire
- Epaisseur de la tête : 1.5mm
- Diamètre : M5
- Pas du filet : 0.8
- Tête 6 pans creux (Allen) de 3mm
- Dureté : 6.8

Moteur pas à pas



Un moteur pas à pas permet de transformer une impulsion électrique en un mouvement angulaire avec une bonne précision de déplacement. Pour le bon fonctionnement de la machine on a opté pour l'utilisation MOTEUR PAS-A-PAS NEMA 23 1.9N.M 1.8° 78MM.

Quatre (4), un pour l'axe (X), deux (2) pour l'axe (Y) et un (1) pour l'axe (Z).

Les roues



C'est l'assemblage entre deux roulements, la rondelle et Roue OpenBuilds Solid V Wheel.

- La Roue OpenBuilds Solid V Wheel est une roue solide qui résiste à la déformation grâce à sa robustesse.
- Les roulements sont des organes mécaniques servant à guider un arbre de transmission en rotation.

Quarante-huit (48).

II.10 ENVIRONNEMENT LOGICIEL DE LA MACHINE DENZER-CN

Les logiciels utilisés dans la commande numérique de notre machine sont :

II.10.1 SolidWorks



SolidWorks est un logiciel propriétaire de conception assistée par ordinateur 3D fonctionnant sous Windows, générant un fichier de format STL ou OBJ.

La fenêtre principale de conception et de dessin de SolidWorks se présente sous la forme suivante :

Au lancement du logiciel SolidWorks, l'écran d'accueil suivant apparaît (figure-II.18) :

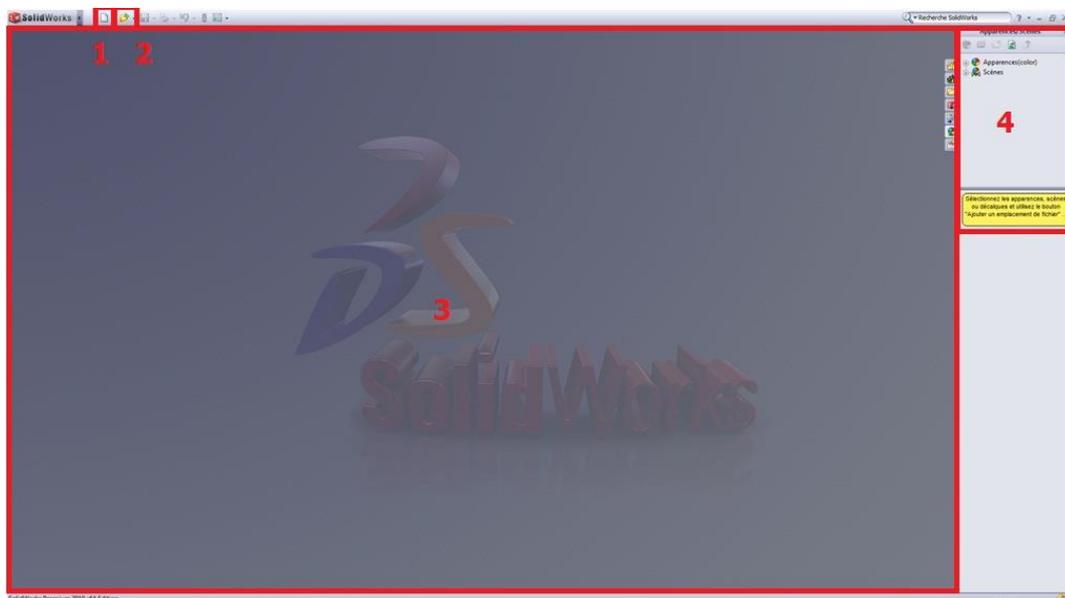


Figure-II.18- Fenêtre au lancement de SolidWorks.

Les principales parties constituant l'interface de **SolidWorks**, sont :

1. Nouveau fichier (Pièce, Assemblage, ...)
2. Ouvrir un projet existant
3. Zone de dessin
4. Apparence/Scène

Pour créer un nouveau projet, il suffit de cliquer sur "Nouveau". Une fenêtre apparaît pour nous demander quelle sorte de projet nous voulons créer (figure-II.19-).

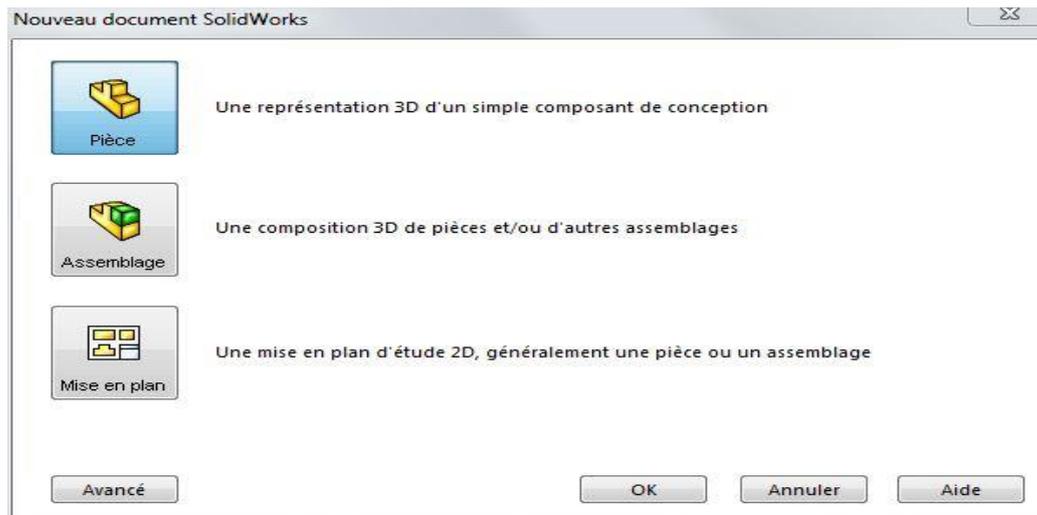


Figure-II.19- Nouveau document SolidWorks.

Les différents choix sont :

- Créer une pièce.
- Créer un assemblage.
- Créer une mise en plan.

Pour la création d'une pièce, on choisit donc « Créer une pièce », puis « OK ». L'interface garnie d'onglets suivante apparaît (figure-II.20-).

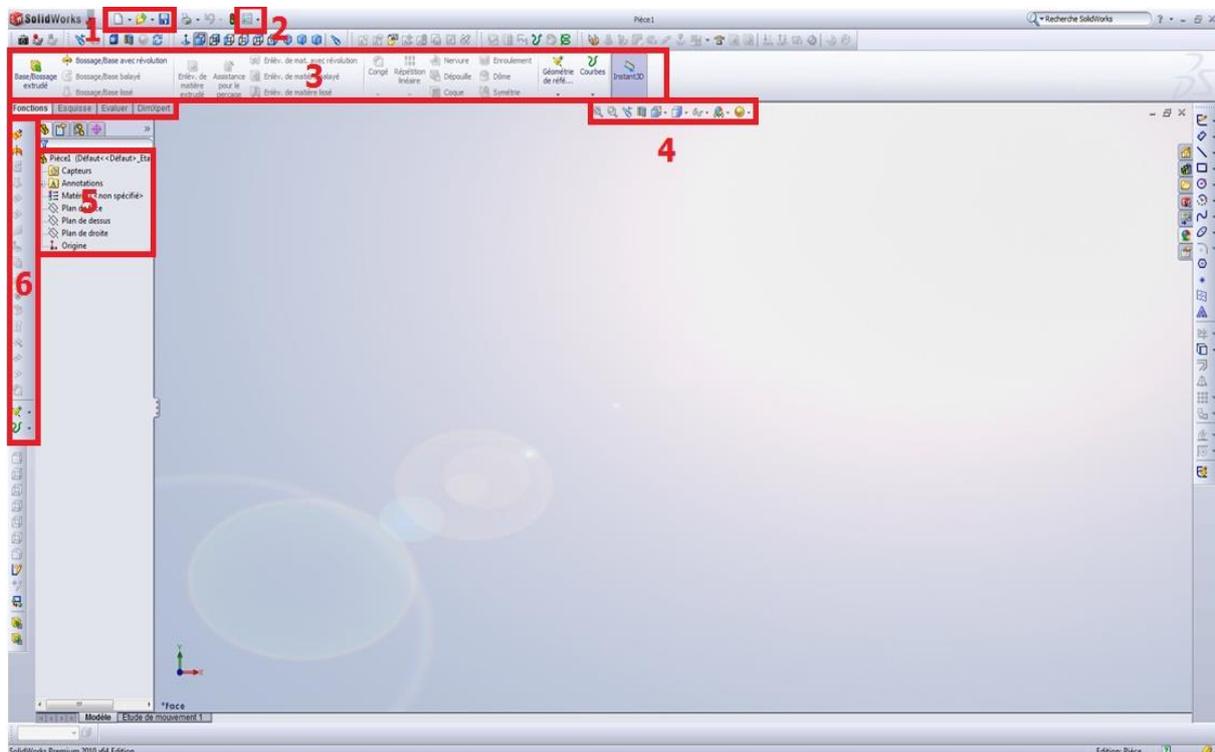


Figure -II.20- L'interface de création de pièce.

1. les boutons "Nouveau", "Ouvrir", et "Enregistrer".
2. Les Options
3. Le bandeau principal.
4. Diverses icônes : Vue en coupe, Zoom, Apparence, Vues, ...
5. L'Arbre de conception.
6. Un bandeau auxiliaire, utilisé pour les surfaces.

Une fois la pièce conçue, on appuie sur « Enregistrer » pour enregistrer et exporté la pièce en format .STL.

II.10.2 Netfabb

Après la conception d'une pièce au format, les pièces résultantes sont généralement remplies d'erreurs. On doit donc, réparer le fichier STL.

Le premier point essentiel pour réparer un fichier STL est de disposer des bons outils. NetFabb en fait partie. Dans notre cas, la version « Basic » gratuite suffit amplement pour réparer un fichier STL.

Une fois lancé, il suffit de charger un fichier (figure-II.21-).

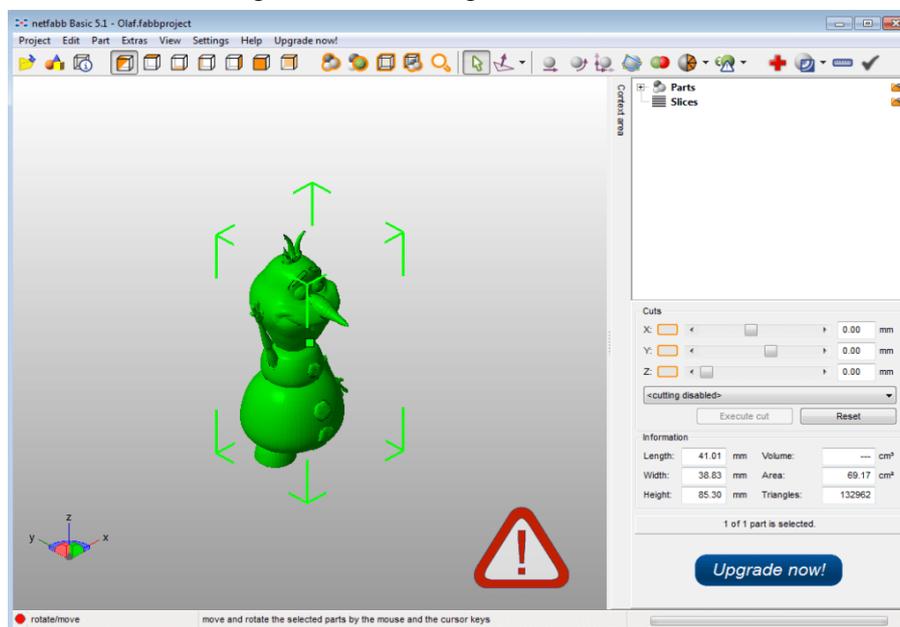


Figure -II.21- Interface Netfabb.

Le gros point d'exclamation nous informe que le fichier comporte des erreurs, pour réparer ces erreurs :

On clique sur la croix rouge, en haut à droite (figure-II.22-).

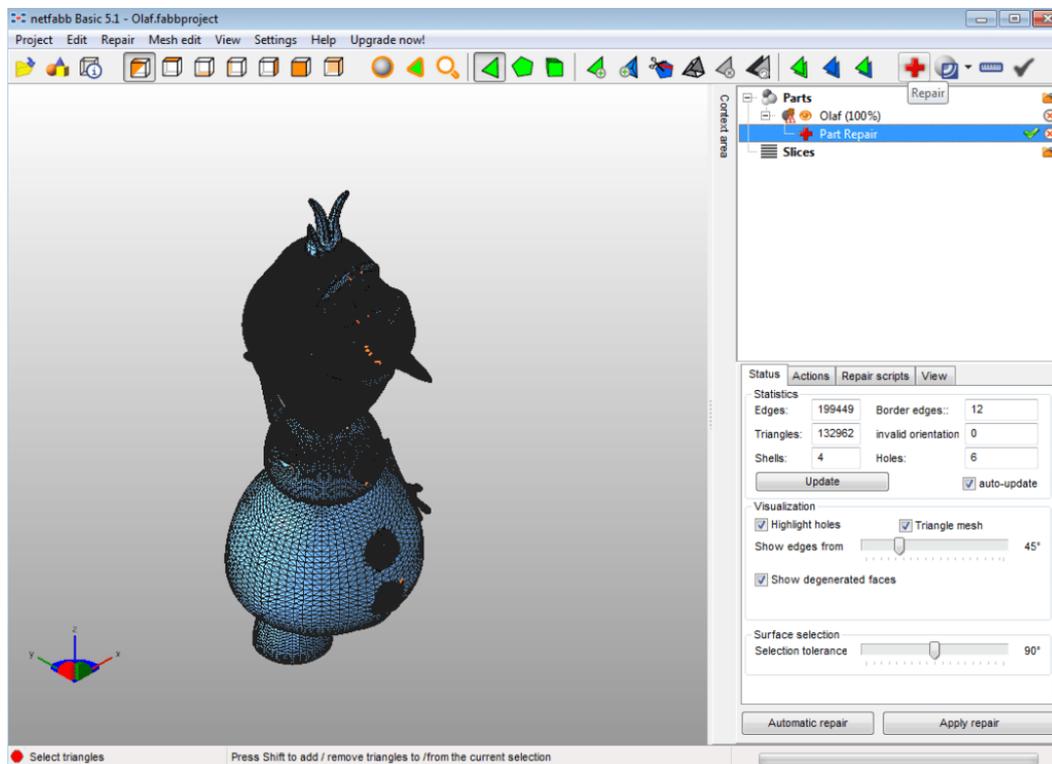


Figure -II.22- Réparations des erreurs.

Un maillage du fichier en noir et une mise en évidence des erreurs de maillage en orange, apparait sur la fenêtre.

Pas besoin de corriger les erreurs unes à unes. Il suffit de cliquer sur « Automatic repair » (figure-II.23-).

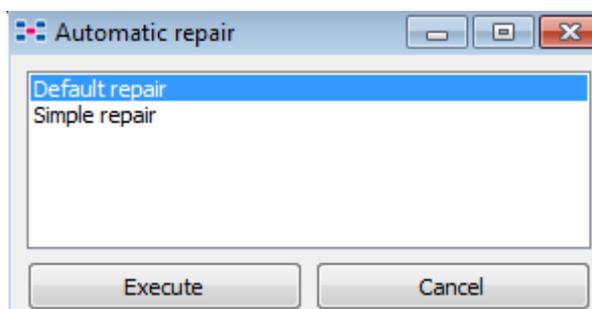


Figure-II.23- Réparation automatique.

Une fenêtre s'ouvre et on clique sur « Execute » Default repair.

Pour valider la réparation, on clique sur « Apply repair » (figure-II.24-) puis on sélectionne « remove old parts » dans la fenêtre qui se lance, cela nous permet de supprimer définitivement les maillages corrompus.

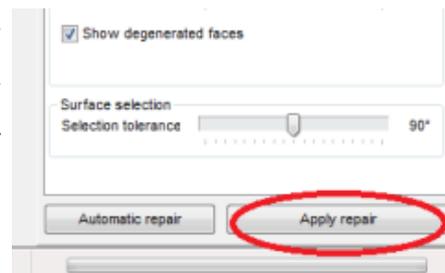


Figure-II.24- Validation de la réparation.

