

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



**UNIVERSITE SAAD DAHLAB DE BLIDA 1**  
**FACULTE DE TECHNOLOGIE**  
**DEPARTEMENT DE MECANIQUE**

**SS Laboratory**  
**University of Boumerdes**

Projet de Fin d'Etudes  
Pour l'obtention du Diplôme de Master en  
Génie Mécanique  
Option : Fabrication Mécanique et Productique

# **ÉTUDE ET CONCEPTION DES SYSTÈMES FABRICATION PAR COMMANDE NUMÉRIQUE**

Proposé et encadré par :  
BENMISRA Abdelkader

Réalisé par :  
CHERIET Salah  
MESSAOUDEN Riadh M'hamed

Année Universitaire 2020/2021

# *Remerciement*

Nulle œuvre n'est exaltante que celle réalisée avec le soutien moral  
des personnes qui nous sont proches.

Je tiens à exprimer ma plus profonde reconnaissance à :

DIEU, pour m'avoir donné la force dans les moments difficiles d'éditer ce mémoire.

Mes parents qui m'ont toujours entouré et motivé à sans cesse devenir meilleur ;

Mes frères et sœurs qui m'ont assisté dans ces moments difficiles et m'ont servi  
d'exemple ;

Professeur TEMMAR, responsable de filière et tous le membre de groupe FMP ;

Mon promoteur Mr BENMISRA pour son aide et sa précieuse attention ;

Tous mes professeurs sans exception de la phase primaire jusqu'à ce que je suis  
aujourd'hui pour leurs disponibilité et conseils ;

Mes collègues qui n'ont cessé de m'encourager ;

Trouvez ici l'expression de ma profonde gratitude et reconnaissance.

# *Dédicaces*

*Nous dédions ce modeste travail à :*

*À nos très chères et douces mères, nos très chers pères qui ont été tous jours auprès de nous, merci de nous avoir permis de réaliser notre parcours sans de nejamais manquer de rien.*

*Nous adressons au ciel les vœux les plus ardents pour la conservation de leur santé et de leur vie.*

*À nos frères et nos sœurs, les mots sont faibles pour exprimer la force de nos sentiments et la reconnaissance que nous vous portons. Que Dieu vous garde pournous.*

*Et à ceux qui partagent nos bonheurs et malheurs nos chers amis BAHINHO, KHALED, BRAHIM .*

*À nos collègues au Centre Universitaire SAAD DAHLEB  
BLIDA 1 et nos camarades de promotion 2020-2021*

*Nous voudrions avoir pitié du professeur TEMMAR MUSTAPHA  
et nous lui dédierons cetravail*

*À tous nos enseignants de notre cursus éducatif*

*À tous ceux qui nous sont chers et que nous aimons et qui nous aiment*

## Chapitre 1

Historique

définition de la commande numérique

I. 3. Définition d'une machine CNC

I. 4. Principe de fonctionnement d'une machine à commande numérique:

4.1. Partie commande

4.2. Partie opérative:

I. 5. Domaine d'utilisation

I. 6. les différents types de machines CNC

I. 7. Classification des MOCN

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

I. *7.1.1. Fonctionnement en boucle ouvert*

I. *7.1.2. Fonctionnement avec commande adaptative*

I. *7.1.3. Fonctionnement en boucle fermé*

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

I. *7.2.1. Commande numérique point à point*

I. *7.2.2. Commande numérique par axiale*

I. *7.2.3. Commande numérique de contournage*

I. 7.3. Classification des MOCN selon le nombre d'axe

I. 8. la structure d'une machine CNC

I. 8.1. le système électronique

I. 8.2. le système mécanique

I. 8.3. Le système de programmation

I. 9. Schéma simple d'une machine à commande numérique

I. 10. Conclusion

## Chapitre 2

II. 1. Introduction :

II. 2. Les Actionneurs :

II. 2.1. 1. Généralité sur les moteurs pas à pas :

II. 2.1. 2. Les différents types de moteurs pas à pas :

2.1. 3. Comparaison des 3 types de moteurs :

II. 2.1. 4. Configuration interne des bobines des moteurs Les moteurs à 5 fils :

II. 2.2. Les pilotes des moteurs pas à pas :

1. **Commande des moteurs pas à pas avec A4988**

2. Commande des moteurs pas à pas avec L297 :

## II. 2. 3. Les capteurs :

II. 2. 3. 1. Définition des capteurs :

II. 2. 3. 2. Les différents types de capteurs :

II. 2. 3. 3. Quelques exemples sur les capteurs :

## II. 2. 4. Les microcontrôleurs :

II. 2. 4. 1. Définition :

II. 2. 4. 2. Composants intégrés :

II. 2. 4. 3. Les avantages des microcontrôleurs :

*2. 4. 4. Les défauts des microcontrôleurs :*

II. 2. 4. 5. Familles de microcontrôleurs :

II. 2. 4. 6. En conclusion :

## II. 2. 5. Arduino UNO :

II. 2. 5. 1. Vue d'ensemble :

II. 2. 5. 2. Pourquoi Arduino :

*II. 2. 5. 3. Alimentation :*

*II. 2. 5. 4. La mémoire :*

II. 2. 5. 5. Les entrées et sorties :

II. 2. 5. 6. Communication :

*III. 2. 5. 7. Programmation :*

## II. 3. Commande PWM (Pulse Width Modulation) :

II. 3. 1. Définition :

II. 3. 2. INTÉRÊT :

II. 3. 3. FRÉQUENCE :

I. 3. 4. . RAPPORT CYCLIQUE :

II. 4. Conclusion :

## Chapitre 3

III. 1. Organigramme

III. 2. G-code :

- III. 2. 1. Historique du langage G-code :
- III. 2. 2. Définition de G-code :
- III. 2. 3. Les usages de G-code
  - III. 3. Programmation de la machine à commande numérique
- IV. 3. 1. Structure d'un programme :
- III. 3. 2. Format d'un bloc :
- III. 3. 3. Format d'un mot :
  - III. 3. 4. Les différents formats des mots :
    - III. 4. Le fichier GRBL :
      - III. 4. 1. Schéma de Pin pour Grbl :
      - III. 4. 2. Fonctionnement de grbl :
      - III. 4. 3. Le branchement :
      - III. 4. 4. Installation GRBL dans la carte ARDUINO:
      - III. 4. 5. Universal G-Code Sender :
    - III. 5. InKscape :
    - III. 6. Mise en marche de la machine :
    - III. 7. Utilisation du firmware grbl :
      - III. 7. 1. paramétrages de la machine CNC :
    - III. 7. 2. Initialisation de la machine :
  - III. 8. Conclusion :

## Chapitre 4

IV. 1. Introduction :

IV. 2. Description du projet :

IV. 3. Composants mécaniques de la machine CNC :

IV. 3. 1. Système vis-écrou :

IV. 3. 2. Accouplement :

IV. 3. 3. Rail de guidage V-slot :

IV. 3. 4. Support des axes de guidage :

IV.3. 5. L'axe de guidage :

IV. 3. 6. Les roulements :

IV. 4. Cinématique d'un axe :

IV. 5. Dimensionnement :

IV. 5. 1. Détermination des masses :

IV. 5. 2. Dimensionnement des vis trapézoïdale :

**1) Dimensionnement de la vis trapézoïdale Z :**

**1) Dimensionnement de la vis trapézoïdale Y :**

**2) Dimensionnement de la vis trapézoïdale X :**

IV. 6. La Conception assistée par ordinateur (CAO) :

IV. 6. 1. SolidWorks :

IV. 6. 2. Fonctionnement:

IV. 6. 3. Pièce:

IV. 6. 4. Assemblages:

IV. 6. 5. Mise en plan:

**IV. 7. La conception de notre machine CNC avec SolidWorks :**

A) PARTIE PIECES :

B) PARTIE ASSEMBLAGES :

IV. 8. Conclusion :

## IV. 1. Introduction :

Ce chapitre est consacré à la conception de la machine CNC. Nous exposerons les composants de la machine à réaliser et nous présentons la conception des modèles des pièces et leur assemblage à l'aide du logiciel **SolidWorks**.

## IV. 2. Description du projet :

Notre projet consiste à concevoir un modèle simple à trois axes d'une machine CNC.

Le travail est reparti en trois parties comme suit :

- La partie mécanique
- La partie électronique
- La partie programmation

Dans cette partie nous nous intéressons à la partie mécanique, les deux autres parties ont été détaillées dans les chapitres précédents.

Notre machine ressemblera à une fraiseuse CNC comme celle de la figure IV. 1

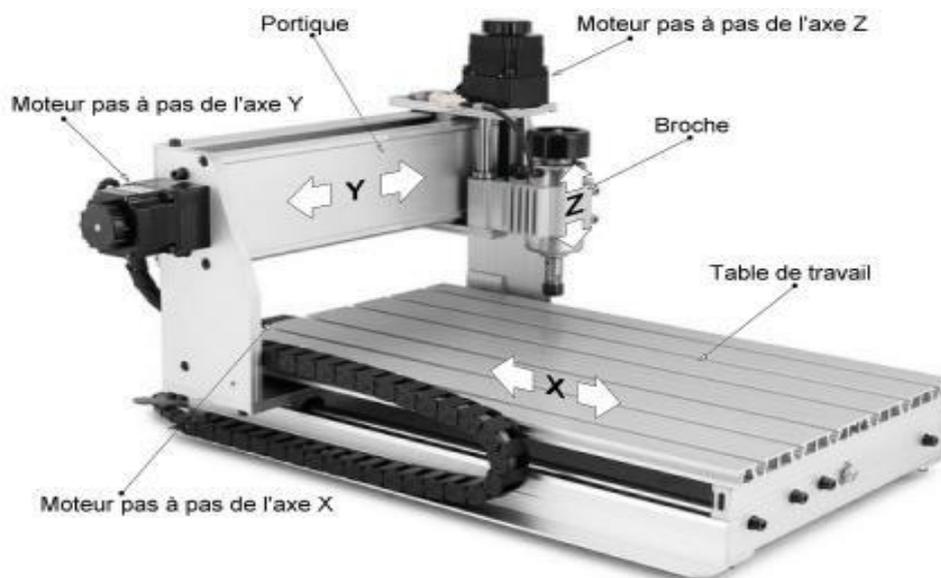


Figure IV. 1. : Modèle d'une fraiseuse.

Le système mécanique est assemblé de telle manière que le mouvement des 3 axes soit réalisé en utilisant des rails linéaires assemblés avec des roulements linéaires. Les moteurs sont montés chacun à l'axe qui est source de mouvement et qui agissent en fonction du signal de commande à

partir du circuit électronique. Chaque moteur est relié à une vis sans fin pour chaque axe qui est chargé de transformer le mouvement du moteur de rotation en mouvement linéaire. Le mouvement contrôlé de chaque axe est obtenu directement par la commande de la rotation du moteur. La vitesse du mouvement de chaque axe peut également être contrôlée par le contrôle direct de la vitesse du moteur, en donnant des signaux de commande nécessaires. Ainsi, la trajectoire de l'outil fixée à l'organe terminal est contrôlée dans chaque axe pour une action bien déterminée.

### IV. 3. Composants mécaniques de la machine CNC :

Dans cette partie, nous passons en revue les différents éléments fonctionnels de notre appareil qui assurent sa bonne performance.

#### IV. 3. 1. Système vis-écrou :

Les trois axes X, Y et Z, permettant le déplacement de l'outil dans les six directions, sont entraînés grâce à la rotation de vis tournant dans l'un ou l'autre sens, selon la direction désirée. La vis sans-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 assure des déplacements de grandes précision.

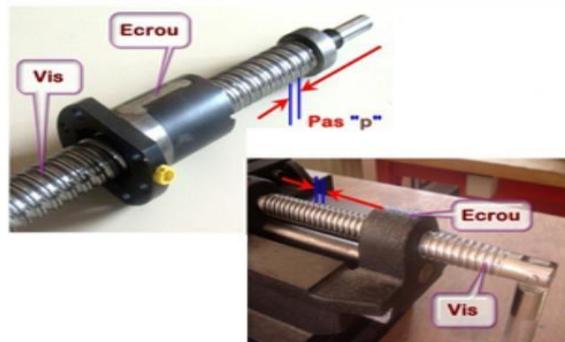


Figure IV. 2. : Vis et écrou à billes.