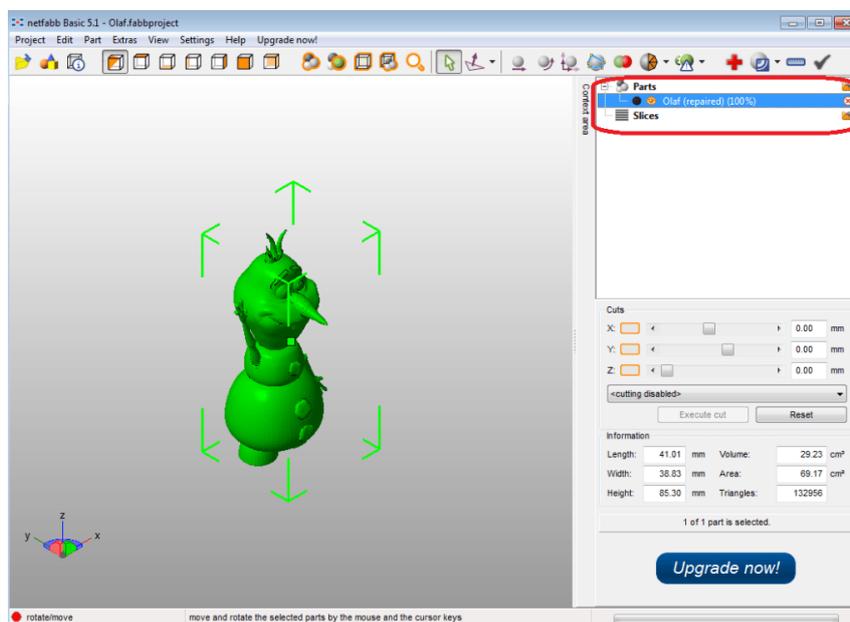


Figure -II.25- Fichier STL réparé.

Et voilà ! Le fichier STL est maintenant réparé à 100%

II.10.3 Inkscape et l'extension G-codetools



Inkscape est un logiciel libre de dessin vectoriel. C'est un outil extrêmement puissant permettant de créer des graphismes au format vectoriel .svg.

Récemment une extension créée par Nick nommée " G-codetools " a été intégrée de façon native à Inkscape. Cette extension permet de convertir des chemins obtenus dans Inkscape en langage G-code, langage permettant de piloter des machines à commandes numériques. Ces machines peuvent être des fraiseuses, des lasers. La figure-II.26- représente la fenêtre principale du logiciel Inkscape.

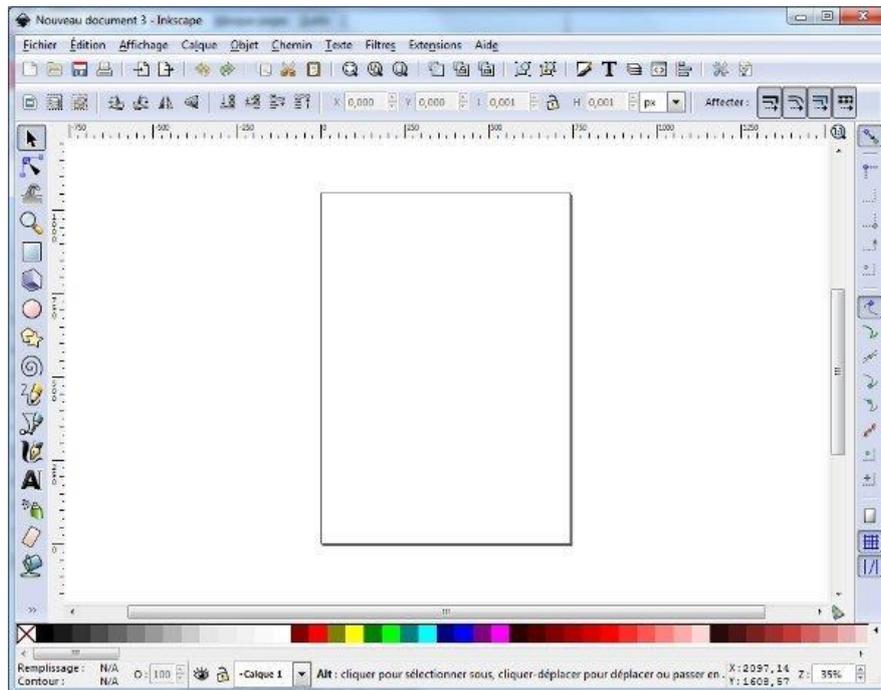


Figure-II.26- Interface inkscape.

II.10.3.1 Utilisation de G-codetools

Une fois que le fichier est importé ou bien la forme à découper est tracée, on va dans le menu " Extensions → Programmation de commande numérique " pour utiliser G-codeTools.

Dans le menu " Extensions → Programmation de commande numérique (G-codeTools) → Points d'orientation ", On place les points d'orientation permettant d'effectuer d'éventuelles transformations (figure-II.27-).

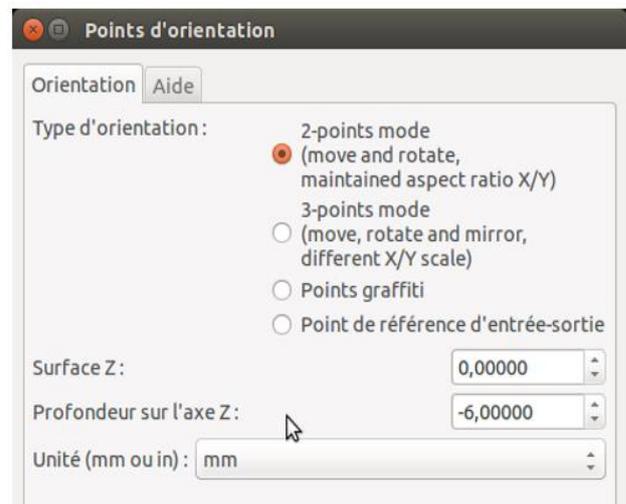


Figure -II.27- Point d'orientation.

Ensuite via le menu " Extensions → Programmation de commande numérique → Bibliothèque d'outils", on choisit l'outil de découpe (figure-II.28-).

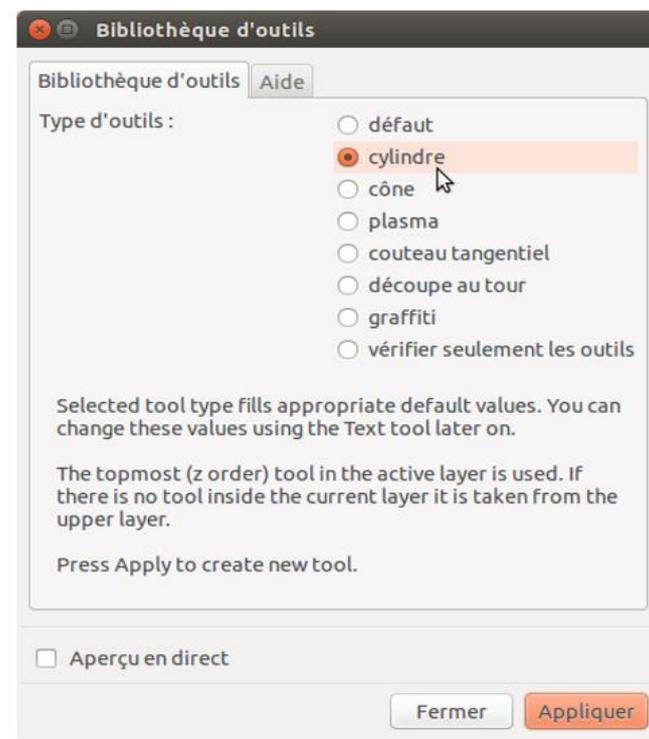


Figure -II.28- Bibliothèque d'outils.

Un cadre vert apparaît alors sur la page (figure-II.29-), il contient différents paramètres modifiables, notamment le diamètre de la fraise et la hauteur de passe de la fraise, en utilisant l'outil "Texte".

name	Default tool
id	default tool
diameter	10.0
feed	400.0
shape	10
penetration angle	90.0
penetration feed	100.0
passing feed	800
depth step	1.0
in trajectory	(None)
out trajectory	(None)
gcode before path	(None)
gcode after path	(None)
sog	(None)
spinlde rpm	(None)
CW or CCW	(None)
tool change gcode	(None)
4th axis meaning	(None)
4th axis offset	0.0
4th axis scale	1.0
fine feed	800

Figure-II.29- Panneau paramètre de modification.

On va maintenant tracer le chemin que devra suivre la fraise. Pour cela, après avoir sélectionné les formes à découper, on va dans le menu "Extensions → Programmation de commande numérique → Chemin vers G-code" (figure-II.30-).

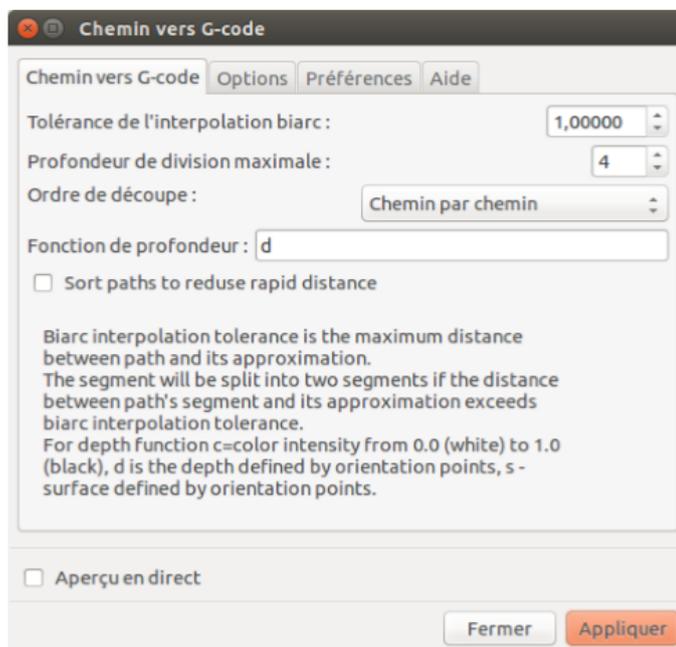


Figure -II.30- Chemin vers G-code.

On trouve plusieurs onglets dont l'onglet " Préférences " permettant d'indiquer à quel emplacement sauvegarder le fichier G-code obtenu (figure-II.31-).

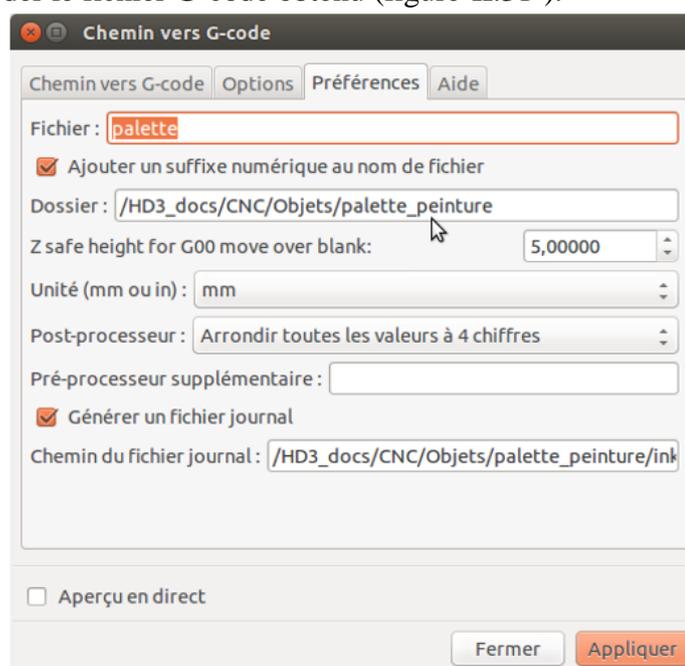


Figure -II.31- Emplacement fichier G-code.

On se place dans le premier onglet " Chemin vers G-code " puis on clique sur le bouton " Appliquer " .

II.10.4 Arduino IDE



Le logiciel Arduino est un environnement de développement (IDE) open source et gratuit, téléchargeable sur le site officiel Arduino.

L'IDE Arduino permet (figure-II.32-) :

- ✓ D'éditer un programme : des croquis (sketch en Anglais),
- ✓ De compiler ce programme dans le langage « machine » de l'Arduino,
- ✓ De téléverser le programme dans la mémoire de l'Arduino,
- ✓ De communiquer avec la carte Arduino grâce au terminal.



Téléverser sur la carte

Vérifier (compiler)

Ouvrir le moniteur série

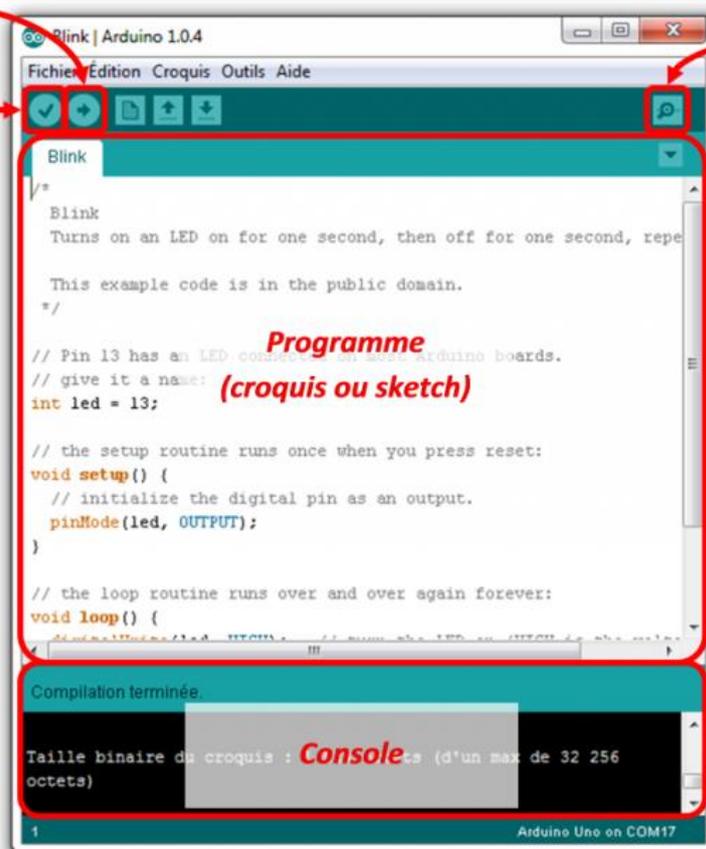


Figure -II.32- Interface de l'ide Arduino.

II.10.5 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 exemple Grbl Contrôleur.

II.10.5.1 Grbl Contrôleur

Le Panneau de Contrôle Grbl intègre de nombreuses fonctions habituellement trouvées sur des machines industrielles. Il “clone” plus ou moins un panneau physique comme on pourrait le trouver sur une machine FANUC ou Haas. Tous les outils nécessaires au réglage et au fraisage sont organisés de manière claire. La figure-II.33- représente l’interface de commande du logiciel Grbl contrôleur.

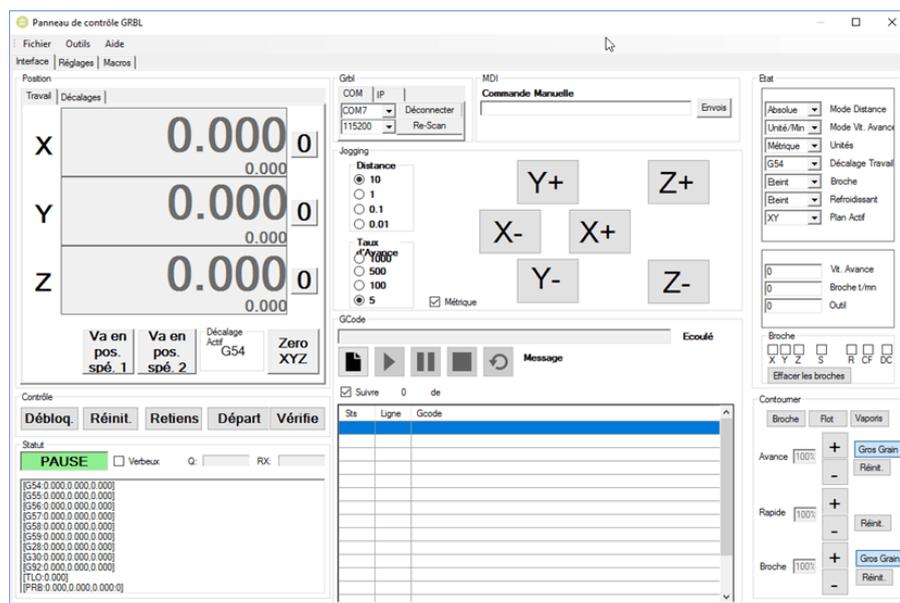


Figure-II.33- Interface de Grbl Contrôleur.