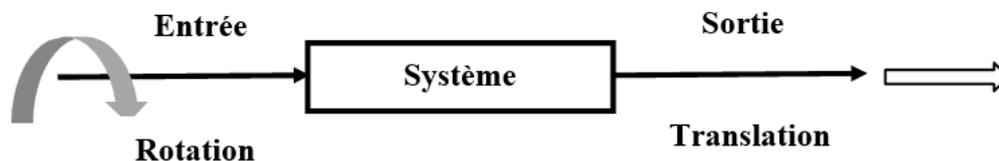
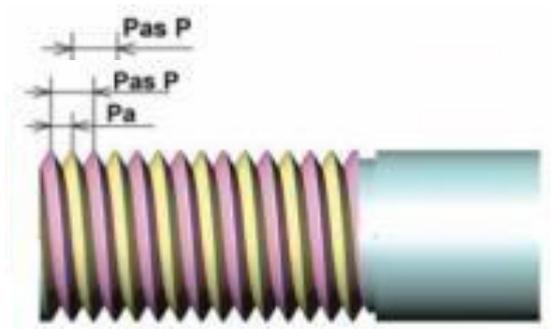


Figure IV. 3. : Transformation du mouvement.

La cinématique du système vis-écrou est définie par le pas de la vis  $P$  en millimètre (mm) qui est la distance parcourue le long de l'axe de l'hélice pour un tour.



a. Vis à un filet  
b. Vis à plusieurs filets

**Figure IV. 4.** : Types de vis à écrou

- (1) Pour une vis a un seul filet (**Figure IV. 4. a**) :  $P = \text{Pas de l'hélice} = \text{Pas du filetage}$   
 (2) Pour une vis a plusieurs filets (**Figure IV. 4. b**) :  $P_a$  : **Pas apparent** et le pas de l'hélice est:

$$P = nP_a \text{ (n est le nombre des filets)}$$

Un tour correspond à un déplacement de 1 pas de l'hélice. Le déplacement est exprimé par la relation :  $L = N_v \cdot n \cdot P_a$  Où :  $N_v$  : étant le nombre de tours effectuée.

Dans notre projet, nous avons utilisé une tige avec 4 filets et un pas de 2mm. La distance parcourue le long de l'axe de l'hélice pour un tour est :

$$L = 1 \times 4 \times 2 = 8 \text{ mm}$$

Sachant que le moteur pas à pas utilisé fait 200 pas par rotation, la distance pour un seul pas est égale à la distance parcourue le long de l'axe de l'hélice pour un tour divisé par 200 :

$$L_{\text{moteur}} = \frac{L}{200} = \frac{8}{200} = 0,04 \text{ mm/pas}$$

#### IV. 3. 2. Accouplement :

L'accouplement est un dispositif de liaison entre deux arbres rotatifs, permettant la transmission du couple. Il existe des accouplements dits « semi-élastiques » qui permettent de rattraper de petits défauts d'alignement (typiquement les défauts d'usinage). Ces accouplements sont généralement constitués de deux parties rigides solidaires des arbres et d'une partie légèrement flexible qui rattrape les défauts d'alignement.



Figure IV. 5. : Accouplement.

Nous avons utilisé 3 accouplements pour accoupler les arbres de moteurs pas à pas (axes X,Y, Z) et les vis à écrous.

### IV. 3. 3. Rail de guidage V-slot :

C'est un profil de rail linéaire de haute qualité en aluminium avec une rainure en V. extrêmement lisse sur les 4 côtés. Il est précis, facile à travailler avec et permet un contrôle illimité de la conception par sa nature modulaire. 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 parallèles tandis que l'axe X et Z contiennent un seul rail chaque un.



Figure IV. 6. : Rail de guidage V-slot.

#### IV. 3. 4. Support des axes de guidage :

Se relie aux rails pour fixer les axes :



Figure IV. 7. : Support des axes de guidage.

#### IV.3. 5. L'axe de guidage :

Permet de guider le mouvement de translation des roulements selon les deux axes de la machine. Les deux axe X et Y est constitué de deux rails parallèles.

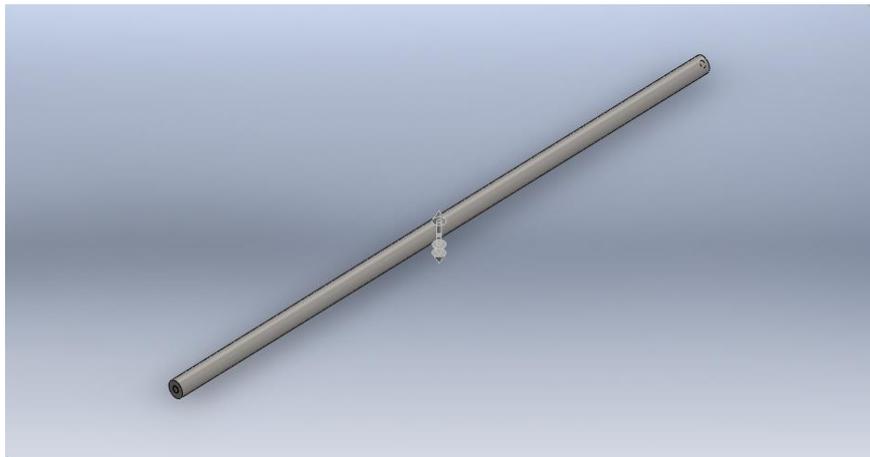


Figure IV. 8. : Axe de guidage.

#### IV. 3. 6. Les roulements :

Les roulements sont des organes mécaniques servant à guider un arbre de transmission en rotation (l'axe X et Y). Ils assurent principalement :

- Le positionnement de l'arbre par rapport à l'alésage.
- Une rotation précise avec un minimum de frottement.



Figure IV. 8. : Roulement

#### IV. 4. Cinématique d'un axe :

Chaque table présente quatre liaisons mécaniques qui sont :

- Liaison encastrement qui représente l'accouplement du moteur.
- Deux liaisons pivots qui assurent la rotation de la vis dans les paliers à roulements.
- Liaison hélicoïdale qui assure la rotation et la translation du système vis à billes.
- Quatre liaisons glissières qui assurent la translation des charriots de guidage.

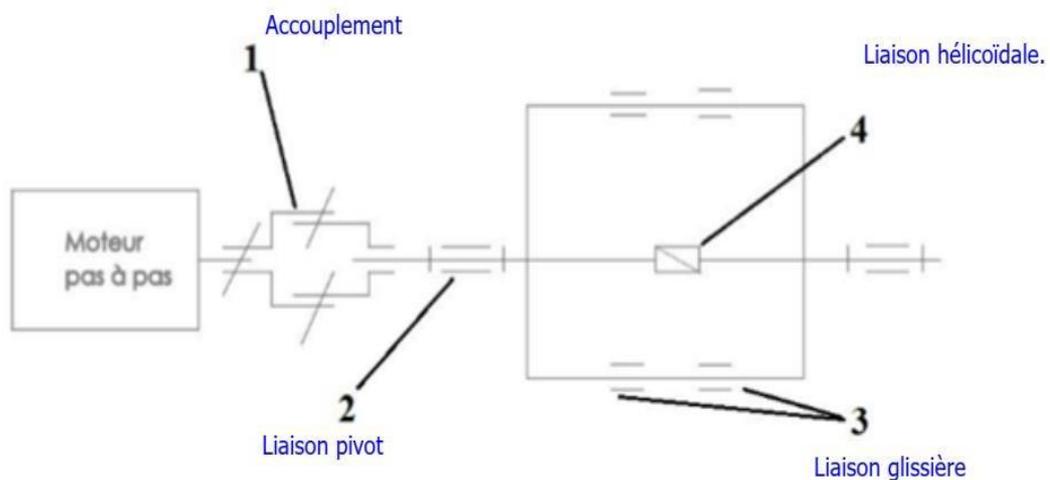


Figure IV. 9. : Dessin représentant les liaisons sur l'axe.

#### IV. 5. Dimensionnement :

Les éléments fonctionnels de la mini fraiseuse à dimensionner sont les suivant:

- Les vis trapézoïdales,
- Les roulements,
- Les douilles à billes,
- Les axes de guidages,
- Accouplements semi-élastiques.

#### IV. 5. 1. Détermination des masses :

Avant de faire l'étude, on doit d'abord calculer la masse de chaque élément de notre machine (figure IV. 10.). Nous avons utilisé deux types de pièces, normalisées et des pièces qui seront usinées. Les masses des pièces normalisées sont tirées d'après les catalogues des fournisseurs. Les masses des pièces à usinées sont calculées approximativement par le logiciel de conception Solid Works 2012. Le tableau III.1 représente les masses des pièces de notre dispositif :

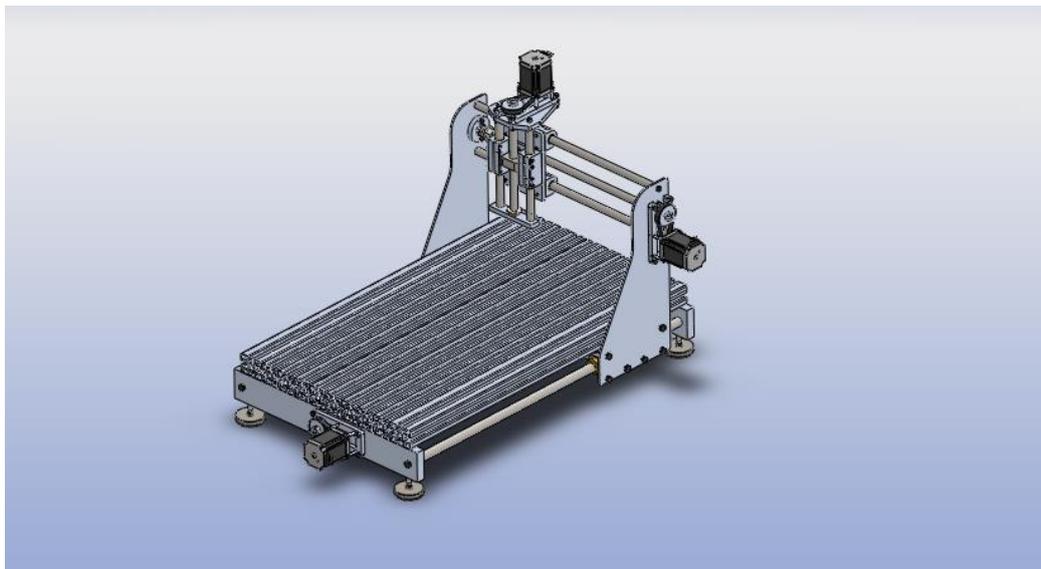


Figure IV. 11. : dispositif 3D

déplacements	Nom de pièce	$m_i$	Masse (kg)
<b>Direction Z</b>	Mini perceuse	$m_1$	0.550
	Paliers à douilles à billes	$m_2$	0.496
	Porte outil	$m_3$	0.286
	Vis trapézoïdale + écrou	$m_4$	0.486
	Support porte outil	$m_5$	0.730
	Axes de guidages	$m_6$	0.7
	Moteur	$m_7$	0.35
<b>Direction Y</b>	Paliers à douilles à billes	$m_8$	1.124
	Vis trapézoïdale + écrou	$m_9$	0.726
	Axes de guidages	$m_{10}$	1.518
	Moteur	$m_{11}$	0.35
	Paliers à douilles à billes	$m_{12}$	1.435
	Vis trapézoïdale + écrou	$m_{13}$	0.726
	Axes de guidages	$m_{14}$	1.462

<b>Direction X</b>	Le plateau	m <sub>15</sub>	1.696
	Moteur	m <sub>16</sub>	0.35

**Tableau IV. 1.:** les masses des pièces utilisées dans la conception

#### IV. 5. 2. Dimensionnement des vis trapézoïdale :

- Le rendement des vis et déécrous

$$\eta = (1 * f * tg\alpha) \div (1 * (f / tg\alpha))$$

Avec  $f = 0.18$  et  $\alpha = 5.2$

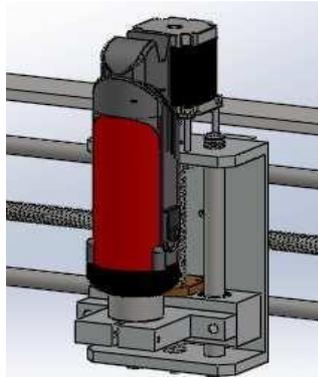
$f$  = coefficient de sécurité

$\alpha$  : L'angle de l'hélice

On trouve :  $\eta = 0.33$

##### 1) Dimensionnement de la vis trapézoïdale Z :

Où:  $F$  est la charge appliquée



**Figure IV. 12.:** système de mouvement Z

a. La charge appliquée sur la vis :

**F** la charge appliquée sur la vis trapézoïdale de l'axe Z est calculé par la formule suivante:  $F_z = (m \times g) + (m \times \gamma) + F_p$

Ou  $g$  : la gravité

$\gamma$  : L'accélération

$F_p$  : l'effort de pénétration

$m$  : l'ensemble des masse sur la vice

$F_p = 1/3 F_c K_c$  :

$F_c = a * f_a * K_c$

$F_c$  : l'effort de couple

$$K_c = 80 \text{ daN/mm}^2 = 800 \text{ N/mm}^2$$

$$a = 0.3 \text{ mm}$$

$$f_a = 0.4 \text{ mm}$$

d'où

$$F_c = 0.3 * 800 * 0.4$$

$$F_c = 96 \text{ N}$$

$$F_p = 1/3 F_c = 96/3$$

$$F_p = 32 \text{ N}$$

$$\alpha = \Delta v / \Delta t$$

On suppose que

$$\Delta t = 0,1 \text{ s}$$

et

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

$$v = 66,66 \text{ mm / s}$$

$$v = 0,06 \text{ m / s}$$

Donc

$$\alpha = 0,06 / 0,1$$

$$\alpha = 0,6 \text{ m / s}^2$$

$$m = m_1 + m_2 + m_3 = 1,332 \text{ Kg}$$

Avec:  $(m_i)$  sont l'ensemble des masses sur la vis et  $(g)$  est la gravitation. Après le calcul

on trouve :

$$P_z = C * \omega$$

$$\omega = (2 * \Pi * \underline{v}) / 60$$

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

$$\omega = 104,66 \text{ rad} / \text{s}$$

$$P_z = 0,089 * 104,66$$

$$= 9,31 \text{ watt}$$

## 2) Dimensionnement de la vis trapézoïdale Y :

### La charge appliquée sur la vis

**F** la charge appliquée sur la vis trapézoïdale de l'axe Y est calculé par la formule suivante:  $F_y = (m \times g) + (m \times \gamma) + F_c$

Ou  $g$  : la gravité

: L'accélération

$m$  : l'ensemble des masse sur la vice  $F_c$  : l'effort de couple

$$F_c = a * f_a * K_c$$

$K_c$  : coefficient de spécifique de coupe : profondeur de passe en mm

$f_a$  : avance en mm/tr

$$K_c = 80 \text{ daN/mm}^2 = 800 \text{ N/mm}^2 \quad a = 0.3 \text{ mm}$$

$$f_a = 0.4 \text{ mm}$$

$$F_c = 0.3 * 800 * 0.4 = 96 \text{ N}$$

$$\alpha = \Delta v / \Delta t$$

On suppose que

$$\Delta t = 0,1 \text{ s}$$

et

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

$$v = 66,66 \text{ mm} / \text{s}$$

$$v = 0,06 \text{ m} / \text{s}$$

**Donc**

$$\alpha = 0,06 / 0,1$$

$$\alpha = 0,6m / s^2$$

$$m = m_1 + m_2 + m_3 + m_4 + m_5 + m_6 + m_7 + m_8 = 4,722 \text{ Kg}$$

Avec: ( $m_i$ ) sont l'ensemble des masses sur la vis et ( $g$ ) est la gravitation. Après le calcul on trouve :

$$F_y = 146,58 \text{ N}$$

Pour des raisons de sécurité la force choisie doit être supérieure à la force calculée, donc on prend  $F = 150 \text{ N}$

### Le couple nécessaire :

Il est donné par la formule suivante :  $C_y = (F_y * p) / (2 * \eta * \pi * 1000)$

$$p = \text{pas de la vis} = 4 \text{ mm}$$

$$C_y = (146,58 * 4) / (2 * 3,14 * 0,33 * 1000)$$

$$C_y = 0,28 \text{ N.m}$$

### La puissance nécessaire :

$$P_y = C * \omega$$

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

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

$$\omega = 104,66 \text{ rd / s}$$

$$P_y = 0,28 * 104,66$$

$$P_y = 29,30 \text{ watt}$$

### 3) Dimensionnement de la vis trapézoïdale X :

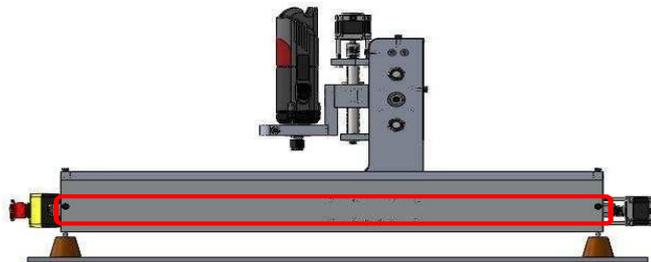


Figure IV. 14. : système de mouvement X

La charge appliquée sur la vis

**F** la charge appliquée sur la vis trapézoïdale de l'axe **X** est calculé par la formule suivante:  $F_x = (m \times g) + (m \times a) + F_c$

Ou  $g$  : la gravité

$a$ : L'accélération

$m$  : l'ensemble des masses sur la vis  $F_c$  : l'effort de coupe

$$F_c = a \cdot f_a \cdot K_c$$

$K_c$  : coefficient de spécificité de coupe : profondeur de passe en mm

$f_a$  : avance en mm/tr

$$K_c = 80 \text{ daN/mm}^2 = 800 \text{ N/mm}^2 \quad a = 0.3 \text{ mm}$$

$$f_a = 0.4 \text{ mm}$$

$$F_c = 0.3 \cdot 800 \cdot 0.4$$

$$F_c = 96 \text{ N}$$

$$\alpha = \Delta v / \Delta t$$

On suppose que

$$\Delta t = 0,1 \text{ s}$$

et

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

$$v = 66,66 \text{ mm / s}$$

$$v = 0,06 \text{ m / s}$$

Donc

$$\alpha = 0,06 / 0,1$$

$$\alpha = 0,6 \text{ m / s}^2$$

$$m = m_{12} + m_{15} = 3,131 \text{ Kg}$$

Avec: ( $m_i$ ) sont l'ensemble des masses sur la vis et ( $g$ ) est la gravitation. Après le calcul on trouve :

$$F_x = 129,19 \text{ N}$$

Pour des raisons de sécurité la force choisie doit être supérieure à la force calculée, donc on prend

$$F = 130 \text{ N}$$

### Le couple nécessaire

Il est donné par la formule suivante :  $C_x = (F_y \cdot p) / (2 \cdot \eta \cdot \Pi \cdot 1000)$

$p$  = pas de la vis = 4 mm

$$C_x = (129,19 \cdot 4) / (2 \cdot 0,33 \cdot \Pi \cdot 1000)$$

$$C_x = 0,25 \text{ N.m}$$

La puissance nécessaire

$$P_x = C^*$$

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

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

$$\omega = 104,66 \text{ rd / s}$$

$$P_x = 0,28 * 104,66$$

$$P_x = 26,16 \text{ watt}$$

## IV. 6. La Conception assistée par ordinateur (CAO) :

La CAO 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 fabriqué réellement, c'est-à-dire en virtuel. Il existe un grand nombre de logiciels de CAO, par exemple [36] :

- SolidWorks.
- FreeCAD.
- OpenCASCADE.
- QCAD..

Vu que nous sommes familiarisés durant notre formation avec SolidWorks, nous avons décidé de l'utiliser.

### IV. 6. 1. SolidWorks :

Créé en 1993 par l'éditeur américain éponyme, SolidWorks a été acheté le 24 juin 1997 par la société Dassault Systèmes [37].

SolidWorks est un logiciel de CAD (computer Aided Design) très puissant et facile à manipuler, dédié au design et la modélisation 3D. Il est très utilisé dans l'industrie pour la fabrication de différentes pièces mécaniques, hydrauliques, électriques ainsi que l'assemblage des pièces et la conception des machines et des appareils de A à Z.



Figure IV. 15. Logiciel SOLIDWORKS

#### IV. 6. 2. Fonctionnement:

SOLIDWORKS est un modéleur 3D utilisant la conception paramétrique. Il génère 3 types de fichiers relatifs à trois concepts de base : la pièce, l'assemblage et la mise en plan. Ces fichiers sont en relation. Toute modification à quelque niveau que ce soit est répercutée vers tous les fichiers concernés.

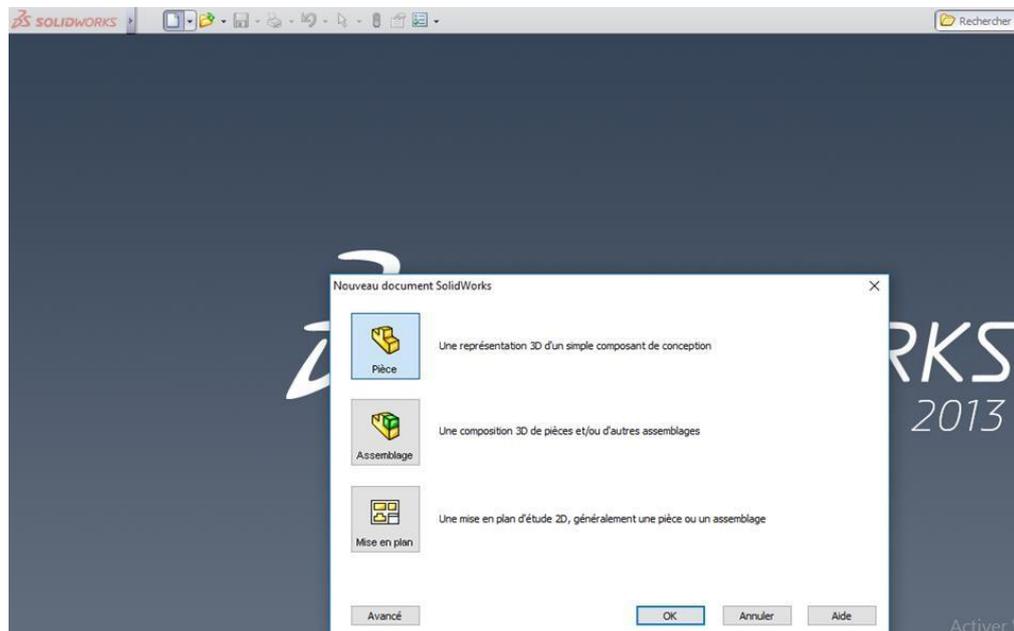


Figure IV. 16. : Une fenêtre du logiciel SOLIDWORKS

#### IV. 6. 3. Pièce:

La pièce est l'objet 3D monobloc. La modélisation d'une telle entité dépendra de la culture de l'utilisateur. Comme de nombreux logiciels conviviaux, SolidWorks permet

d'aboutir à un même résultat apparent par des voies souvent différentes. C'est lors de la retouche de ces fichiers ou de leur exploitation qu'on appréciera la bonne méthode.

Une pièce est la réunion d'un ensemble de fonctions volumiques avec des relations d'antériorité, des géométriques, des relations booléennes (ajout retrait)... Cette organisation est rappelée sur l'arbre de construction. Chaque ligne est associée à une fonction qu'on peut renommer à sa guise. Pour obtenir un volume, suivre les étapes :

- Définir une origine
- Choisir un plan
- Tracer une esquisse

Générer un volume.

#### IV. 6. 4. Assemblages:

Les assemblages sont obtenus par la juxtaposition de pièces. La mise en position de pièces est définie par un ensemble des contraintes d'assemblage associant, deux entités respectives par une relation géométrique (coïncidence, tangence, coaxialité...).

Dans une certaine mesure, ces associations de contraintes s'apparentent aux liaisons mécaniques entre les pièces. Le mécanisme monté, s'il possède encore des mobilités, peut être manipulé virtuellement. On peut alors aisément procéder à des réglages à l'aide des différents outils disponibles (déplacement composants, détection de collision ou d'interférence, mesure des jeux, etc.) .

##### Étapes de l'assemblage

- Ajouter des pièces dans un assemblage
- Déplacer et faire pivoter des composants dans un assemblage

On peut positionner et orienter les composants à l'aide de contraintes qui créent des relations entre les composants.

#### IV. 6. 5. Mise en plan:

Une fois les pièces ou assemblages créés, il est possible de générer automatiquement les mises en plan (représentation 2D) avec insertion automatique des côtes et liaisons entre les vues 2D et le modèle 3D. De plus, des fonctions d'habillage (texte, hachure, cotation...) permettent à l'utilisateur d'annoter rapidement un plan. Pour faire des mises en plan, il est tout d'abord nécessaire d'avoir des fonds de plan pour y projeter les dessins. Ces fonds de plans ont un format (A4, A3,...), une orientation (portrait

ou paysage) et contiennent éventuellement un cartouche. Uncertain nombre de fonds de plan de base sont proposés à l'origine, mais il est préférable, avant de commencer, de personnaliser les fonds de plan que l'on utilisera par la suite.