Une opération de fraisage typique requière le déplacement manuel par petits pas au point origine 0 pour la pièce considérée. Souvent cela demande une mise au point sur chacun des axes. C’est pour cette raison qu’il y a un bouton zéro près de chaque axe. Une fois qu’on exécute un fichier G-Code, on ne devrait plus être capable d’envoyer des commandes manuelles. Ici les deux (2) sont liés, une opération bloque l’autre. Lorsque qu’une condition d’alarme est détectée par Grbl, l’exécution du G-Code est stoppée. Les boutons de contrôle et la visualisation est

groupée en sous-panneaux qui sont activés/désactivés en fonction de ce que le Panneau de Contrôle Grbl est en train de faire. Un onglet pour les réglages (figure-II.34-) fourni un certain nombre de zones éditables qui modifient le fonctionnement du panneau, où ce qui est envoyé à Grbl. On peut aussi modifier directement les réglages internes à Grbl depuis ce même onglet.

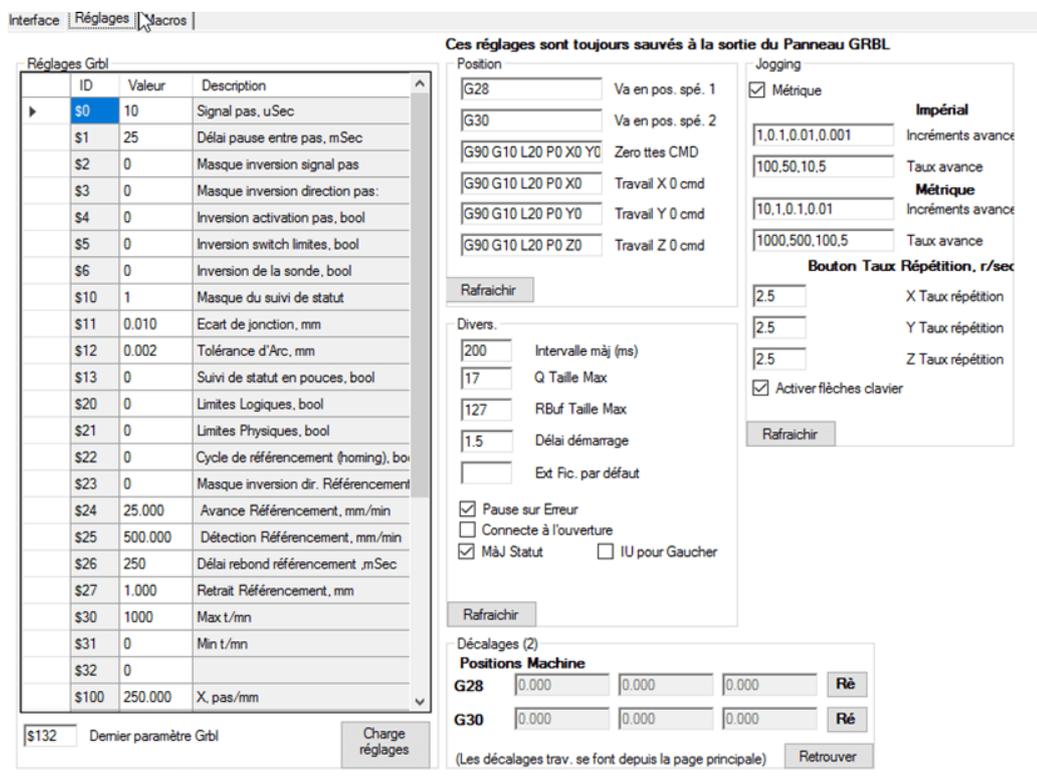


Figure-II.34- Panneau des réglages Grbl.

II.11 CONCLUSION

Dans ce chapitre, nous avons présentés la description générale de la fraiseuse « DENZER-CNC » (composants, caractéristiques et le mode de fonctionnement), ainsi qu’une étude générale du logiciel de commande graphique et du programme de commande de la machine CNC.

On peut, donc, dire que cette partie du travail est un portail pour notre réalisation et conception matérielle.



CHAPITRE III :

Automatisation

et

simulation de la

commande

numérique

III.1 INTRODUCTION

Ce chapitre est consacré à la fabrication et l'application automatique de la machine DENZER-CNC.

Pour cette raison, plusieurs étapes ont été envisagées :

- Simulation sur logiciel Isis Proteus : pour valider la faisabilité et le bon fonctionnement des circuits déjà développées théoriquement.
- Une fois la simulation effectuée et le bon fonctionnement des circuits vérifié, on passe à l'étape de la mise en marche de la machine.
- Faire un test avec la machine DENZER-CNC, usinage d'une pièce.

En outre, la fiche technique de la machine est présentée dans cette partie.

III.2 SIMULATION DE LA COMMANDE NUMERIQUE

La simulation est un outil utilisé par le chercheur, l'ingénieur, le militaire, etc. pour étudier les résultats d'une action sur un élément sans réaliser l'expérience sur l'élément réel.

Lorsque l'outil de simulation utilise un ordinateur on parle de simulation numérique. Les chercheurs, les ingénieurs, les militaires et bien d'autres professionnels se posent souvent la question : quel est le résultat que j'obtiens si j'exerce telle action sur un élément ?

Le moyen le plus simple serait de tenter l'expérience, c'est-à-dire d'exercer l'action souhaitée sur l'élément en cause pour pouvoir observer ou mesurer le résultat. Dans de nombreux cas l'expérience est irréalisable, trop chère ou contraire à l'éthique. On a alors recours à la simulation : rechercher un élément qui réagit d'une manière semblable à celui que l'on veut étudier et qui permettra de déduire les résultats.

Dans notre cas, nous allons utiliser un logiciel très connu des électroniciens « PROTEUS ».

III.2.1 Le logiciel PROTEUS

Proteus est une suite logicielle destinée à l'électronique. Développé par la société Labcenter Electronics, les logiciels inclus dans Proteus permettent la CAO (conception assistée par ordinateur) dans le domaine électronique. Deux logiciels principaux composent cette suite logicielle : ISIS principalement connu pour éditer des schémas électriques, ARES dédié à la création de circuits imprimés.

Pour la simulation de la commande numérique sur les moteurs responsables des mouvements axiaux de la machine DENZER-CNC, nous allons utiliser le logiciel ISIS de PROTEUS.

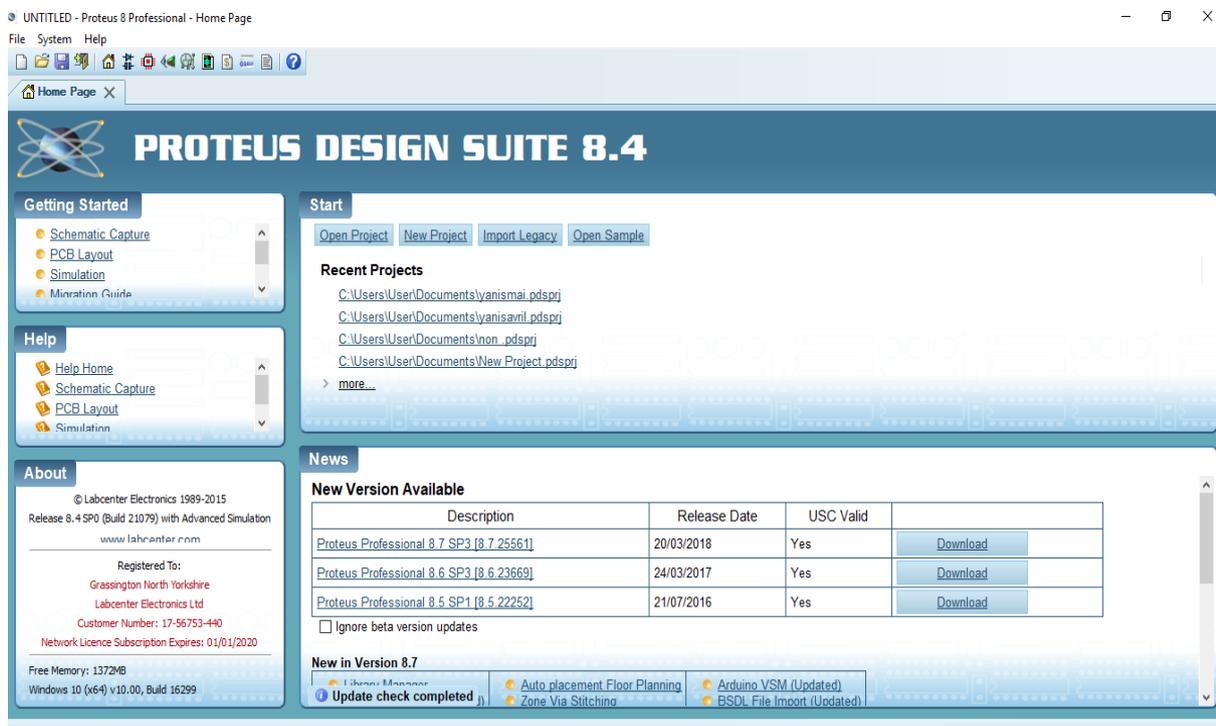


Figure –III.1- Home page de PROTEUS.

III.2.1.1 PROTEUS ISIS

Le logiciel ISIS de PROTEUS est principalement connu pour éditer des schémas électriques et de les simuler de manière très facile.

Cette simulation permet, entre autre, de déceler certaines erreurs de conception et de fabrication.

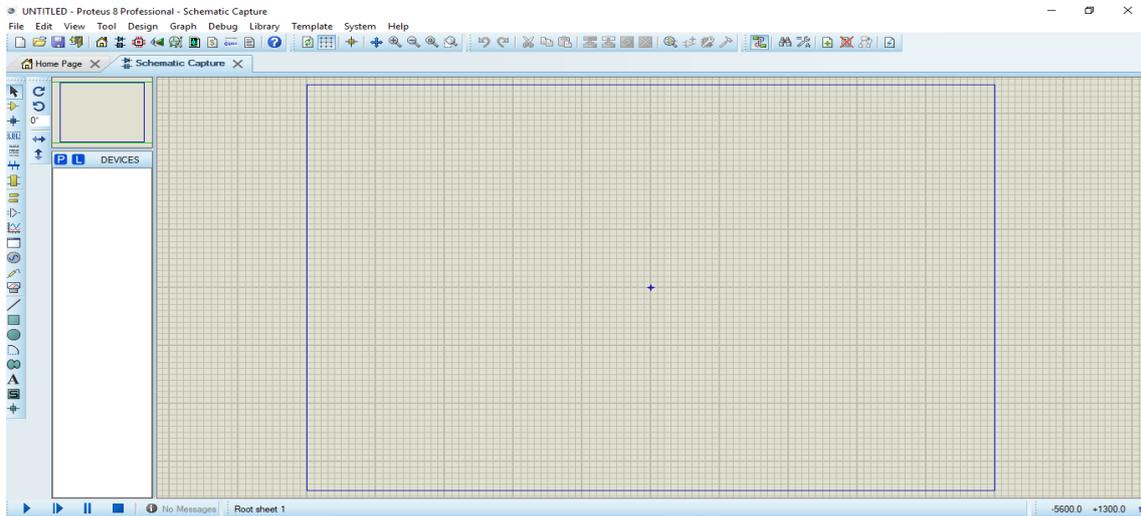


Figure –III.2- Nouvelle page ISIS.

III.2.2 Commande des moteurs pas à pas

Le moteur NEMA 23 (57BYGH627) est un moteur pas à pas hybride, avec aimantation permanente du rotor, bipolaire avec des pas de 1.8 degrés soit 200 pas par tour.

a. Pont en 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.3-) le schéma de principe d'un pont en H :

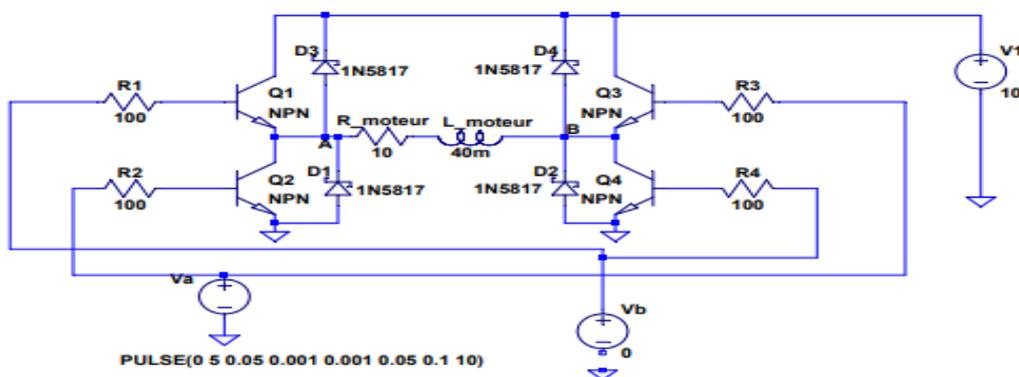


Figure –III.3– Schéma du pont en H.

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).

b. Circuit avec arduino

Le circuit L298N comporte deux ponts en H à transistors bipolaires, pouvant en principe délivrer chacun jusqu'à 2 A, sous réserve que la chaleur dissipée soit bien évacuée. Voici (figure –III.4-) un circuit complet avec les connexions à un Arduino.

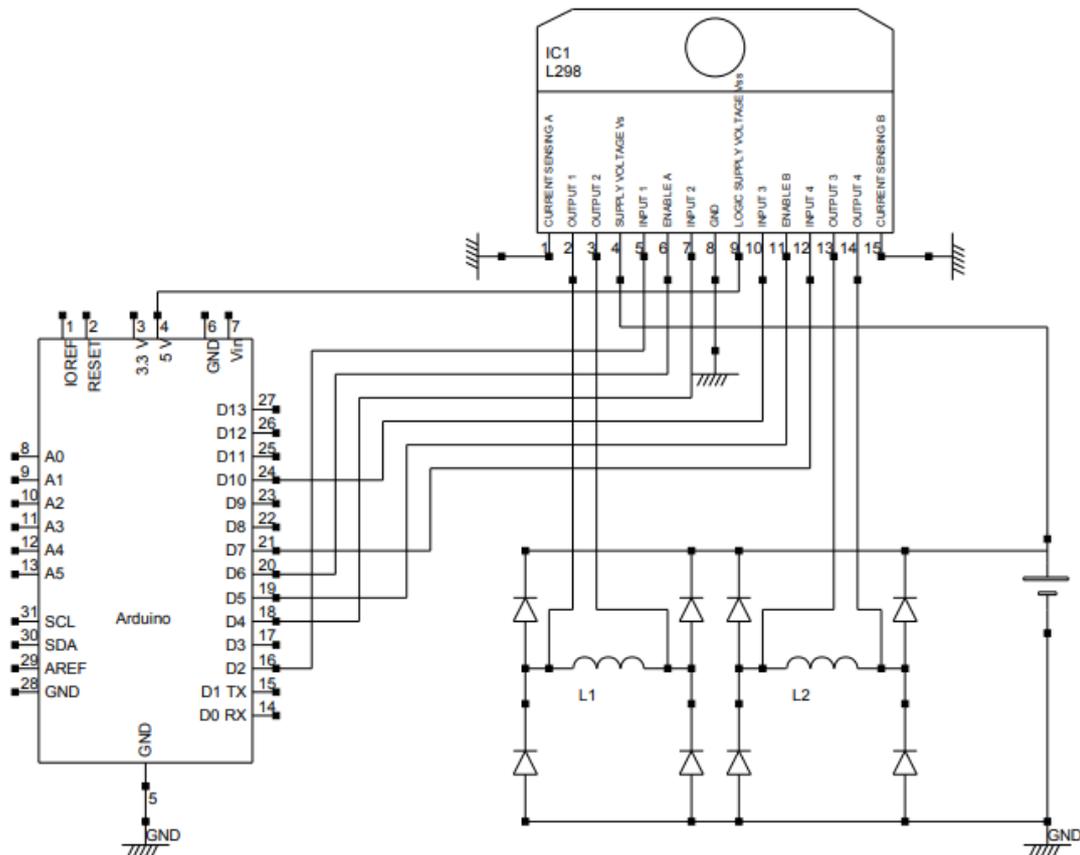


Figure –III.4- Circuit L298 connecté à un Arduino.