Étapes de création d'une mise en plan

- Ouvrir un modèle de mis en plan et éditer un fond de plan
- Insérer des vues standards d'un modèle de pièce
- Ajouter des annotations de modèle et de référence
- Ajouter une autre feuille de mise en plan e. insérer une vue nommée f. imprimer la mise en plan [38]

Pour la conception de notre machine CNC nous utilisant SolidWorks en modélisation 3D. On modifie chacune des pièces afin de les assembler pour obtenir la forme finale de la machine.



Figure IV. 17 : Fenêtre principale de SolidWorks

## IV. 7. La conception de notre machine CNC avec SolidWorks :

### A) PARTIE PIECES :

La conception des pièces de la machine, avec SolidWorks, se déroule en 3 étapes :

- Sélection du plan de face,
- Conception de la forme en 3D en utilisant l'option « bossage extrudé »
- Création des différents détails qui caractérisent chacune de pièces.

➤ *Rail de guidage :*

Nous avons défini la même base pour tous les rails mais avec des longueurs différentes.

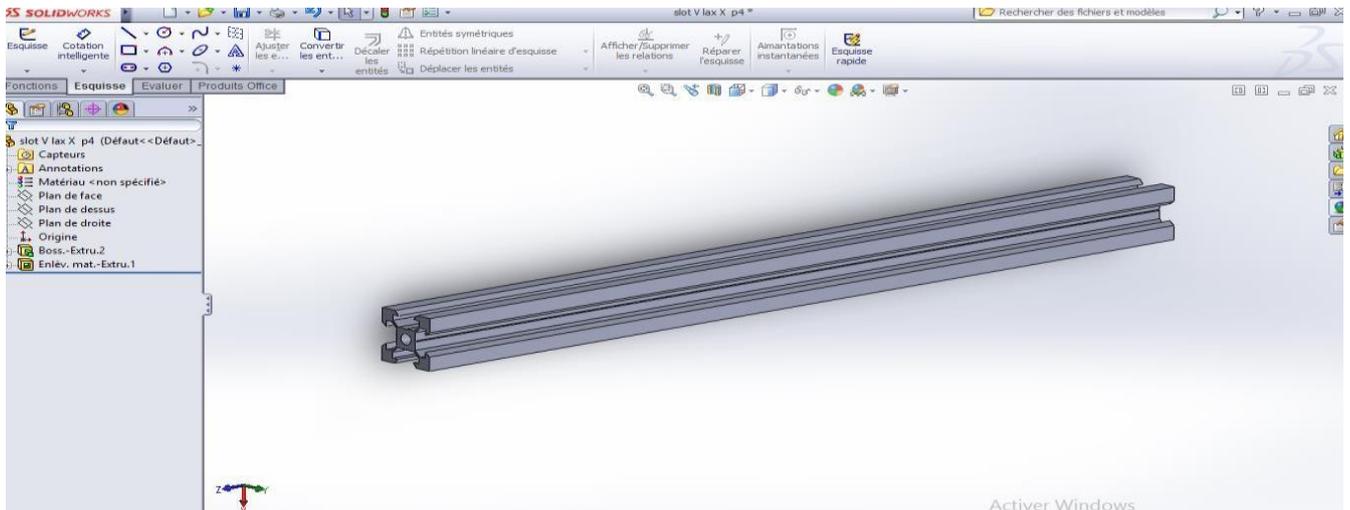


Figure IV. 18. : Rail de guidage.

Tableau IV. 2. : Caractéristiques de rails de guidage utilisés.

Axe	Dimensions (mm)	Nombre
Ox	20 × 20 × 280	5
Oy	20 × 20 × 320	2
Oz	20 × 20 × 220	2

➤ *Support de cadre :*

Des petits triangles sur la face latérale et deux trous pour les vis de fixation sont créés sur la forme en 3D de la pièce du Support de cadre.

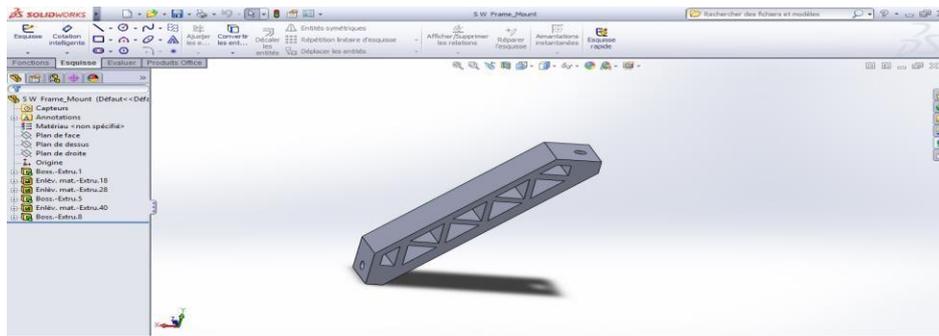


Figure IV. 2. : Support de cadre.

➤ *Les axes du guidage :*

Nous avons dessiné un cercle de 8mm de diamètre, après avoir défini la longueur de l'axe, nous avons obtenu la forme finale en 3D. Pour faciliter l'assemblage de la pièce avec le support, nous avons minimisé les extrémités de l'axe.

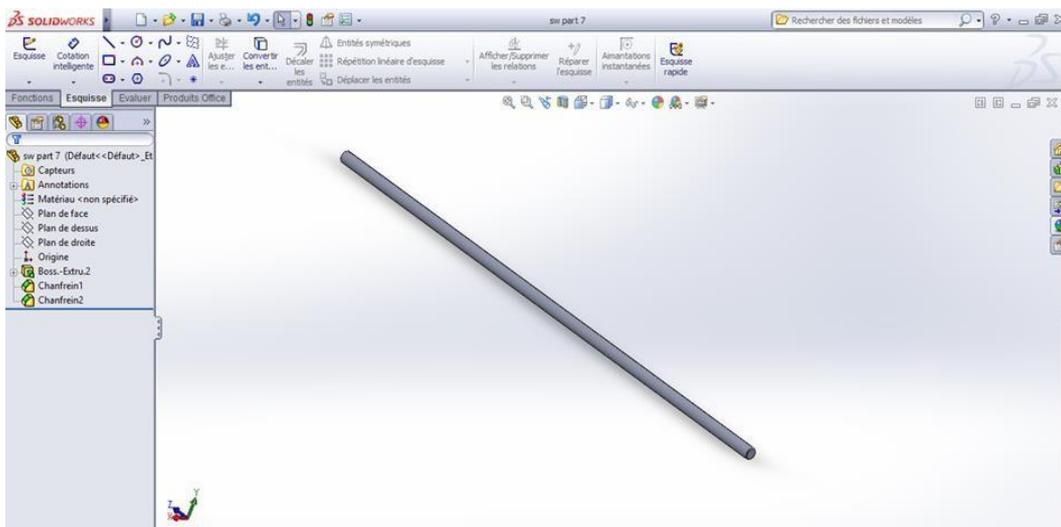


Figure IV. 19. : Axe de guidage.

➤ *Support de moteur :*

Il contient deux trous pour le fixer sur le rail et 5 points pour fixer le moteur dessus.

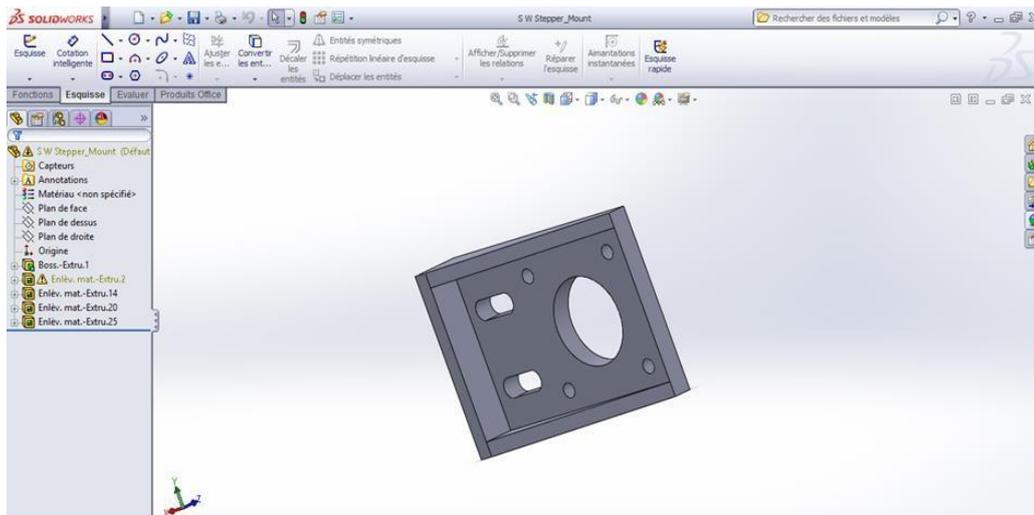


Figure IV. 20. : Support de moteur pas-à-pas.

➤ *Moteur pas à pas Nema 17*

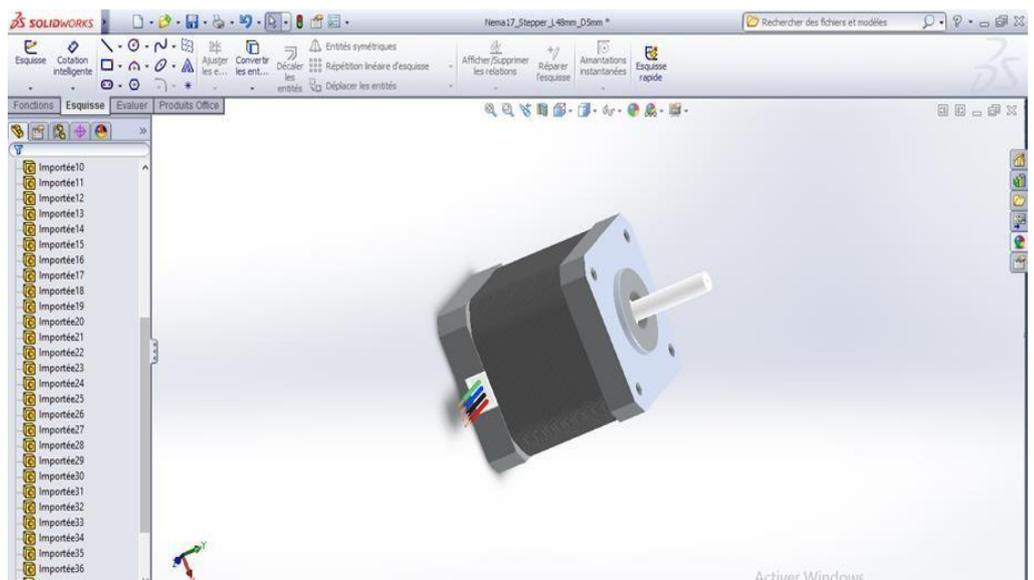


Figure 75 : Moteur pas-à-pas Nema 17.

➤ *Accouplement :*

Cette pièce cylindrique assure une liaison entre le moteur et la vis sans fin-écrou, ceci permet la transmission de rotation.

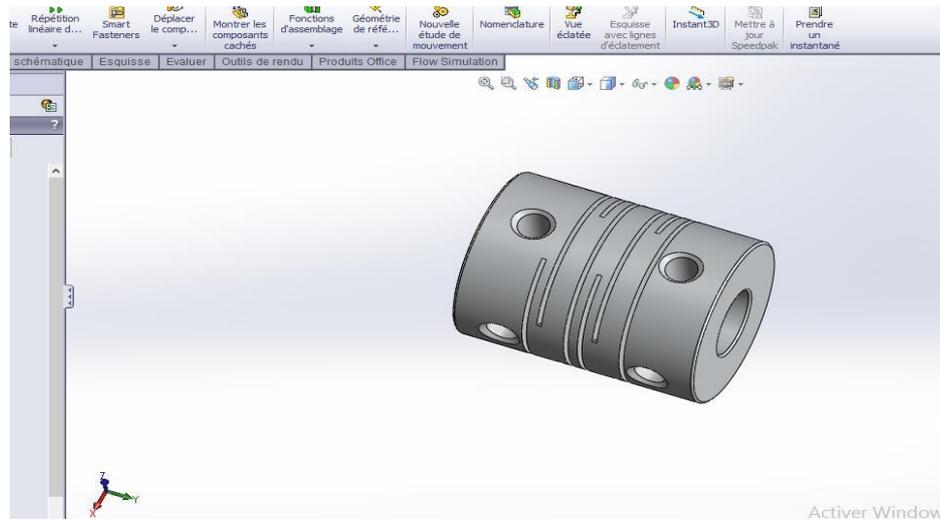


Figure IV.20. : Accouplement.