Le pont A sert à alimenter la phase A, le pont B à alimenter la phase B. Les deux sorties du pont A sont OUTPUT1 et OUTPUT2. Le pont A est piloté par les trois entrées logiques INPUT1, INPUT2 et ENABLEA. Lorsque ENABLEA=1, un sens du courant est obtenu avec INPUT1=1 et INPUT2=0, alors que le sens opposé est obtenu avec INPUT1=0 et INPUT2=1. Lorsque ENABLEA=0, le pont est inactif (le courant est nul). Voici la table de correspondance entre les sorties de l'Arduino utilisées et les entrées du L298 :

- D2 : INPUT 1.
- D4 : INPUT 2.
- D6 : ENABLEA.
- D10 : INPUT 3.
- D7 : INPUT 4.
- D5 : ENABLE B.

Les entrées ENABLEA et ENABLEB sont pilotées par des sorties PWM de l'Arduino, ce qui permet éventuellement d'appliquer un signal PWM pour limiter le courant moyen, comme on le fait pour les moteurs à courant continu.

c. Signaux de commande

On convient de noter les niveaux logiques dans l'ordre INPUT1, INPUT2, INPUT3, INPUT4. Voici les niveaux à appliquer pour effectuer un pas moteur dans un sens (on change l'ordre pour le sens opposé).

Mode à pas entier, une phase alimentée à la fois (One Phase ON, Full Step) : 1000, 0010, 0100, 0001.

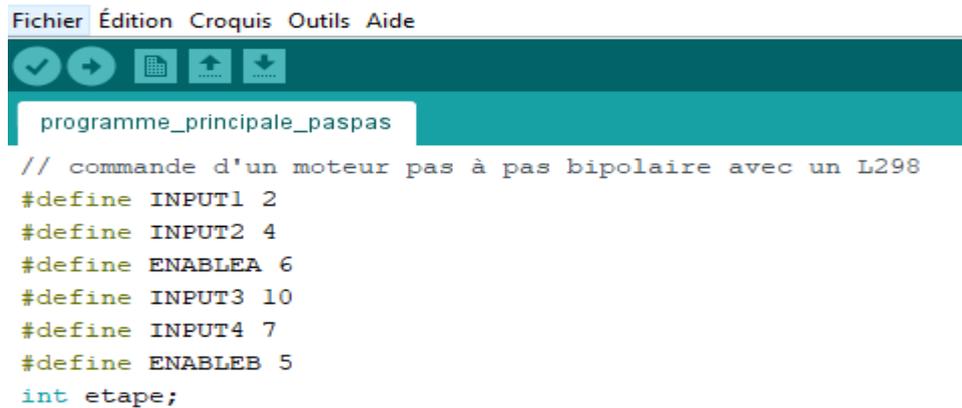
Mode à pas entier, deux phases alimentées en même temps (Two Phase ON, Full Step): 1010, 0110, 0101, 1001.

Mode demi-pas : 1000, 1010, 0010, 0110, 0100, 0101, 0001, 1001. À noter qu'il existe des circuits logiques (comme le L297), qui permettent de générer ces séquences en synchronisation avec un signal d'horloge. Dans le cas présent, on utilisera l'arduino pour générer les séquences, ce qui représente une charge de travail minimale pour celui-ci.

d. Programmation sur Arduino

Le programme suivant montre comment piloter un moteur pas à pas avec le circuit de la figure –III.4-.

On commence par définir les sorties utilisées et la variable globale « etape », qui mémorisera l'étape de la séquence d'impulsion en cours.

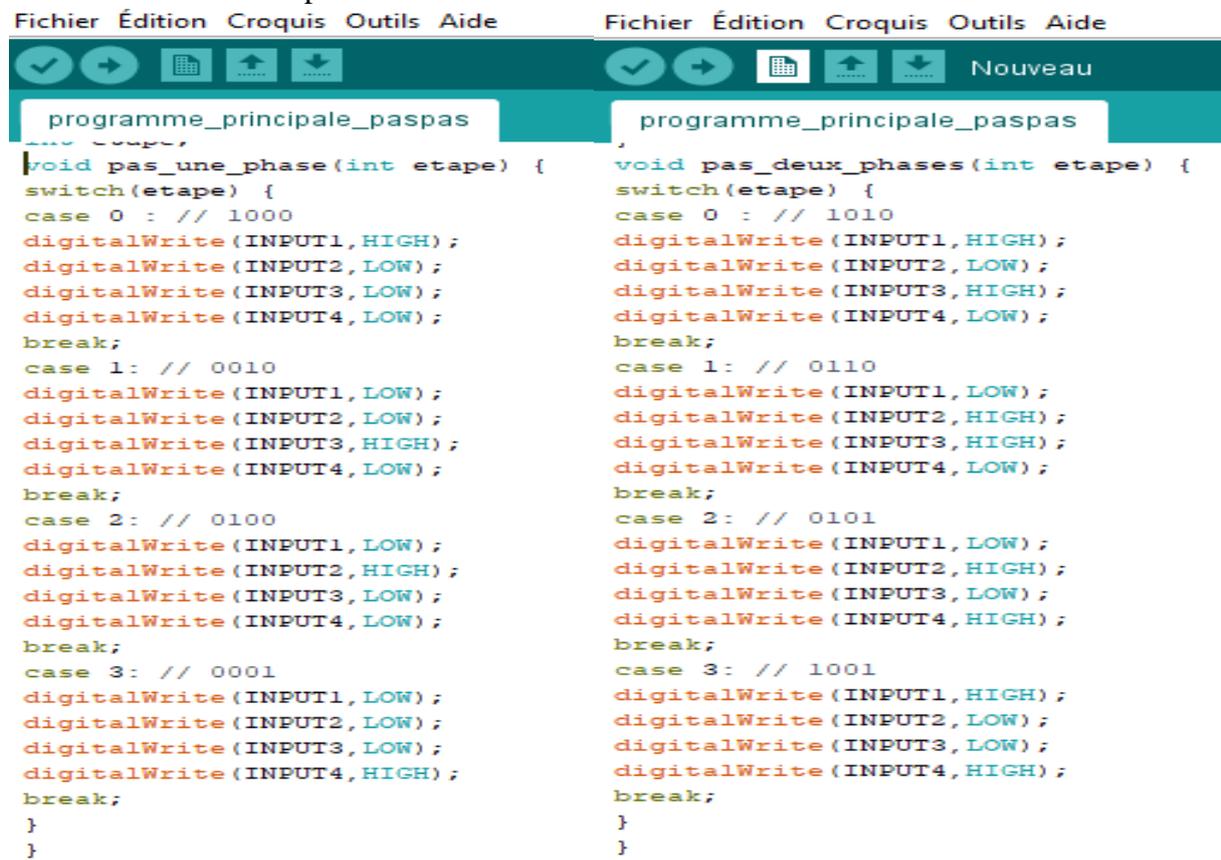


```

Fichier  Édition  Croquis  Outils  Aide
programme_principale_paspas
// commande d'un moteur pas à pas bipolaire avec un L298
#define INPUT1 2
#define INPUT2 4
#define ENABLEA 6
#define INPUT3 10
#define INPUT4 7
#define ENABLEB 5
int etape;
    
```

Figure –III.5- Déclaration des sorties.

Les trois fonctions suivantes appliquent les signaux logiques pour l'une des étapes, pour les trois modes décrits plus haut :



```

Fichier  Édition  Croquis  Outils  Aide
programme_principale_paspas
void pas_une_phase(int etape) {
  switch(etape) {
    case 0 : // 1000
      digitalWrite(INPUT1,HIGH);
      digitalWrite(INPUT2,LOW);
      digitalWrite(INPUT3,LOW);
      digitalWrite(INPUT4,LOW);
      break;
    case 1: // 0010
      digitalWrite(INPUT1,LOW);
      digitalWrite(INPUT2,LOW);
      digitalWrite(INPUT3,HIGH);
      digitalWrite(INPUT4,LOW);
      break;
    case 2: // 0100
      digitalWrite(INPUT1,LOW);
      digitalWrite(INPUT2,HIGH);
      digitalWrite(INPUT3,LOW);
      digitalWrite(INPUT4,LOW);
      break;
    case 3: // 0001
      digitalWrite(INPUT1,LOW);
      digitalWrite(INPUT2,LOW);
      digitalWrite(INPUT3,LOW);
      digitalWrite(INPUT4,HIGH);
      break;
  }
}

Fichier  Édition  Croquis  Outils  Aide
programme_principale_paspas
void pas_deux_phases(int etape) {
  switch(etape) {
    case 0 : // 1010
      digitalWrite(INPUT1,HIGH);
      digitalWrite(INPUT2,LOW);
      digitalWrite(INPUT3,HIGH);
      digitalWrite(INPUT4,LOW);
      break;
    case 1: // 0110
      digitalWrite(INPUT1,LOW);
      digitalWrite(INPUT2,HIGH);
      digitalWrite(INPUT3,HIGH);
      digitalWrite(INPUT4,LOW);
      break;
    case 2: // 0101
      digitalWrite(INPUT1,LOW);
      digitalWrite(INPUT2,HIGH);
      digitalWrite(INPUT3,LOW);
      digitalWrite(INPUT4,HIGH);
      break;
    case 3: // 1001
      digitalWrite(INPUT1,HIGH);
      digitalWrite(INPUT2,LOW);
      digitalWrite(INPUT3,LOW);
      digitalWrite(INPUT4,HIGH);
      break;
  }
}
    
```

```

Fichier Édition Croquis Outils Aide
programme_principale_paspas
void pas_demi_pas(int etape) {
switch(etape) {
case 0 : // 1000
digitalWrite (INPUT1,HIGH);
digitalWrite (INPUT2,LOW);
digitalWrite (INPUT3,LOW);
digitalWrite (INPUT4,LOW);
break;
case 1: // 1010
digitalWrite (INPUT1,HIGH);
digitalWrite (INPUT2,LOW);
digitalWrite (INPUT3,HIGH);
digitalWrite (INPUT4,LOW);
break;
case 2: // 0010
digitalWrite (INPUT1,LOW);
digitalWrite (INPUT2,LOW);
digitalWrite (INPUT3,HIGH);
digitalWrite (INPUT4,LOW);
break;
case 3: // 0110
digitalWrite (INPUT1,LOW);
digitalWrite (INPUT2,HIGH);
digitalWrite (INPUT3,HIGH);
digitalWrite (INPUT4,LOW);
break;
case 4: //0100
digitalWrite (INPUT1,LOW);
digitalWrite (INPUT2,HIGH);
digitalWrite (INPUT3,LOW);
digitalWrite (INPUT4,LOW);
break;
case 5: //0101
digitalWrite (INPUT1,LOW);
digitalWrite (INPUT2,HIGH);
digitalWrite (INPUT3,LOW);
digitalWrite (INPUT4,HIGH);
break;
case 6: //0001
digitalWrite (INPUT1,LOW);
digitalWrite (INPUT2,LOW);
digitalWrite (INPUT3,LOW);
digitalWrite (INPUT4,HIGH);
break;
case 7: //1001
digitalWrite (INPUT1,HIGH);
digitalWrite (INPUT2,LOW);
digitalWrite (INPUT3,LOW);
digitalWrite (INPUT4,HIGH);
break;
}
}

```

Figure –III.6- Les différents modes.

Les fonctions suivantes incrémentent ou décrémentent la variable « etape » selon le sens de mouvement choisi et appliquent les signaux pour l'étape atteinte.

```

Fichier Édition Croquis Outils Aide
programme_principale_paspas
void pas_une_phase_sens_1() {
etape++;
if (etape>3) etape = 0;
pas_une_phase(etape);
}
void pas_une_phase_sens_2() {
etape--;
if (etape<0) etape = 3;
pas_une_phase(etape);
}
void pas_deux_phases_sens_1() {
etape++;
if (etape>3) etape = 0;
pas_deux_phases(etape);
}
void pas_deux_phases_sens_2() {
etape--;
if (etape<0) etape = 3;
pas_deux_phases(etape);
}
void pas_demi_pas_sens_1() {
etape++;
if (etape>7) etape = 0;
pas_demi_pas(etape);
}
void pas_demi_pas_sens_2() {
etape--;
if (etape<0) etape = 7;
pas_demi_pas(etape);
}

```

Figure –III.7- Incrémentation ou décrémentement de la variable « etape ».

La fonction suivante applique un signal PWM sur les entrées ENABLEA et ENABLEB du L298. Dans la plupart des cas, on utilisera soit la valeur 255 soit la valeur 0.



```

Fichier Edition Croquis Outils Aide
programme_principale_paspas $
}
void pwm(int pwm) {
  analogWrite(ENABLEA, pwm);
  analogWrite(ENABLEB, pwm);
}
void setup() {
  pinMode(INPUT1, OUTPUT);
  pinMode(INPUT2, OUTPUT);
  pinMode(INPUT3, OUTPUT);
  pinMode(INPUT4, OUTPUT);
  pinMode(ENABLEA, OUTPUT);
  pinMode(ENABLEB, OUTPUT);
  analogWrite(ENABLEA, 0);
  analogWrite(ENABLEB, 0);
  etape = 0;
}

```

Figure –III.8- Application du signal PWM si les entrées ENABLE.

Voici un exemple avec des blocs de 100 pas suivi d'un temps d'attente de 3 secondes. La durée d'application du courant est choisie, et la durée totale du pas (en millisecondes). Pour faire tourner le moteur à une vitesse bien précise, on calculera la durée du pas en tenant compte du nombre de pas par tour du moteur. Le courant de repos est nul, ce qui suppose que la charge au repos du moteur soit faible.

```

void loop() {
  int k;
  unsigned int duree_courant = 10;
  unsigned long duree_pas = 50;
  unsigned long temps;
  for (k=0; k<100; k++) {
    temps = millis();
    pas_une_phase_sens_1();
    pwm(255); // courant max
    while (millis()-temps < duree_courant) {
      // autre chose à faire pendant le pas moteur
    }
    pwm(0); // courant de repos, en fonction de la charge
    while (millis()-temps < duree_pas) {
      // autre chose à faire pendant le pas moteur
    }
  }
  pwm(0); // courant de repos, en fonction de la charge
  delay(3000);
}

```

Figure –III.9- Exemple.

e. Simulation de l'exemple sur proteus

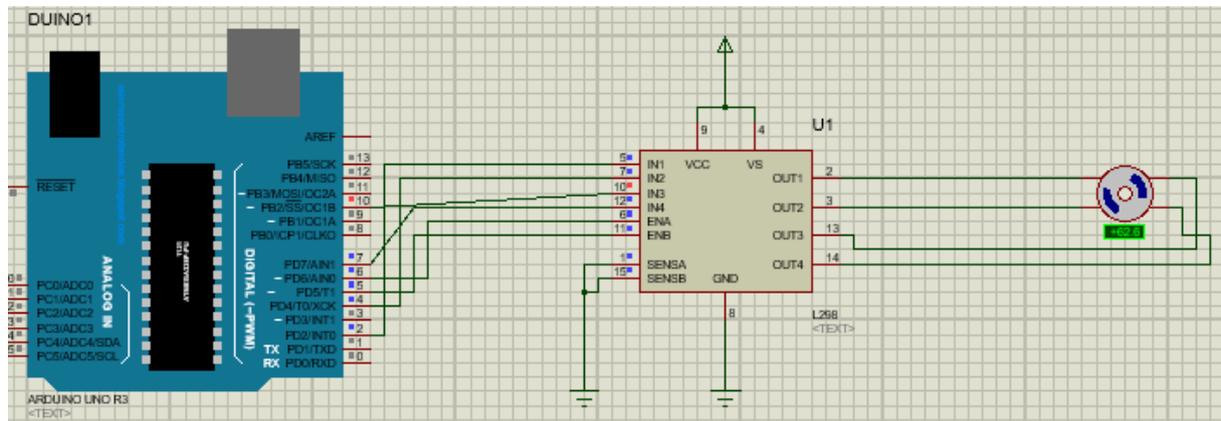


Figure –III.10- Simulation de l'exemple sur proteus.

Plusieurs types de commandes peuvent être envisagés quand il s'agit d'un moteur pas à pas. Parmi ces commandes nous citons :

- ✓ La commande directe avec un amplificateur de la famille L298.
- ✓ La commande dir/step avec le couple L297 et L298.

Les circuits intégrés L297 et L298 sont les composants les plus utilisés lorsqu'il s'agit de piloter des moteurs pas à pas. Ce sont les composants standards par excellence car ils ont été conçu afin d'être utilisé ensemble mais ils peuvent également être inclus dans un quelconque montage.

III.2.3 La simulation avec le couple L297-L298

Pour saisir le schéma, il faut créer un nouveau projet puis placer les composants qui doivent être sélectionné à partir de la bibliothèque des composants.

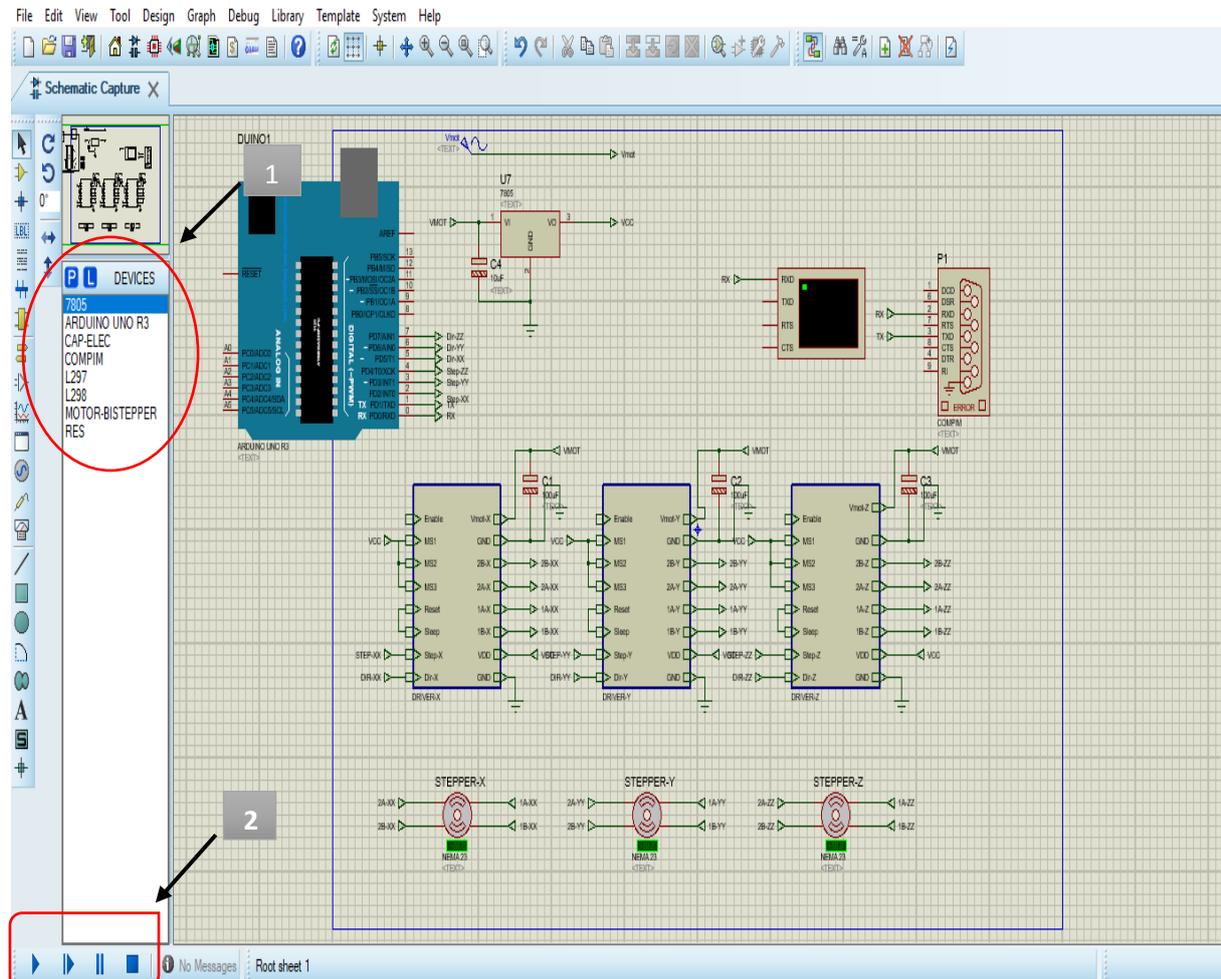


Figure-III.11- Montage électronique dans ISIS PROTEUS.

1. liste des composants.
2. Bouton pour lancer et arrêter la simulation.

III.2.3.1 liste des composants

Les composants électroniques utilisés dans la simulation sont décrits comme suit :

➤ Arduino UNO

Dans cette version de PROTEUS, l'arduino n'est pas compris dans la bibliothèque des composants. Pour cela, il faut, au préalable, télécharger les fichiers ARDUINO.IDX et ARDUINO.LIB et les intégrer dans la bibliothèque (LABRARY) de PROTEUS.

➤ Motor-bistepper

Motor-bistepper est un moteur pas à pas bipolaire librement configurable. Dans notre cas, les moteurs utilisés sont des NEMA 23, nous les avons, donc, configuré comme tel :

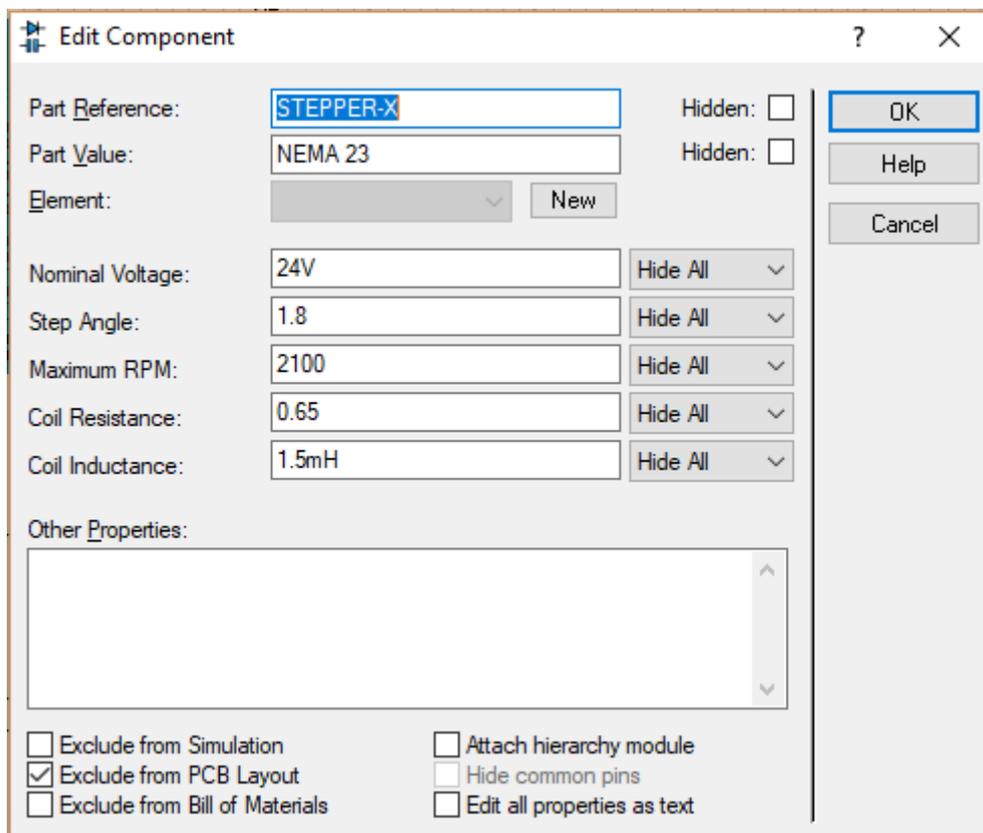


Figure –III.12- Configuration du moteur pas à pas.

➤ **Les drivers moteurs**

Chaque moteur pas à pas est piloté par un driver sous forme de couple (L297-L298).

• **Circuit L298**

Le circuit intégré L298 est le circuit complémentaire du L297. Il permet la simplification extrême de la construction d'une carte de commande pour moteur pas-à-pas. Le circuit L298 est un double pont de commande de puissance, il permet l'utilisation d'une tension élevée. Il peut être utilisé aussi pour la commande de deux moteurs à courant continu. Le schéma interne du L298 contient deux ponts de 4 transistors de puissances commandées par les portes logiques, ainsi que la connexion des résistances de mesure du courant consommée par la charge reliée aux émetteurs de chaque paire de transistors.

• **Circuit L297**

Le circuit L297 est un contrôleur de moteur pas à pas, il fonctionne avec un circuit de puissance à double pont. Il lui suffit de lui fournir les signaux d'horloge (CLOCK : pour avancer des pas), de direction et de mode afin de piloter le moteur pas-à-pas. Le L297 génère alors la séquence de commande de l'étage de puissance.

Le circuit L297 possède deux étages principaux :

- Un translateur (séquenceur) qui génère les différentes séquences de commande.
- Un double hacheur PWM (Pulse With Modulation) qui régule le courant traversant les bobinages du moteur.

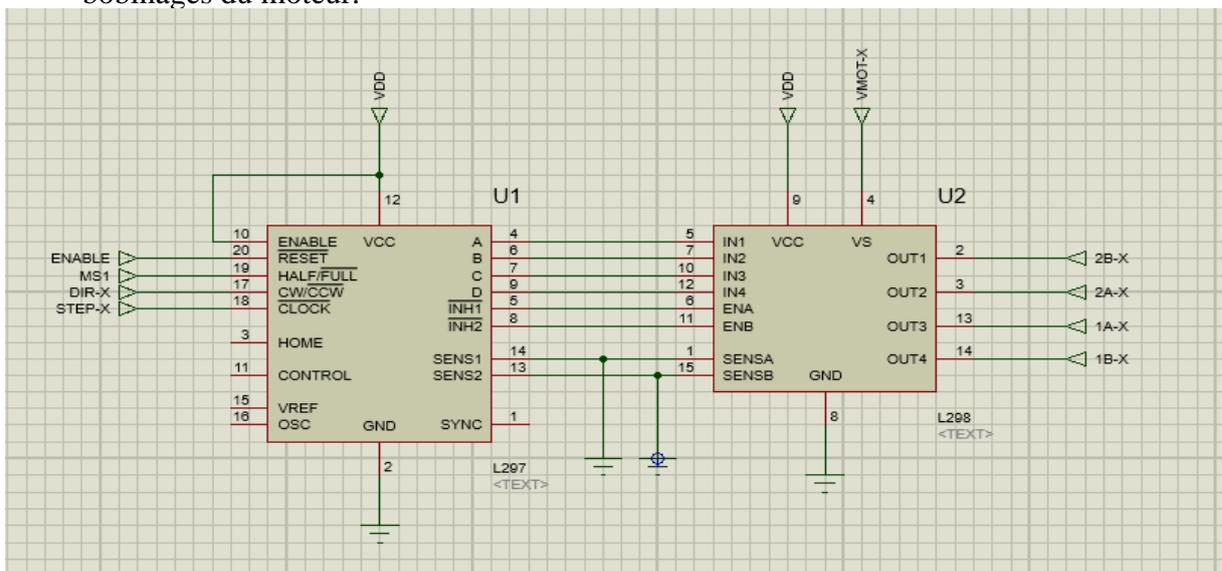


Figure-III.13- Le couple L297-L298.

➤ **7805**

Un régulateur de tension est un circuit intégré. Son rôle est celui d'une source de tension. A savoir qu'il doit fournir une tension constante pour n'importe quel courant de sortie, ou n'importe quelle charge.

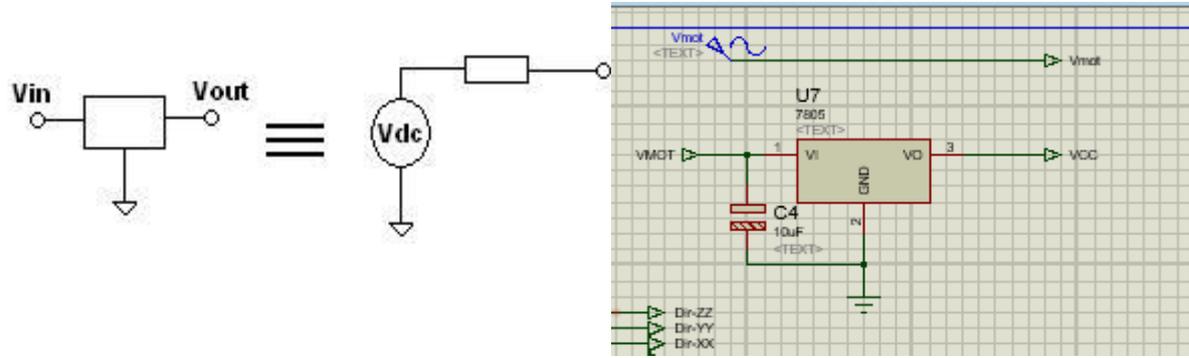


Figure-III.14- Le régulateur de tension 7805.

Ils permettent la conception rapide d'alimentations abordables. Ces composants étant très souvent utilisés dans l'industrie électronique. La tension à sa sortie peut être fixe (cas des 7812, 7805 etc...) ou ajustable (LM317 etc.). Elle peut aussi bien être positive, que négative.

➤ **Compim**

Le port série virtuel COMPIM nous permet de connecter l'arduino à un port virtuel préalablement créé.

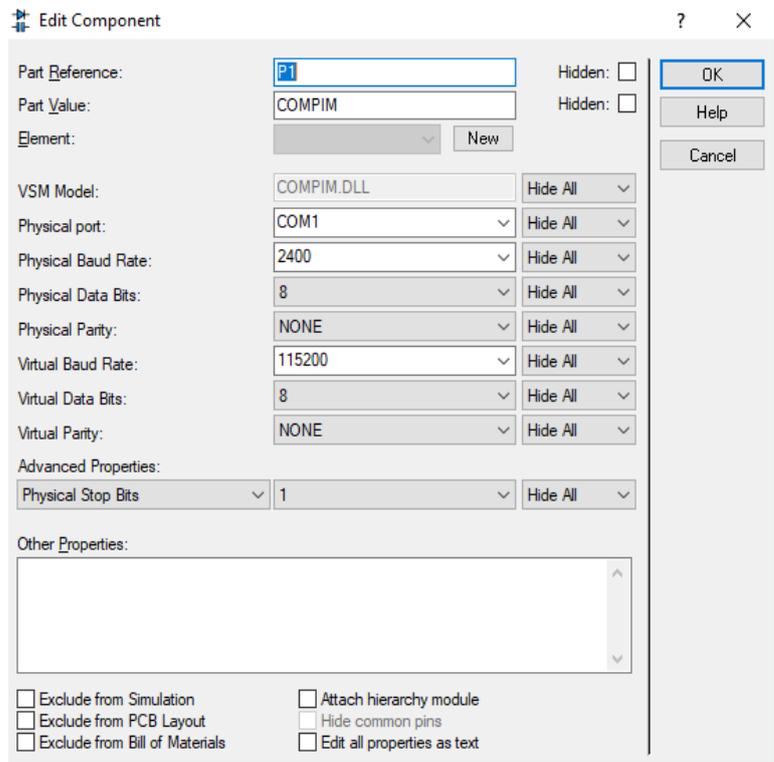


Figure-III.15- Configuration du port série.

III.2.3.2 Le programme de commande

Le programme de commande des moteurs pas à pas « Grbl » est téléversé dans l'arduino et pour cela, le programme « GrblUpload » doit être exécuté dans l'application Arduino IDE.