➤ *Vis sans fin (tige) :*

Cette vis est dessinée selon les caractéristiques présentées dans le tableau suivant

Type	Trapezoidal (mm)
Nombre de filet	4
Diamètre de vis	8
Pas	2
Longueur	300

Tableau IV. 3 : Caractéristiques de la vis utilisée.

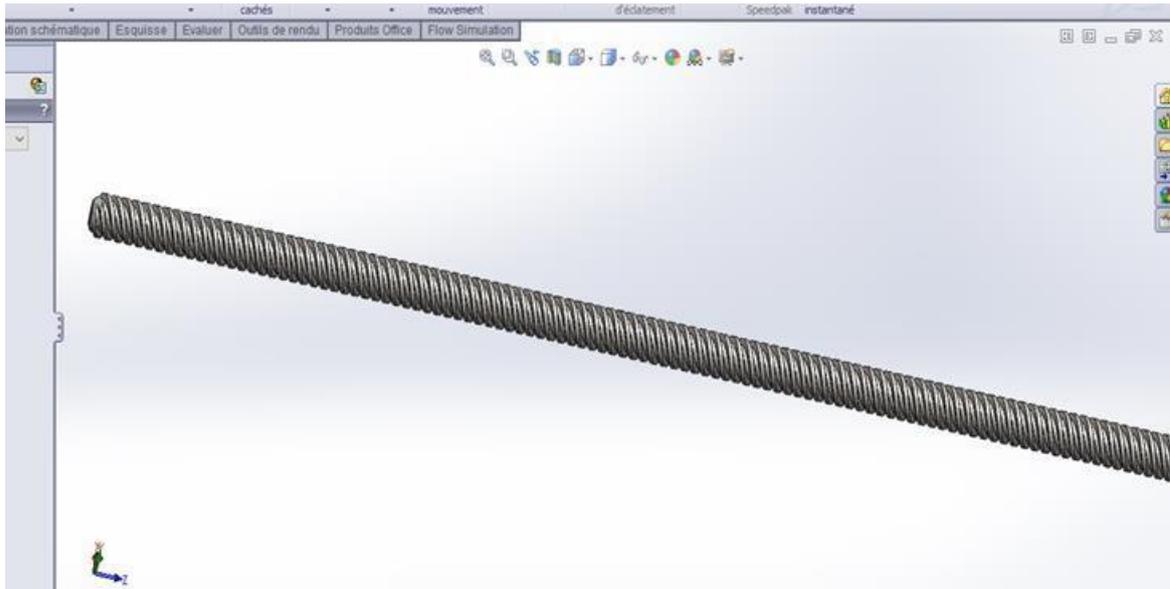


Figure IV. 22. : Vis sans fin.

➤ *Carriage :*

La pièce la plus compliquée, contient un trou pour la vis sans fin dans lequel se place l'écrou avec 4 points de fixation. De plus, elle contient deux entrées permettant le glissement des axes et des points de fixation latéraux pour le stylo.

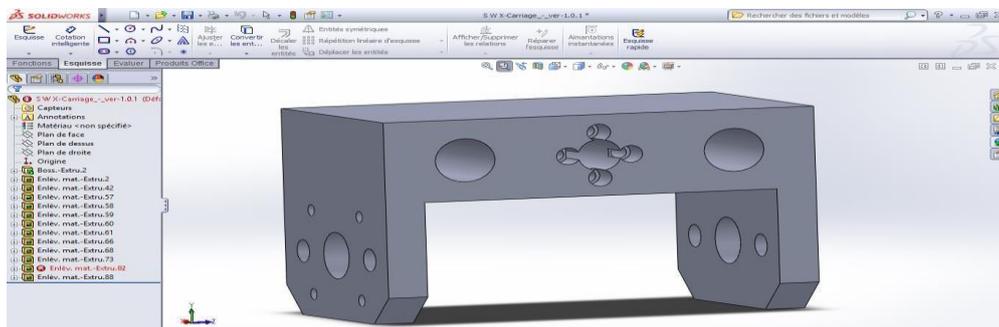


Figure IV.23. : Carriage.

➤ *Routeur Motion :*

Conçu pour se placer sur les axes et permet le déplacement de la table sur l'axe Y.

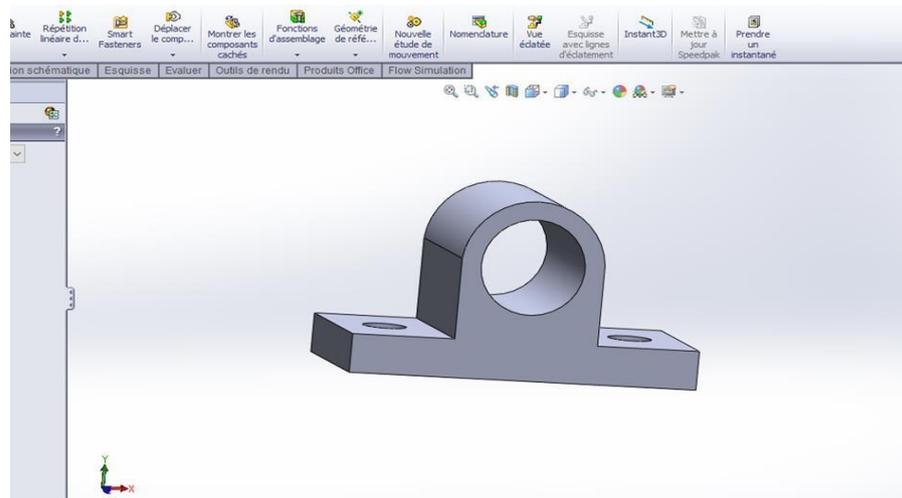


Figure IV. 24. : Routeur Motion.

➤ *La table :*

Pour qu'elle assure un mouvement sur l'axe Y, elle est conçue de façon à se fixer sur les quatre routeurs motion et les nuts.

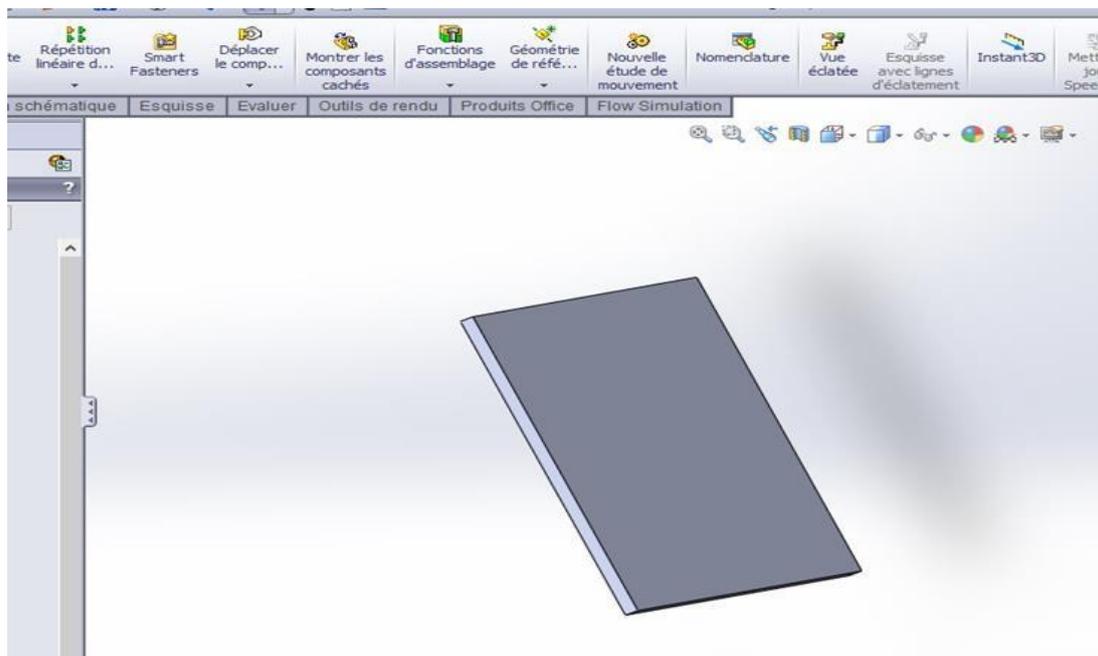


Figure IV. 25. : La table du CNC.

## B) PARTIE ASSEMBLAGES :

### Etape 1 : le cadre du CNC :

1. L'assemblage du cadre est constitué de neuf rails, tous reliés entre eux.
2. Deux supports cadre situées aux deux côtés.

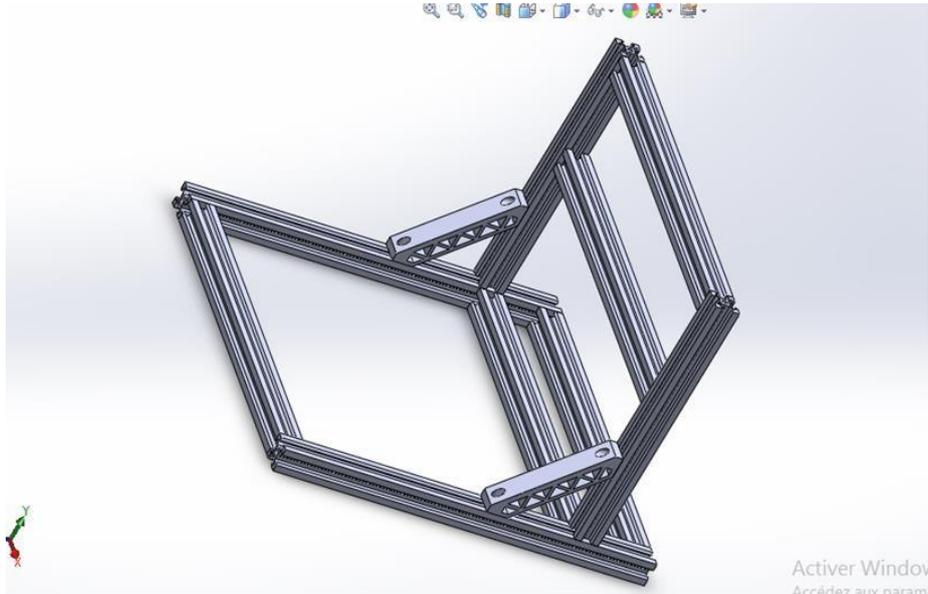


Figure IV. 26. : Forme d'assemblage du cadre de CNC.

### Etape 2 : l'assemblage sur l'axe X :

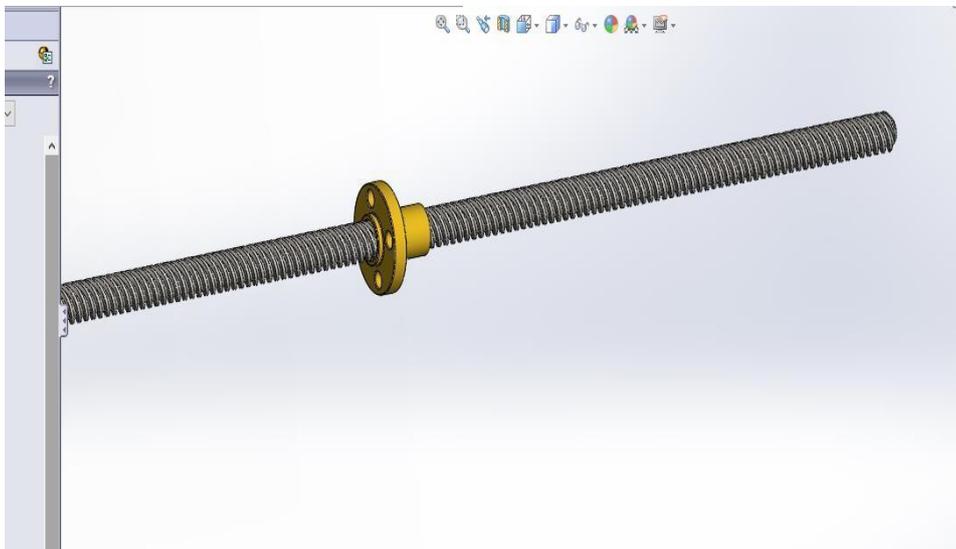


Figure IV. 27. : Assemblage de vis-écrou dans l'axe X.