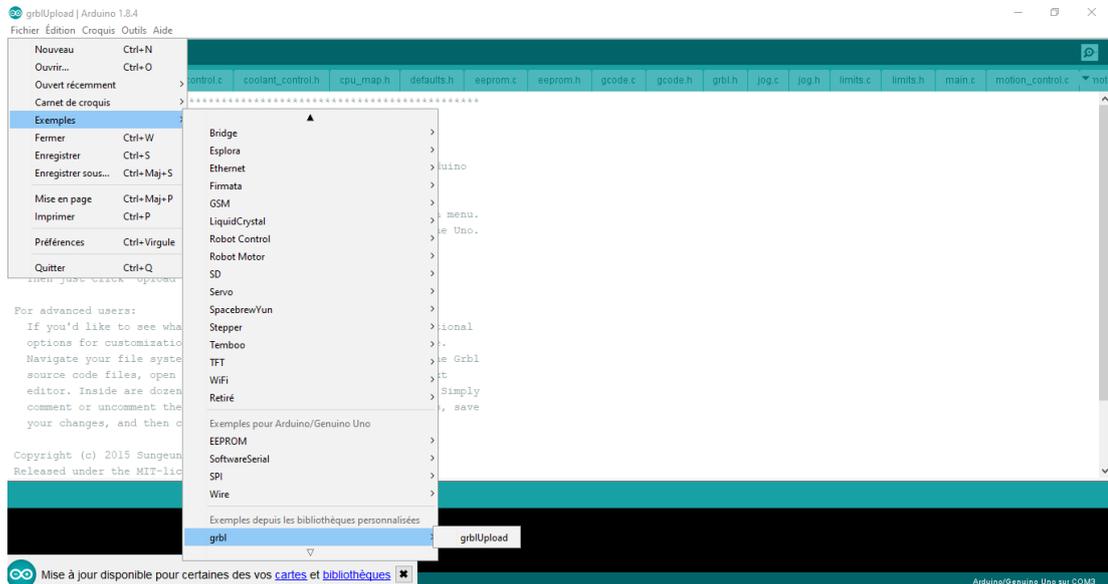


Figure-III.16- Lancement du GrblUpload.

Et repérer l'emplacement de sauvegarde du fichier « grblUpload.ino.elf »

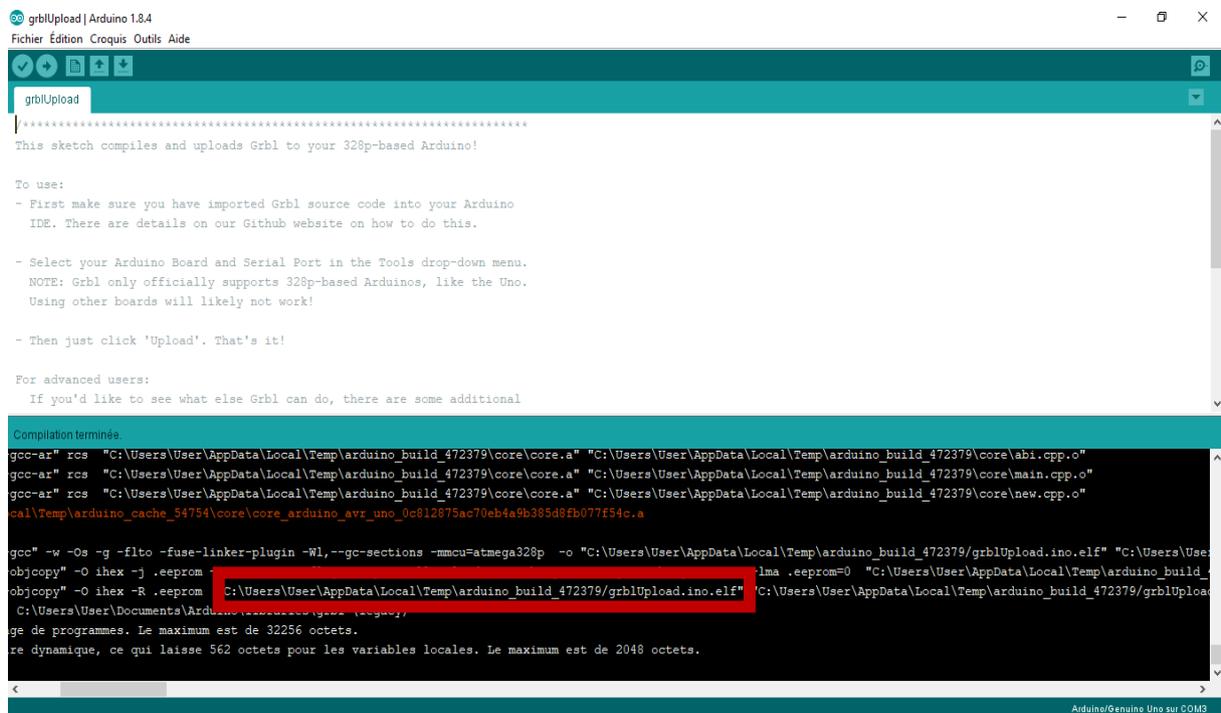


Figure-III.17- Emplacement du fichier « grblUpload.ino.elf ».

Pour associer le programme au processeur (Arduino), il suffit de charger le fichier grblUpload.ino.elf, en suivant son adresse d'enregistrement, dans l'Arduino.

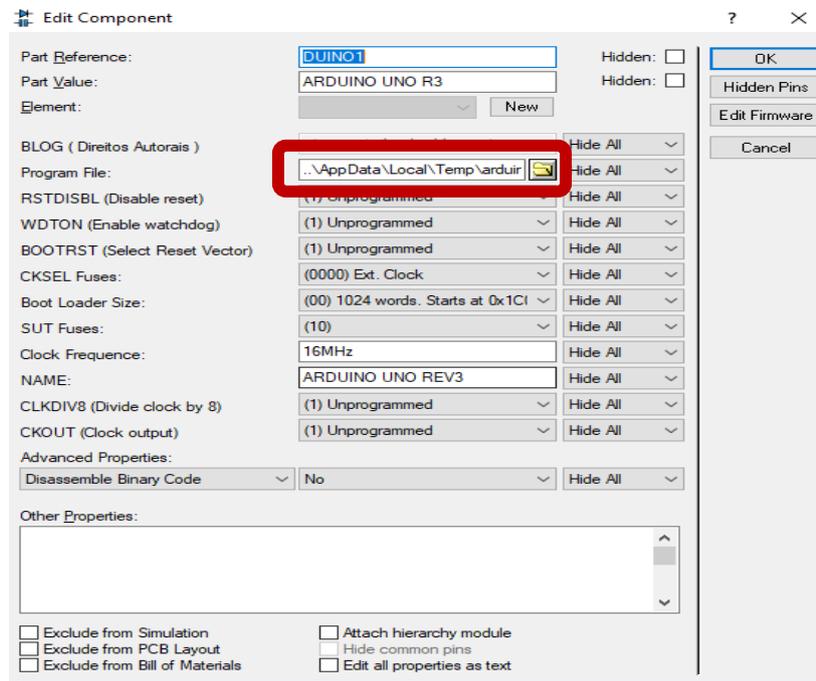


Figure –III.18- Chargement du programme dans l'Arduino.

III.2.3.3 Le logiciel de commande « Universel G-code Sender »

Universel G-code Sender est un lecteur de G-code libre pour piloter les moteurs des machines à commande numérique qui fonctionne en Grbl.

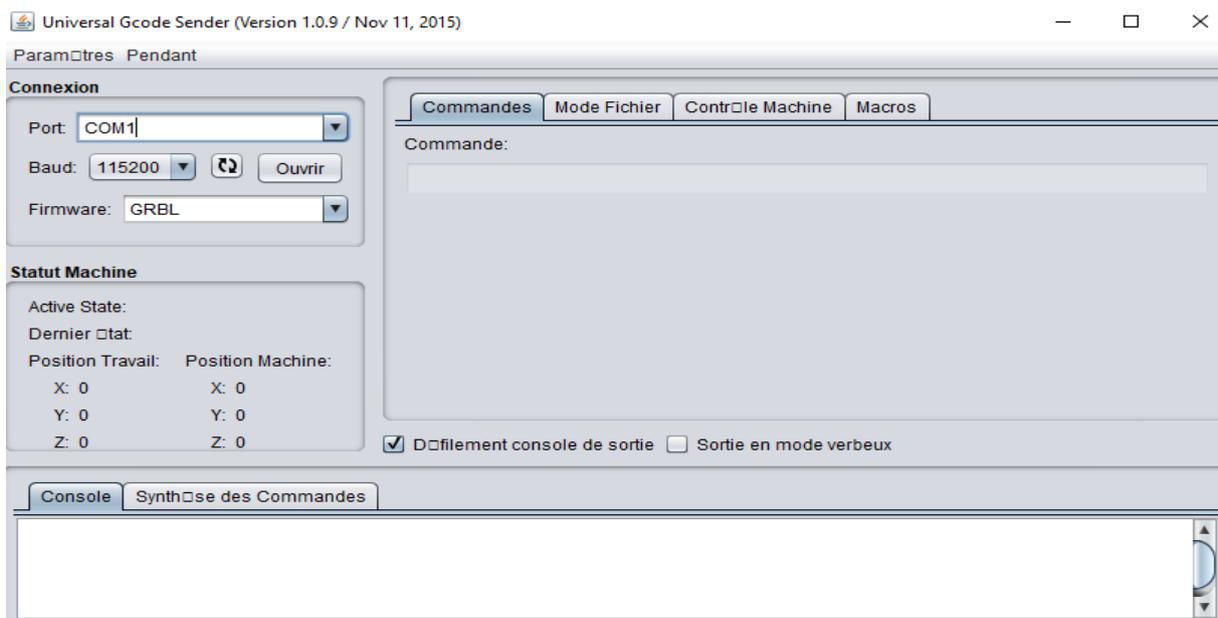


Figure -III.19- Interface de UGS.

UGS propose plusieurs façons de commander la machine, soit en écrivant la commande (G-code), en important directement un fichier G-code ou bien en contrôlant les moteurs

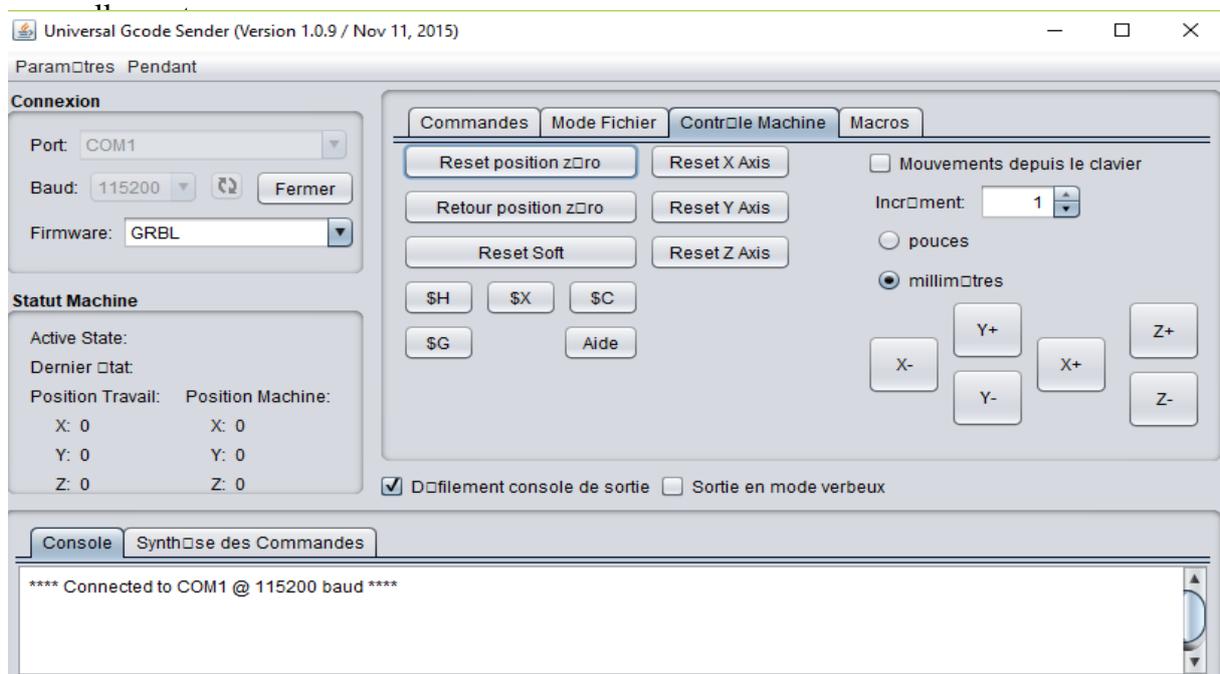


Figure-III.20- commande en mode manuel.

III.2.3.4 Virtual Serial Port Emulateur

VSPE est un émulateur de port série virtuel. Le port série virtuel COM1 est créé pour connecter le logiciel de commande à l'arduino.

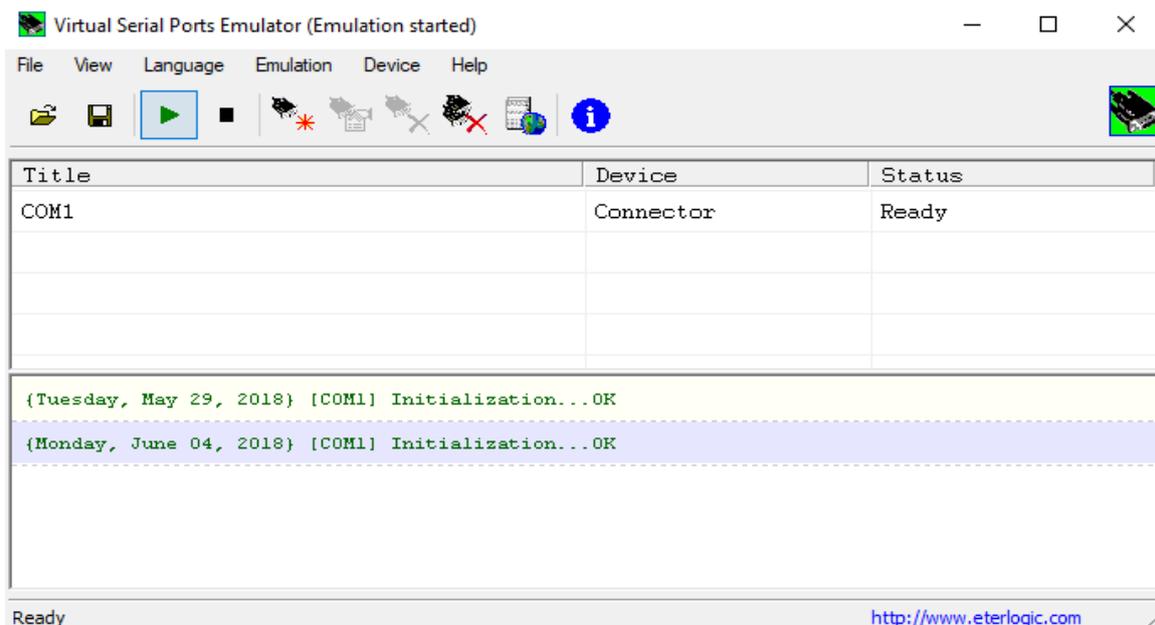


Figure –III.21- Interface de VSPE.