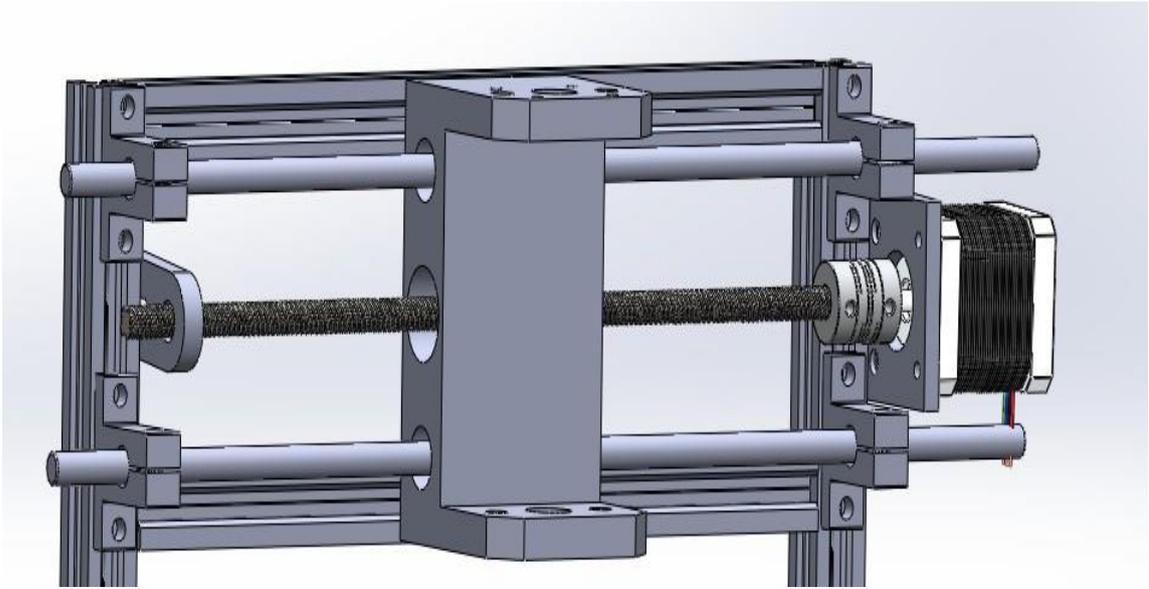


Figure IV. 28. : Assemblage final dans l'axe X.

### Étape 3 : l'assemblage sur l'axe Y :

Nous procédons comme suit

- Assemblage de vis-écrous (de même que pour l'axe X : Figure 89)
- Assemblage Routeur Motion avec Palier linéaire

Le résultat final de l'assemblage sur l'axe Y est illustré sur la figure suivante :

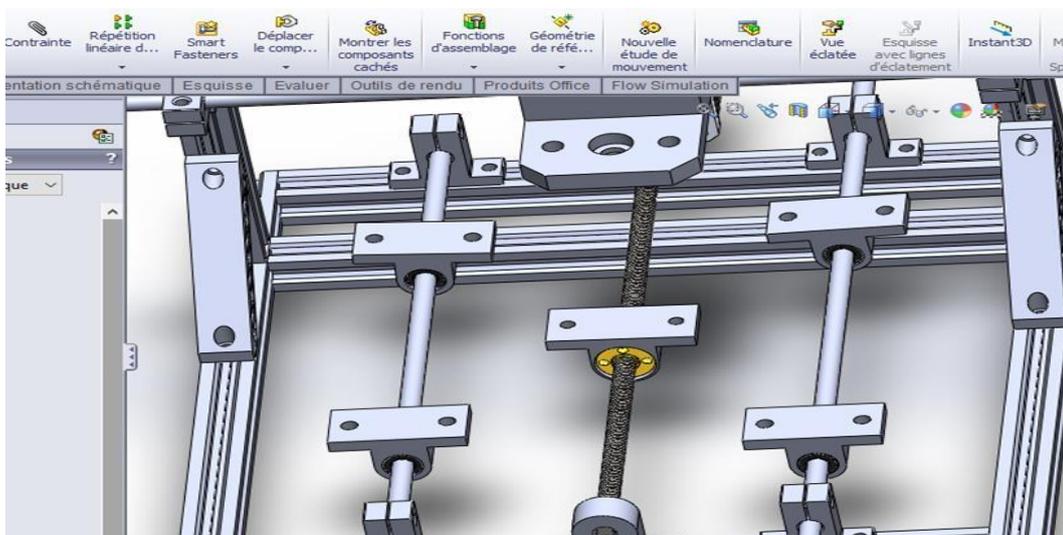


Figure IV. 29 : Assemblage final sur l'axe Y.

**Étape 3 : Assemblage du Moteur pas à pas nema17 sur les axes :**

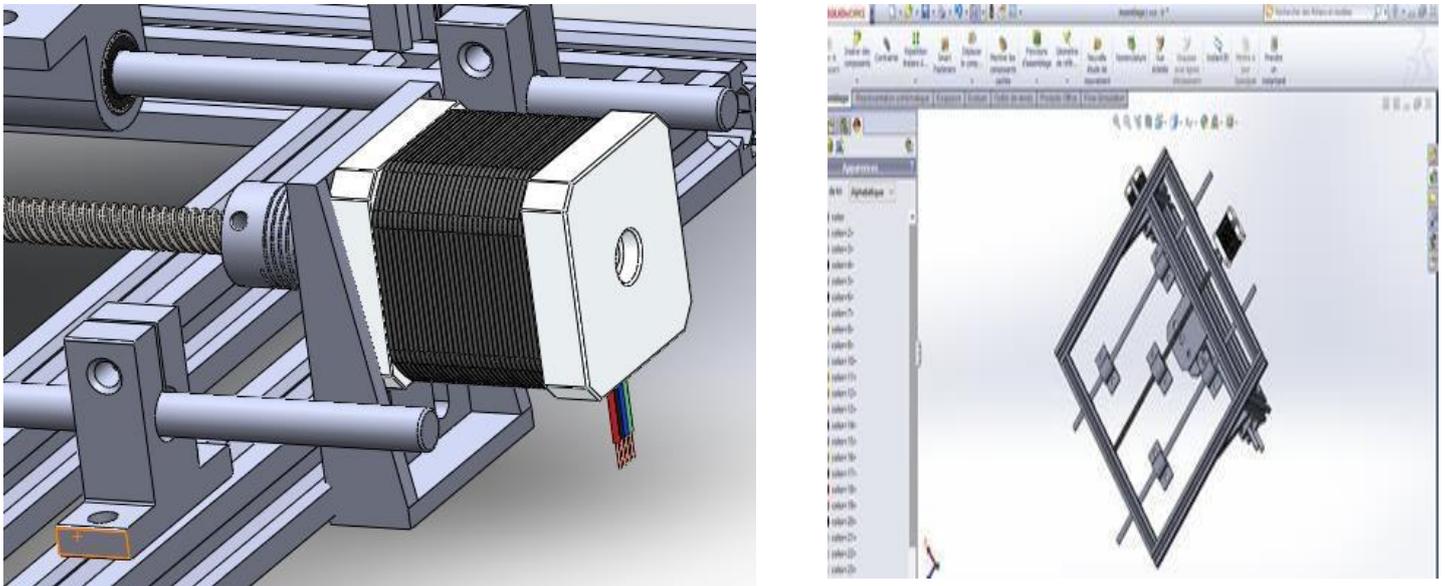


Figure IV. 30. : Moteur pas à pas nema17 dans l'axe Y.

**Étape 4 : l'assemblage final du CNC :** Dans cette étape, nous plaçons la table du CNC sur l'axe Y :

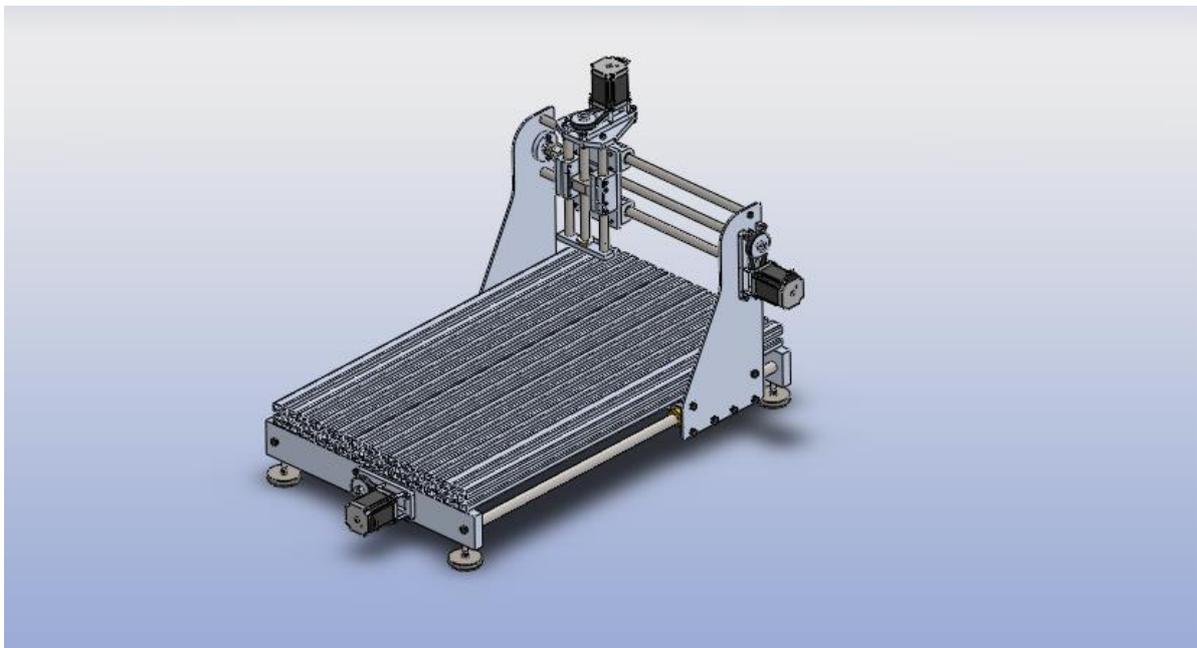


Figure IV. 31. : Forme finale de la machine CNC.



## IV. 8. Conclusion :

Malgré les difficultés rencontrées, notre objectif de concevoir une machine CNC est atteint. Nous avons réussi à faire fonctionner la machine on la commandant par nos propres programmes. Et on a concevoir les pièces nécessaire a l'aide de logiciel SolidWorks.







LSS Laboratory of Boumerdes Mr BENMISRA ABDELKADER 0777013173 State of progress of the work: **Subject N° 06 Etude et Conception d'une CNC.**

**Sujet N°06 Etude et Conception d'une CNC.**

**LSS Laboratoire de Boumerdes**

**Mr BENMISRA ABDELKADER 0777013173**

**Etat d'avancement des travaux des étudiants :**

- Cheriet Salah M 2 Fabrication Mécanique et Productique [Année 2021].
- Mesaouden Med Riadh M 2 Fabrication Mécanique et Productique [Année 2021].

State of progress of the students' work:

- Cheriet Salah M 2 Mechanical and Production Manufacturing [Year 2021].
- Mesaouden Med Riadh M 2 Mechanical and Production Manufacturing [Year 2021].

Objet : APERCU SUR LE SOMMAIRE DE PFE DE CHRIET & MESSAOUDEN M 2 F.M.P. [Année 2021].

Thème :

## *Etude et Conception d'une CNC*

**Proposé encadré et dirigé** : Mr. BENMISRA Abdelkader

# **Chapitre I**

## **Généralités sur les Systèmes de Fabrication en Programmation par Commande Numérique**

### **I. 1. Historique**

Il était en 1947 lorsque le contrôle numérique est né. Il a commencé quand John C. Parsons de la Parsons Corporation, Traverse City, Michigan, un fabricant de pales de rotor d'hélicoptère, ne pouvait pas faire ses modèles assez vite. Ainsi, il a inventé un moyen de couplage matériel informatique avec un foret de gabarit. M. Parsons a utilisé des cartes perforées pour faire fonctionner son chiffre système Ron. 1949 a été l'année d'un autre « **besoin urgent** ». La commande Matériel US Air a réalisé que les pièces pour ses avions et missiles ont été de plus en plus complexes. En outre, comme les dessins ont été constamment améliorés, les changements du dessin ont souvent été faits. Ainsi, dans leur recherche de méthodes de production plus rapide, un contrat d'étude Air Force a été attribué à la Parsons Corporation. Les mécanismes d'asservissement laboratoire de l'Institut de technologie du Massachusetts (MIT) étaient le sous-traitant. En 1951, le MIT a repris le travail complet, et en 1952, le prototype de la machine à commande numérique d'aujourd'hui, une machine

Cincinnati HydrogelMilling modifié, a été démontrée avec succès. Le terme contrôle numérique a été l'origine du MIT.