III.2.3.5 Lancement de la simulation

Au lancement de la simulation, on obtient les résultats suivants :

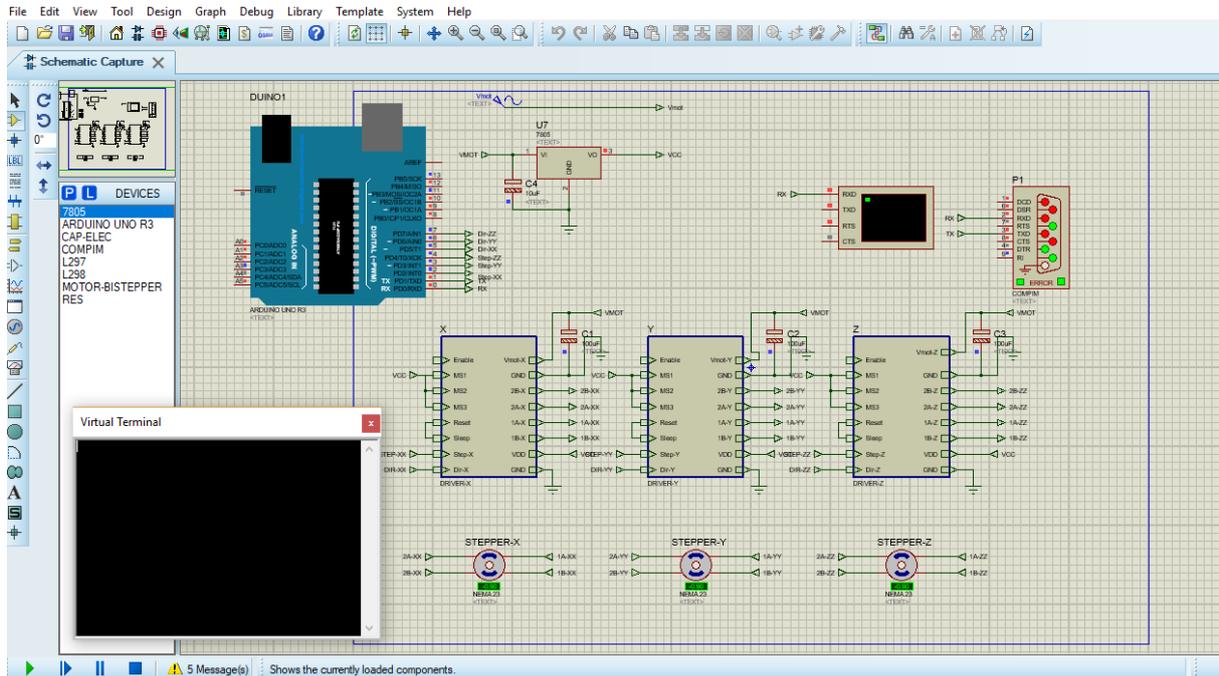


Figure –III.22- Lancement de la simulation.

Après l'importation du fichier g-code, préalablement créé, on appuie sur le bouton 'Envoi' pour lancer la fabrication automatique.

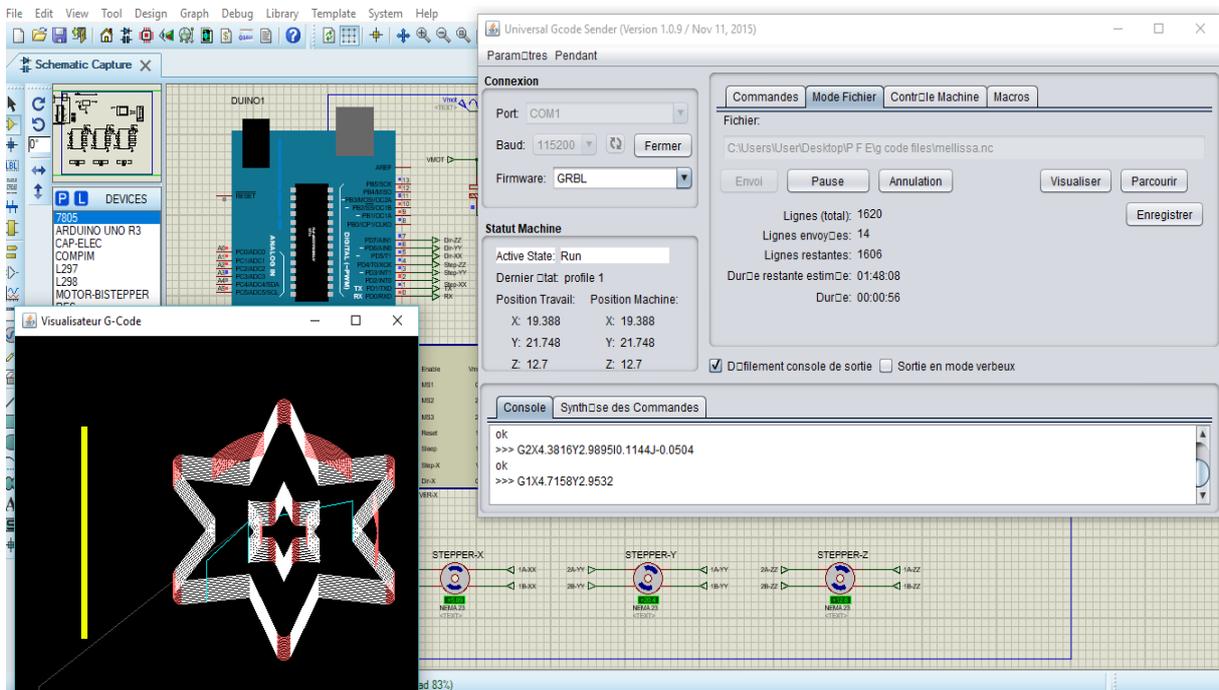


Figure –III.23- commande en mode automatique.

III.3 MISE EN MARCHÉ DE LA MACHINE DENZER-CNC

Après la simulation effectuée et vérifiée, on passe à la mise en marche de la machine DENZER-CNC, qui consiste à usiner une matière brute en une pièce avec des caractéristiques spécifiques.

Pour cela, on doit passer par de nombreuses étapes qu'on résume comme suit :

III.3.1 Conception de la pièce à usiner

Pour la conception de la pièce à usiner, il existe plusieurs logiciels de conception comme SolidWorks, Inkscape, CamWorks et même des sites internet qui proposent une large palette de différentes pièces finies.

III.3.2 Génération du fichier G-code de la pièce à usiner

Pour la création d'un fichier G-code, on peut, soit utiliser un logiciel de conception qui génère directement le fichier G-code de la pièce, tel que les logiciels Inkscape, CorelDraw, Cambam, CharlyGraal et SolidWorks avec l'extension EFICN ou en utilisant une pièce conçue avec l'un des logiciels de conception déjà cités, en important le fichier de la pièce au format STL, DXF ou OBJ.

En outre, il existe des sites internet de conception 3D qui génèrent des fichiers G-code.

III.3.3 Configuration de l'interface de commande numérique

Il y a deux types différents de réglages possible, à la compilation, en modifiant des valeurs dans un fichier, et en interactif, alors que le programme s'exécute.

Le premier cas, c'est la modification de la configuration de base de certains fichiers qui seront compilés avec Grbl lorsqu'il est flashé sur l'Arduino.

Dans l'Arduino IDE, le fichier config.h qui se trouve dans le répertoire GRBL Contient les paramètres de configuration les plus importants, il faut, donc, les modifier ou pas pour adapter au mieux le fichier à la machine spécifique.

```

gblUpload config.h coolant_control.c coolant_control.h cpu_map.h defaults.h eeprom.c eeprom.h gcode.c gcode.h grbl.h jog.c jog.h limits.c limits.h main.c motion_control.c
#ifndef config_h
#define config_h
#include "grbl.h" // For Arduino IDE compatibility.

// Define CPU pin map and default settings.
// NOTE: OEMs can avoid the need to maintain/update the defaults.h and cpu_map.h files and use only
// one configuration file by placing their specific defaults and pin map at the bottom of this file.
// If doing so, simply comment out these two defines and see instructions below.
#define DEFAULTS_GENERIC
#define CPU_MAP_2560_INITIAL

// To use with RAMPS 1.4 Board, comment out the above defines and uncomment the next two defines
//#define DEFAULTS_RAMPS_BOARD
//#define CPU_MAP_2560_RAMPS_BOARD

// Serial baud rate
// #define BAUD_RATE 230400
#define BAUD_RATE 115200

// Define realtime command special characters. These characters are 'picked-off' directly from the
// serial read data stream and are not passed to the grbl line execution parser. Select characters
// that do not and must not exist in the streamed g-code program. ASCII control characters may be
    
```

Figure –III.24- Configuration du fichier config.h.

Dans deuxième cas, le paramétrage se fait alors que l’Arduino est branché et qu’une liaison série avec ce dernier est créer. Il existe un grand nombre d’interface de commande numérique des machines CNC, des logiciels qui permettent d’envoyer le g-code et de contrôler Grbl, que ce soit depuis un PC, un Mac où une machine Linux, Universel G-code Sender et Grbl Contrôleur sont les plus complets.

Et ils ont tous les mêmes paramètres de réglage Grbl représenté dans la figure –III.17-.

```

Grbl 0.9j ['$' for help]
$0=10 (step pulse, usec)
$1=25 (step idle delay, msec)
$2=0 (step port invert mask:00000000)
$3=0 (dir port invert mask:00000000)
$4=0 (step enable invert, bool)
$5=0 (limit pins invert, bool)
$6=0 (probe pin invert, bool)
$10=3 (status report mask:00000011)
$11=0.020 (junction deviation, mm)
$12=0.002 (arc tolerance, mm)
$13=0 (report inches, bool)
$20=0 (soft limits, bool)
$21=0 (hard limits, bool)
$22=0 (homing cycle, bool)
$23=0 (homing dir invert mask:00000000)
$24=25.000 (homing feed, mm/min)
$25=500.000 (homing seek, mm/min)
$26=250 (homing debounce, msec)
$27=1.000 (homing pull-off, mm)
$100=250.000 (x, step/mm)
$101=250.000 (y, step/mm)
$102=250.000 (z, step/mm)
$110=600.000 (x max rate, mm/min)
$111=600.000 (y max rate, mm/min)
$112=600.000 (z max rate, mm/min)
$120=10.000 (x accel, mm/sec^2)
$121=10.000 (y accel, mm/sec^2)
$122=10.000 (z accel, mm/sec^2)
$130=550.000 (x max travel, mm)
$131=350.000 (y max travel, mm)
$132=130.000 (z max travel, mm)
ok
    
```

Figure –III.25- les paramètres de réglage du Grbl.

Après la création du fichier G-code de la pièce à usiner, ce dernier sert à commander la machine automatiquement, pour cela, il suffit de l’importer sur un logiciel d’interface Grbl en liaison direct avec l’Arduino.

III.3.4 Lancement de L'usinage automatique de la pièce

Pour lancer l'usinage (le fraisage) automatique de la pièce :

- ❖ Brancher L'arduino à l'ordinateur via un port USB.
- ❖ Configurer et téléverser le programme GrblUpload dans l'Arduino via port USB.
- ❖ Configurer l'interface de commande Grbl.
- ❖ Initialiser les axes au point zéro manuellement.
- ❖ Importer le fichier G-code de la pièce à usiner dans l'interface de commande Grbl.
- ❖ Mise sous tension de l'armoire de commande et la boîte d'alimentation.
- ❖ Mise sous tension des moteurs de mouvements axiaux.
- ❖ Allumage de la broche.
- ❖ Envoi du fichier G-code et lancement de l'usinage.

III.3.5 Exemple d'usinage d'une pièce

Pour le teste de la machine, on effectue l'exemple suivant :

❖ Conception de la pièce et génération du fichier g-code :

A l'aide du logiciel Inkscape, on conçoit une pièce mécanique et on génère directement un fichier d'usinage grâce à l'extension gcodetools en réglant les paramètres suivants :

- ✓ Surface de Z (surface de l'outil de coupe) :2.00000
- ✓ Profondeur sur l'axe Z :-6.00000
- ✓ Unité : mm (millimètres).
- ✓ Type d'outils : Cylindriques.
- ✓ Ordre de découpe : Chemin par chemin.
- ✓ Hauteur de Z sure pour un déplacement G00 sur vide : 5.00000.
- ✓ Chemin vers g-code (emplacement d'enregistrement) : C:\Users\User\Desktop\PF E\PFE

❖ Configuration et televersement du programme Grbl dans l'arduino :

Après branchement de l'Arduino à l'ordinateur, avec le logiciel Arduino IDE, on doit d'abord configurer le fichier config.h qui se trouve dans le répertoire GRBL et qui contient les paramètres de configuration les plus importants comme :

- Le nombre d'interrupteurs de butée pour un seul axe, dans notre cas c'est un (1), et c'est la configuration par défaut, donc on ne modifie pas. Dans le cas contraire, il suffit de dé-commenter la ligne sept (7) du programme suivant en effaçant les deux (2) slashes (//).

```

1 // If your machine has two limits switches wired in parallel to one axis, you
2 // this feature. Since the two switches are sharing a single pin, there is no
3 // which one is enabled. This option only effects homing, where if a limit is
4 // alarm out and force the user to manually disengage the limit switch. Otherw
5 // limit switch for each axis, don't enable this option. By keeping it disable
6 // homing cycle while on the limit switch and not have to move the machine off
7 // #define LIMITS_TWO_SWITCHES_ON_AXES

```

Figure –III.26- réglage du nombre d'interrupteurs de butée pour un seul axe.

- Définition des états initiaux des fins de courses, normalement ouverts ou normalement fermés, dans notre cas, leurs états initiaux sont normalement ouverts et c'est la configuration par défaut, donc on ne modifie pas.

```

1 // Inverts pin logic of the control command pins. This essentially means when
2 // you can use normally-closed switches, rather than the default normally-open
3 // NOTE: If you require individual control pins inverted, keep this macro disal
4 // the CONTROL_INVERT_MASK definition in cpu_map.h files.
5 // #define INVERT_ALL_CONTROL_PINS // Default disabled. Uncomment to enable.

```

Figure –III.27- Définition des états initiaux des fins de courses.

- Contrôle de la vitesse de la broche, dans notre cas, ce n'est pas nécessaire car la broche n'est pas commandable automatiquement.

```

1 // Enables variable spindle output voltage for different RPM values. On the A
2 // enable pin will output 5V for maximum RPM with 256 intermediate levels and
3 // NOTE: IMPORTANT for Arduino Unos! When enabled, the Z-limit pin D11 and sp
4 // The hardware PWM output on pin D11 is required for variable spindle output
5 #define VARIABLE_SPINDLE // Default enabled. Comment to disable.
6
7 // Used by the variable spindle output only. These parameters set the maximum
8 // "S" g-code values to correspond to the maximum and minimum pin voltages. Th
9 // equally divided voltage bins between the maximum and minimum spindle speeds
10 // max rpm, and 250 min rpm, the spindle output voltage would be set for the
11 // "S1000" @ 5V, "S250" @ 0.02V, and "S625" @ 2.5V (mid-range). The pin output
12 // BG Note: As the spindle max speed is said to be 12K, that's what I set here
13
14 #define SPINDLE_MAX_RPM 18000.0 // Max spindle RPM. This value is equal to 100
15 #define SPINDLE_MIN_RPM 0.0 // Min spindle RPM. This value is equal to (1/256)

```

Figure –III.28- Contrôle de la vitesse de la broche.

Comme la configuration de base du GrblUpload corépond parfaitement à notre machine, on téléverse directement le programme GrblUpload dans l'Arduino via port USB.

❖ Configuration de l'interface de commande numérique :

Dans l'interface de commande numérique Grbl Contrôleur, on règle les paramètres suivants :

- ✓ Nom du port : 7.
- ✓ Vitesse du port : 115200.

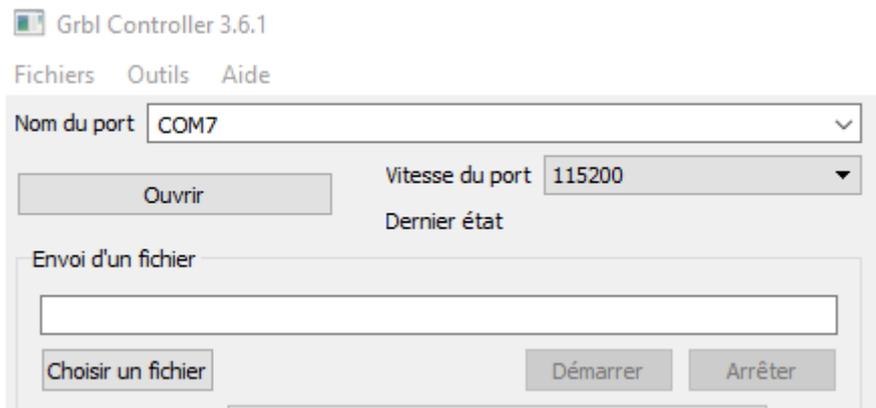


Figure –III.29- Configuration Grbl contrôleur.

On appuie sur « ouvrir », pour pouvoir accéder au paramètres avancé et aux configurations Grbl.