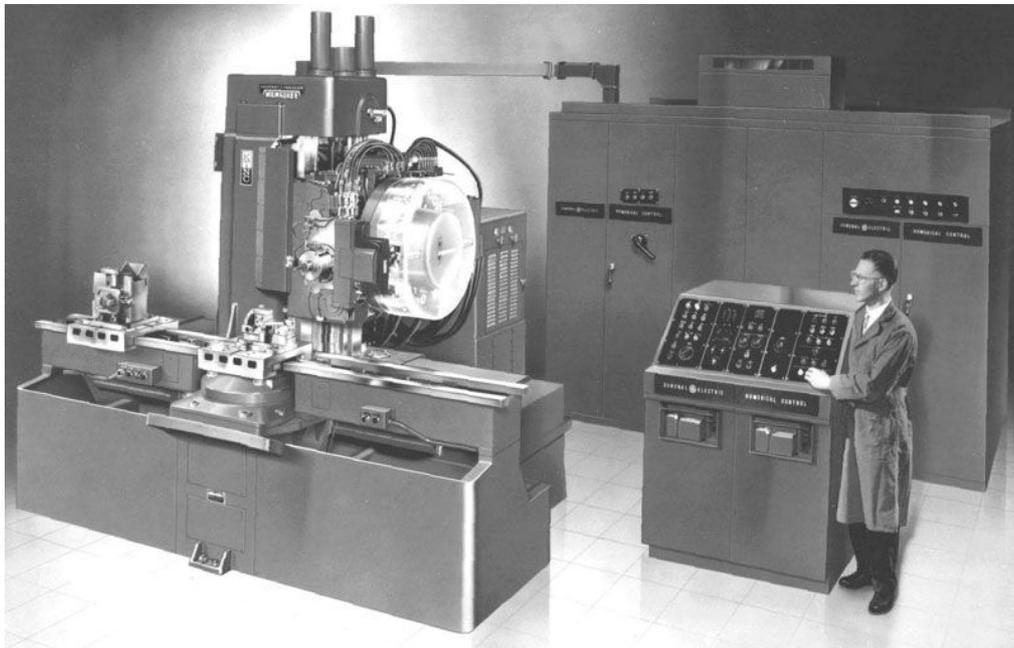


Figure. I.1 : Machine CNC en 1950

## I. 2. définition de la commande numérique

La commande numérique est un mode de commande dans lequel les valeurs désirées d'une variable commandée sont définies selon un **code numérique** (la machine-outil constitue le principal domaine d'application de la commande numérique). C'est une somme d'automatismes dans laquelle les ordres de mouvement ou de déplacement, la vitesse de ces déplacements et leur précision, sont donnés à partir d'informations numériques. Ces informations sont codées sur des supports tels que : rubans perforés, cassettes ou disquettes magnétiques ou simplement sauvegardés en « **mémoire** » dans le cas des dernières générations de commandes numériques à calculateur intégré (**CNC**). L'ensemble de ces informations de pilotage des **machines - outil** (MO) est élaboré sous forme de programme à exécution séquentielle. Les temps de réponse de telles commandes avoisinant la dizaine de microsecondes, il sera tout naturellement possible d'espérer piloter la machine suivant des trajectoires plus ou moins complexes, en vitesse et position.

### I. 3. Définition d'une machine CNC

C'est une machine totalement ou partiellement automatique à laquelle les ordres sont communiqués grâce à des codes qui sont portés sur un support matériel (disquette, USB, ...). Lorsque la machine-outil est équipée d'une commande numérique capable de réaliser les calculs des coordonnées des points définissant une trajectoire (interpolation), on dit qu'elle est à calculateur. Elle est appelée **CNC (Commande Numérique par Calculateur)**. Le premier rôle d'une machine **CNC (Computerized Numerical Control)**, est de générer des mouvements. Elle recevra des valeurs de positionnement, de vitesse et d'accélération et générera, suite à un traitement, des consignes numériques en sortie. Elle dispose d'une grande puissance de calcul et d'une plus grande souplesse d'utilisation qu'un automate programmable. Ce dernier est néanmoins qu'un complément de la commande CNC pour ce qui est de la gestion des entrées telles que : interrupteur, bouton d'arrêt d'urgence, ...etc.. En résumé, la commande CNC va générer des mouvements selon des consignes numériques.



**Figure. I.2:** Une machine CNC router HD-1325

### I. 4. Principe de fonctionnement d'une machine à commande numérique:

Une machine-outil à commande numérique est composée de deux principales parties une **partie command** est une **partie opérative**, voir Figure

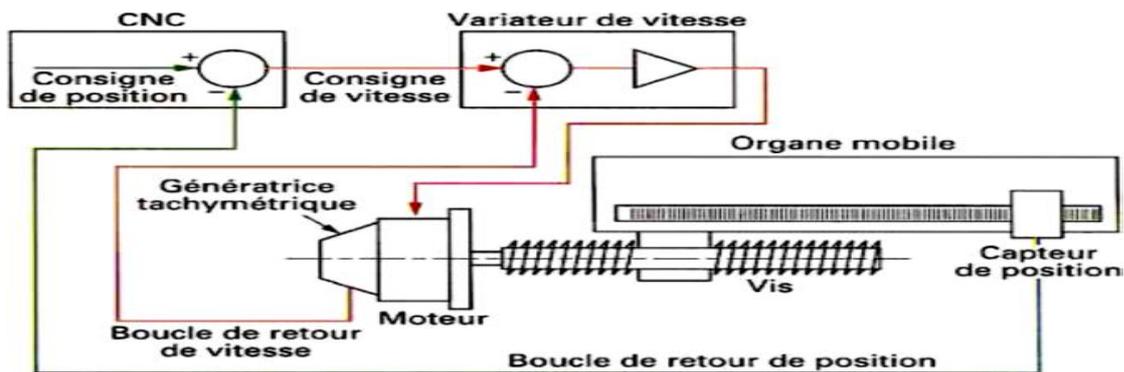
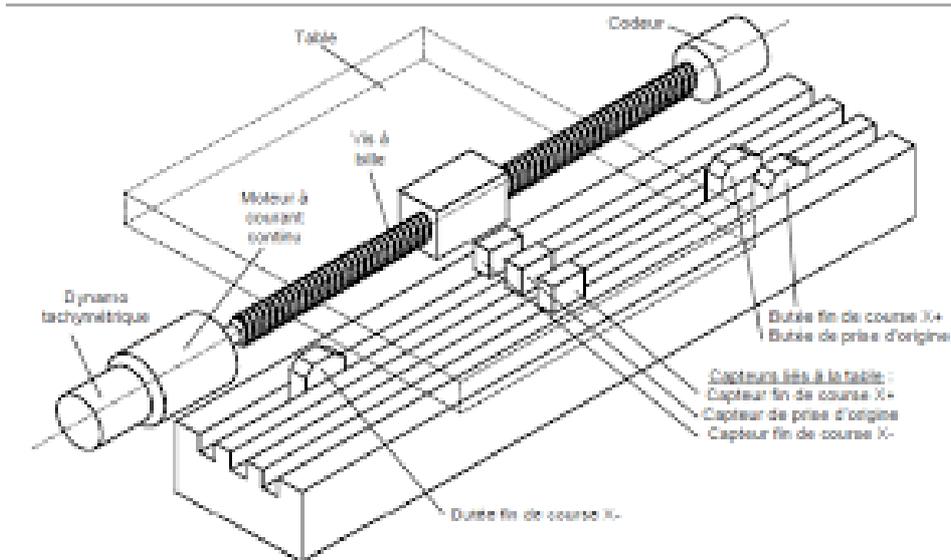


Figure. I.3 : Asservissement de la machine

I. 4.1. **Partie commande:** 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.



**Figure. I.4.** Eléments de la partie opérative.

**Tâche effectuée:** 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.

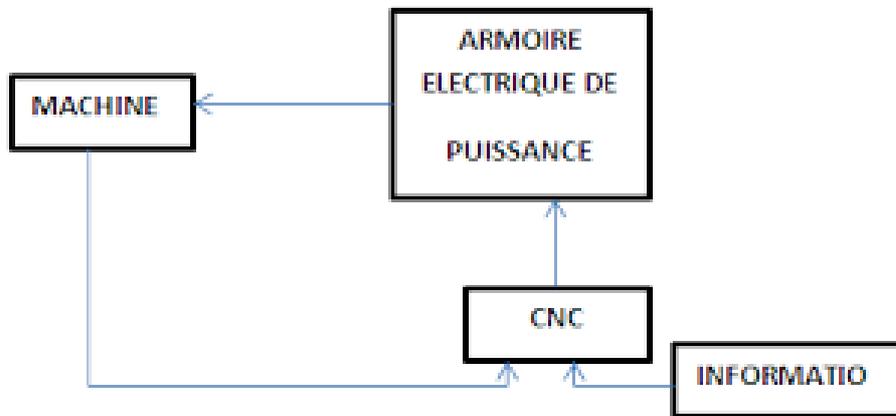
**Armoire électrique de relayage ou armoire de puissance:** Elle est composée :

- Automate programmable gérant toutes les entrées – sorties ;
- Relais ;
- Electrovanne ;
- Cartes variateurs d'axes (une par axe) ;
- Contacteurs (1 par élément de machine : axes, broche) ;
- Interrupteur général avec sécurité.

**I. 4.2. Partie opérative:** Différente d'une machine conventionnelle et constituée d'une armoire dans laquelle on trouve :

- Pupitre permettant de rentrer les commandes à l'aide d'un clavier,
- Lecteur de données (ce lecteur peut être une option lors de l'achat de la machine),
- Sortie RS 232 pour les liaisons avec les Périphériques externes,
- Ecran de visualisation de toutes les données enregistrées,

- Calculateur,
- Cartes électroniques (commandes d'axes, mémoire ...). La partie commande est alimentée par un courant faible et ne peut donc pas alimenter en direct les moteurs de la machine (voir schéma ci-dessous).



**Figure I.5.** Fonction originale d'une commande numérique.

## I. 5. Domaine d'utilisation

L'utilisation de la commande numérique ne se limite pas aux machines-outils travaillant par enlèvement de la matière avec des outils coupants. Elle est présente sur des installations de découpe par faisceau laser, en électroérosion que ce soit en

Défonçage ou en découpe par fil, en poinçonnage ou pliage de produits en feuille, pour la mise en place des composant, lors des opérations d'assemblage,... Elle sert aussi à piloter des tables traçantes, les machines à mesurer tridimensionnelles, les robots. La machine à outil à commande numérique représentent le moyen de production le plus important des pièces mécaniques.

Elle nécessite des gestes précis et/ou répétitifs pour effectuer diverses opérations: percer, scier, rectifier, découper, fraiser, plier, graver, tarauder, souder, visser, déposer un matériau. Les matériaux qui peuvent être usinés sont très divers, la caractéristique principale qui les différencie est la dureté. Plus le matériau est dur, plus il faudra que la CNC, ainsi que l'outil qui l'équipe aient la qualité et la puissance nécessaires pour usiner le matériau. D'où de nombreux dispositifs CNC différent entre eux principalement par l'outil qui est utilisé.

Les machines CNC sont employées dans de nombreux secteurs industriels : métallurgies, bois, textiles... Elles sont associées à des nouvelles technologies ; laser, électroérosion, jet d'eau.

## I. 6. les différents types de machines CNC

On distingue plusieurs types de machines :

**Les machines à enlèvement de copeaux** : les perceuses, les tours 2 et 4 axes, les centres de tournage 5 axes, les fraiseuses 2 axes, 3 axes, les centres d'usinage, 3 à 5 axes, les rectifieuses, les affuteuses, les machines d'usinage à très grande vitesse.