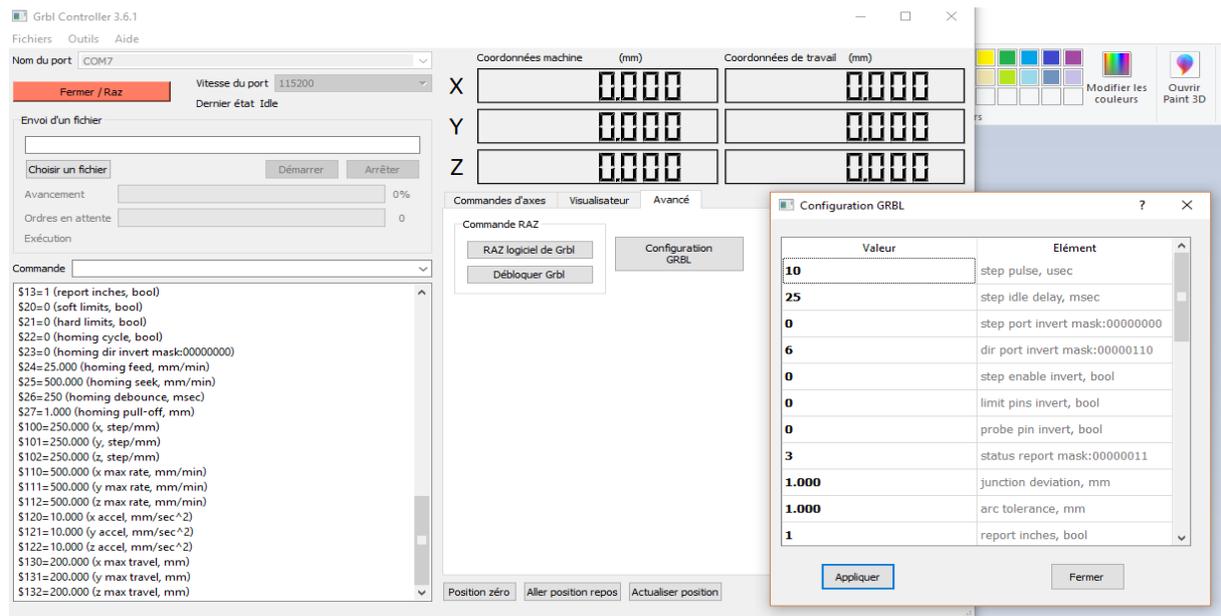


Figure –III.30- Configuration avancé Grbl contrôleur.

❖ Importer le fichier G-code

On importe le fichier g-code créé ultérieurement dans l'interface de commande Grbl.
(figure –III.31-).

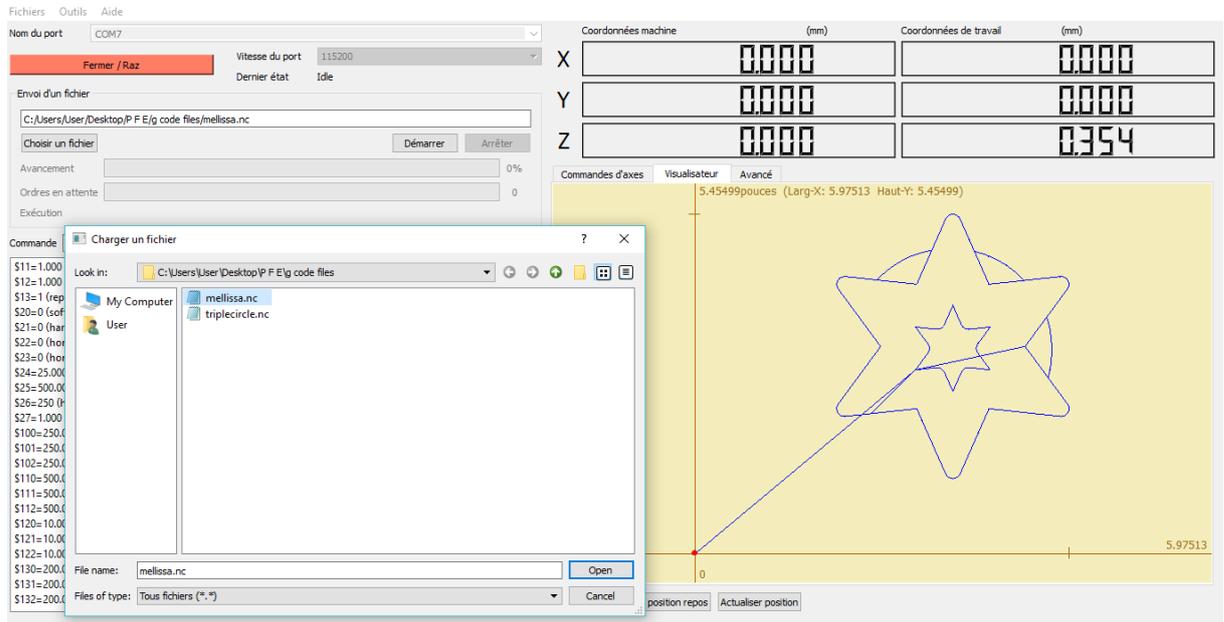


Figure –III.31- Importation du fichier d'usinage.

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

Après branchement de l'armoire de commande à une source de courant/tension, on met sous tension les moteurs pas à pas. (Figure –III.32-).

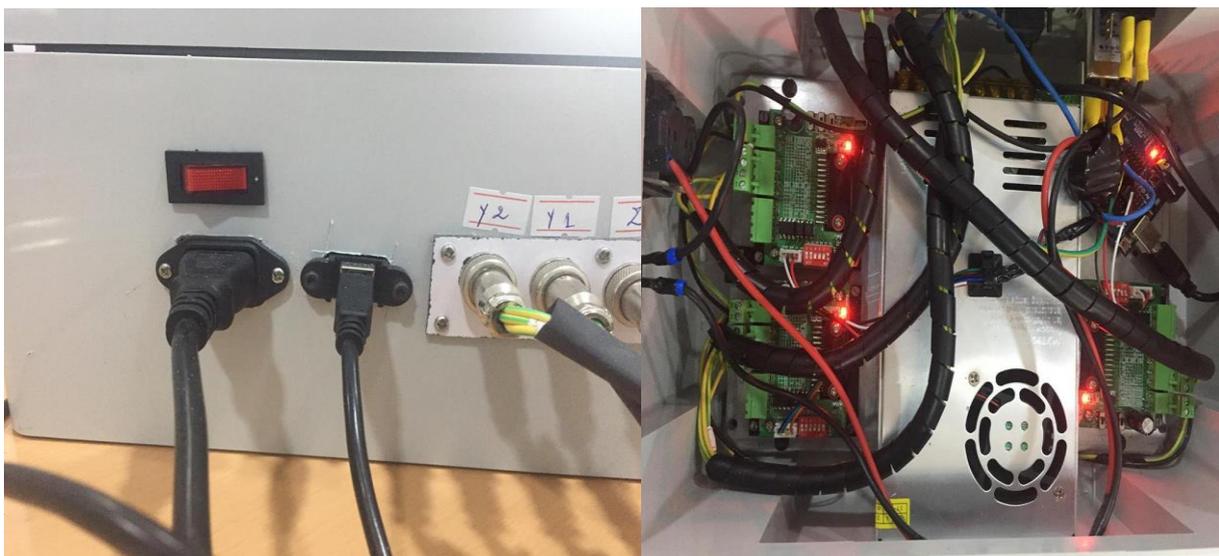


Figure –III.32- La mise sous tension.

❖ **Allumage de la broche et lancement de l'usinage**

Une fois l'armoire de commande sous tension, il ne reste plus qu'à allumer la broche et de lancer l'usinage de la pièce. (Figure –III.33-).

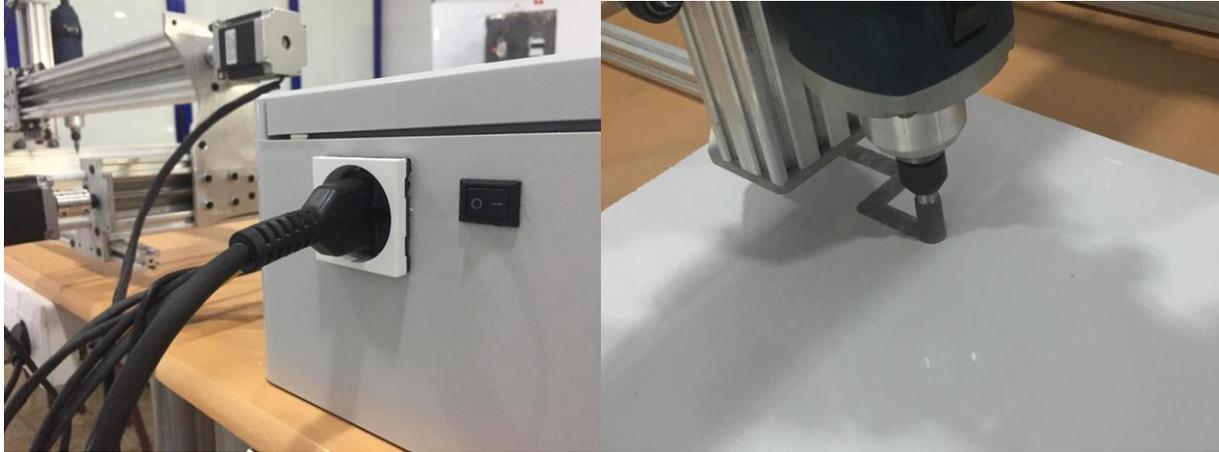


Figure –III.33- lancement de l'usinage.

III.4 FINALISATION DU PROJET (Fiche technique)

MACHINE A COMMANDE NUMÉRIQUE POUR FRAISAGE "DENZER CNC"

Spécifications:

Vitesse de broche :	2800 tr/min.
Puissance débitée par la broche:	350 W.
puissance des moteurs de mouvement:	80 W.
Vitesse maximale des moteurs de mouvement:	
Courses maximales de la broche (X/Y/Z):	
Distance broche table:	
Poids net:	
Tension:	230 V.

Caractéristiques:

- * Broche verticale entraînée par un moteur pas à pas.
- * Aucune carte additionnelle nécessaire dans l'ordinateur.
- * Construction robuste en aluminium pour l'industrie.
- * Fonctionne avec n'importe quel ordinateur du commerce.

Application:

- * Fabrication de plaque de signalisation.
- * Fabrication de maquette d'architecture miniature.
- * Découpe de carte.
- * Prototypage.
- * Fraisage, découpe et gravure.
- * Usinage de poche.
- * Usinage 3D.

Fonctions:

Fraiseuse automatique à commande numérique, entraînée par des vices sans fin avec broche verticale, cette machine permet l'usinage sur trois axes des métaux léger, bois et plastiques, la machine est équipée de quatre moteurs pas à pas et d'une armoire de commande. machine universelle, idéal pour un atelier.

Figure –III.34- Fiche technique de la machine DENZER-CNC.

III.5 CONCLUSION

Dans ce chapitre, on a pu voir que beaucoup de travaux ont été réalisés concernant la machine CNC, nous avons présenté plusieurs types de logiciels utilisés pour la simulation et la commande de machine CNC.

Cette partie du travail présente les différentes étapes de la mise en marche de la machine DENZER-CNC avec un exemple d'application.

CONCLUSION GENERALE

Notre projet de fin d'étude consiste à détailler ainsi qu'à réaliser une carte électronique de commande des moteurs pas à pas responsable des mouvements axiaux de la machine DENZER-CNC et à commander le déplacement de la broche de la machine.

Dans la première partie, nous avons d'abord évoqués une idée générale sur les machines-outils à commande numérique (MOCN) ainsi qu'un aperçu sur la conception et la fabrication assistée par ordinateur (CFAO).

Dans la deuxième partie, nous avons décrit le principe de fonctionnement de la machine DENZER-CNC ainsi que ses principaux composants. Aussi, nous avons vus les différents logiciels agissant sur la commande numérique des moteurs pas à pas de la machine.

Dans la dernière partie, nous présentant les étapes à suivre pour la mise en marche de la machine et l'usinage d'une pièce.

Le but initial de notre projet est la réalisation et l'automatisation d'une machine à commande numérique, plus particulièrement la conception de l'armoire de commande et la génération de la commande numérique des moteurs pas à pas de la machine avec les matériaux et les moyens dont nous avons disposé, ce qu'on pense avoir réalisé. Néanmoins, DENZER-CNC reste une machine en plein développement qu'on pourrait améliorer avec diverse technologie comme :

- Un pupitre de commande avec un lecteur USB ou de carte mémoire pour une manipulation plus facile sans le branchement à une machine de commande externe tel qu'un ordinateur.
- Une broche commandable numériquement avec une plus grande gamme de vitesse.
- Plus de plans au niveau sécuritaire.

L'élaboration de ce travail dans le cadre du projet de fin d'étude, nous a permis d'approfondir nos connaissances théoriques acquises durant notre formation et d'acquérir une

CONCLUSION GENERALE

bonne expérience au niveau de la réalisation pratique. Ce projet nous a donné une meilleure idée sur la complémentarité entre le volet théorique et le volet pratique.

REFERENCES BIBLIOGRAPHIQUES

- [1] A. Fouillé Dunod, la commande des machines-outils, Histoire du Fer Bibliothèque de Travail n° 918
- [2] Gilles Patri, « Aérospatiale : une nouvelle ère industrielle. Des outils classiques à la commande numérique », dans Revue aérospatiale, N° hors-série 20 ans d'Aérospatiale, janvier 1990
- [3] <https://bu.univ-ouargla.dz/master/pdf/Bentaleb-gasmi.pdf?idmemoire=55>
- [4] Gilles prud'homme, « commande numérique des machines-outils » Technique de l'ingénieur, génie mécanique, usinage, B7130, D. Duret, « simulation de gamme d'usinage », revue de l'ingénieur et du Technicien de l'enseignement technique, n°229,1981, pp. 34-37, 1995.
- [5] Gilles Patri, « Aérospatiale : une nouvelle ère industrielle. Des outils classiques à la commande numérique », dans Revue aérospatiale, N° hors-série 20 ans d'Aérospatiale, janvier 1990
- [6] <https://fr.wikipedia.org/wiki/Fraisage>
- [7] <http://newsoof.ru/simulatore-cnc-gratis-in-italiano/>
- [8] <http://www.acheter-vendre-machines.fr/2008/07/2-principe-de-fonctionnement-dun-tour.html>
- [9] http://www.swissmechanic.ch/documents/2609f_production_a_commande_numerique_Tournage.pdf
- [10] [https://fr.wikipedia.org/wiki/Rectification_\(m%C3%A9canique\)](https://fr.wikipedia.org/wiki/Rectification_(m%C3%A9canique))
- [11] <http://www.larousse.fr/archives/grande-encyclopedie/page/11526>
- [12] https://fr.wikipedia.org/wiki/%C3%89lectro-%C3%A9rosion#cite_ref-1
- [13] <http://www.metabricoleur.com/t2002-electro-erodeuse-de-conception-amateur>
- [14] A. Cheikh, N. Cheikh, cours de CFAO1, Université de Tlemcen, Algérie, 2011
- [15] D. Gelin, M. Vincent « éléments des fabrications », Edition marketing, paris, Mars 1995
- [16] <http://www.usimm.ca/quelles-sont-les-differences-entre-lusinage-3-axes-et-5-axes/>
- [17] <http://akou91.over-blog.com/article-construction-d-une-table-de-fraisage-a-commande-numerique-diy-cnc-router-98803054.html>
- [18] <http://sam.electroastro.pagesperso-orange.fr/dossiers/pasapas/moteurpas2.htm>
- [19] Takashi Kenjo and Akira Sugawara, «stepping motors and their microprocessor controls » Oxford University Press, ISBN 0-19-859385-6
- [20] <http://ressources.univ-lemans.fr/AccesLibre/UM/Pedago/physique/02/electri/pasapas.html>
- [21] Pierre Mayé : «Moteurs électriques pour la robotique» ; 2ém édition, octobre 2013

[22] Copyright, Moteur industrie. Tous droits réservés, Moteur industrie 33 rue Paul Gauguin
31100 Toulouse, 27/01/2016

[23] <https://blog.bendinelli.net/index.php?post/2016/02/06transformer-une-d%C3%A9coupeuse-laser-avec-arduino-et-Grbl>

[24] Patrick BLAIN, technique de l'ingénieur (CAO et méthodologie de conception).
Document B 2 810

[25] https://fr.wikipedia.org/wiki/Fabrication_assist%C3%A9e_par_ordinateur