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

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

**Les presses** : métal, injection plastique.

**Les machines à bois** : à portique ou col de cygne.

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

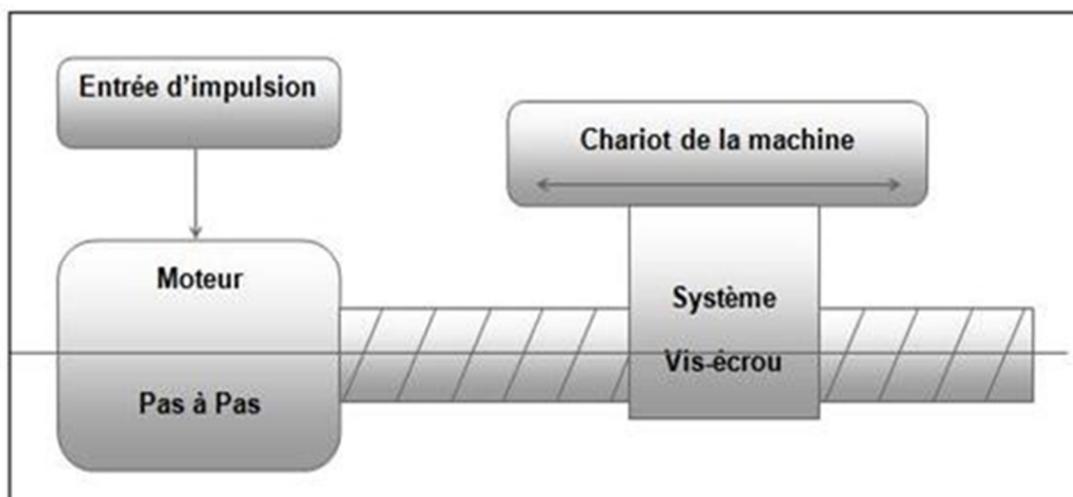
## I. 7. Classification des MOCN

Les machines-outils à commande numérique (MOCN) sont classées suivant le : Mode de fonctionnement de la machine, Nombre d'axes de la machine, et le mode d'usinage.

### I. 7.1. Classification des MOCN selon le mode de fonctionnement

#### I. 7.1.1. Fonctionnement en boucle ouverte

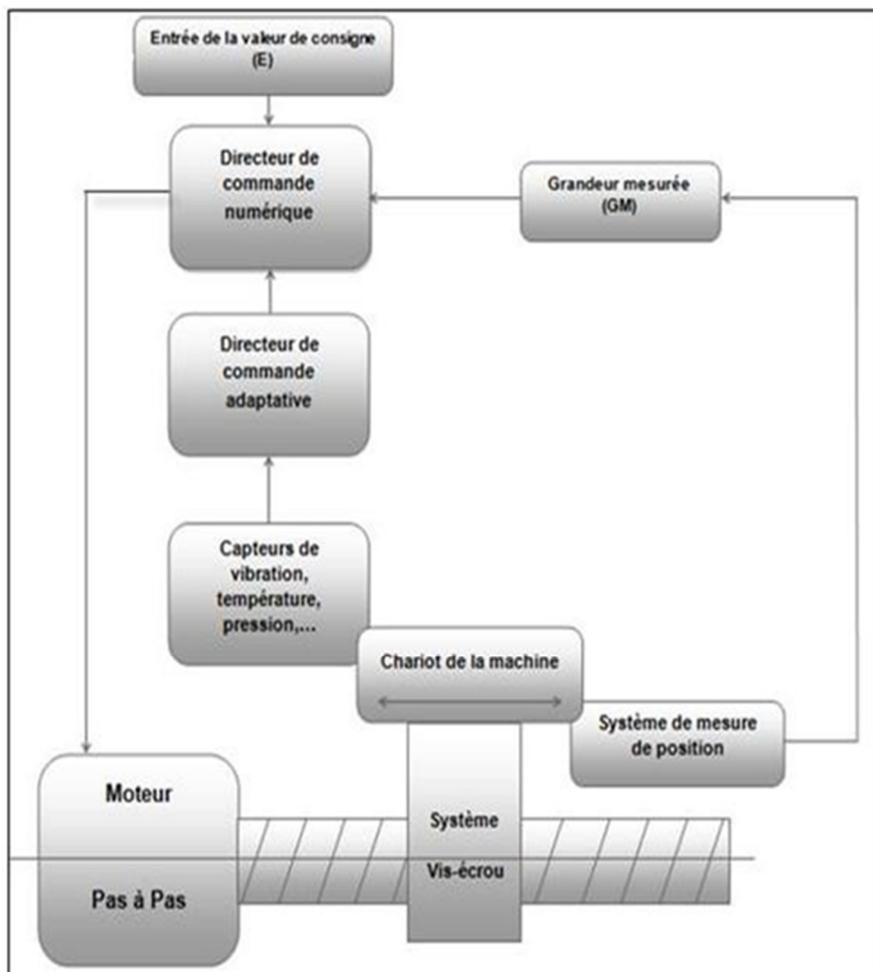
En boucle ouverte, comme la Figure I.5, le système assure le déplacement du chariot mais ne le contrôle pas.



**Figure I.6.** Fonctionnement en boucle fermée.

**I. 7.1.2. Fonctionnement avec commande adaptative**

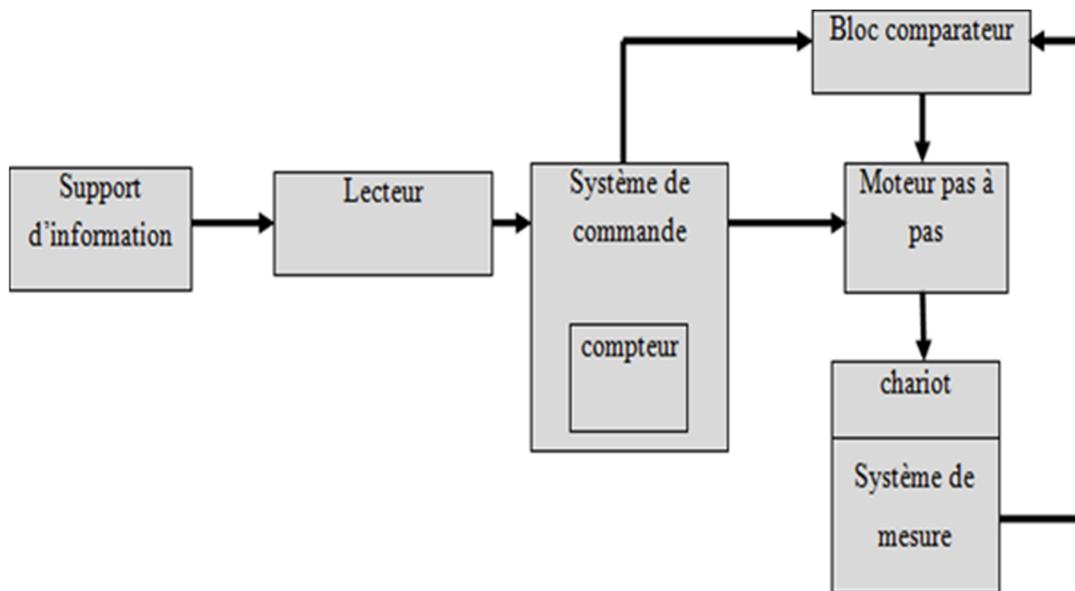
La commande adaptative réalise d'une façon continue et automatique l'adaptation des conditions de coupe. Des capteurs relèvent les valeurs de couple de la broche, l'amplitude de vibration de la broche, la température au point de coupe. Ces informations sont transmises à une unité spéciale qui les envoie 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.7.** Commande adaptative.

**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).comme illustre la (Figure I.6)



**Figure I.8.** Fonctionnement en boucle fermée.

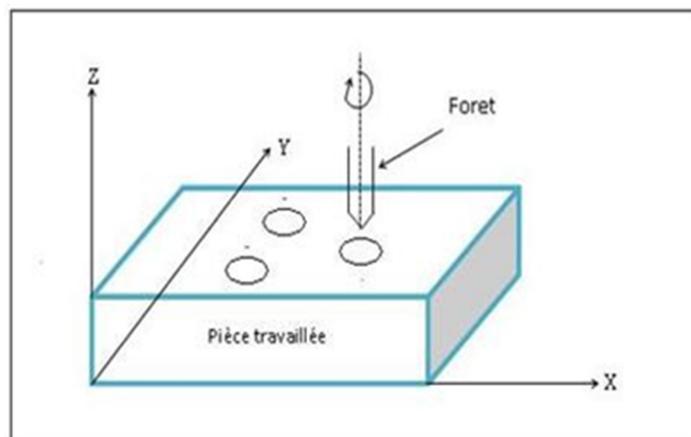
### I. 7.2. Classification des MOCN selon le mode d'usinage

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. 7.2.1. Commande numérique point à point

C'est la mise position de l'outil ou de la pièce Par déplacements non synchronisés. Le Mouvement de coupe (usinage) n'est possible que lorsque le mouvement de positionnement. Exemples d'opération d'usinage : perçage, alésage, lamage taraudage, petit fraisage.

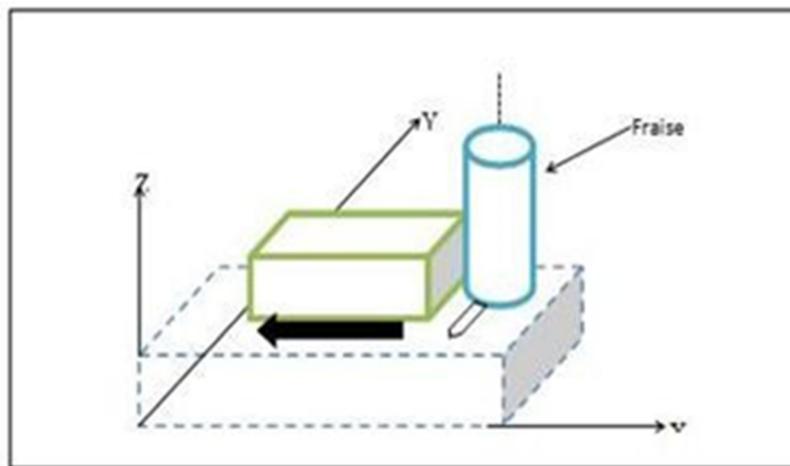


**Figure I.9.** Commande Numérique point à point.

### *I. 7.2.2. Commande numérique par axiale*

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

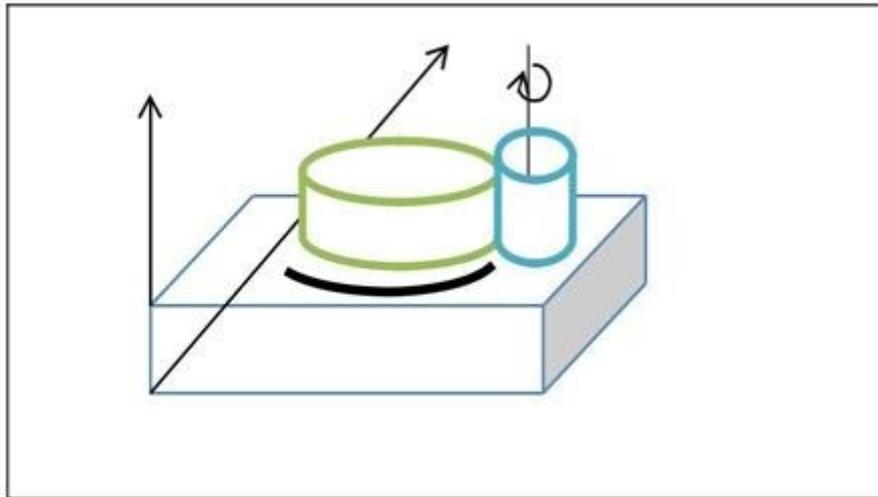
Exemples d'opération d'usinage : tournage, fraisage, alésage.



**Figure I.10.** Commande paraxiale.

### *I. 7.2.3. Commande numérique de contourage*

Ce sont des déplacements synchronisés des divers axes avec la vitesse d'avance programmée. Les trajectoires sont décomposées en éléments de droites ou de cercles dans un ou plusieurs plans. Exemples d'opération d'usinage : toute opération possible sur un centre de tournage ou Centre d'usinage.



**Figure I.11.** Commande numérique de contournage.

### I. 7.3. 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 continu. Un demi-axe définit la direction dans laquelle l'avance n'est pas contrôlable numériquement mais contrôlé par pistes, cames ou plateaux diviseurs.

Translation			Rotation	
primaire	Secondaire	Tertiaire	Primaire	Secondaire
X	U	P	A	D
Y	V	Q	B	E
Z	W	R	C	

**Tableau I.1.** Axe des différents mouvements possibles.

Chaque mouvement de translation ou de rotation est donc représenté par un axe défini une lettre affectée de signe + ou - . La figure.1.11. et la montre l'ensemble des axes qu'un DCN peut contrôler.

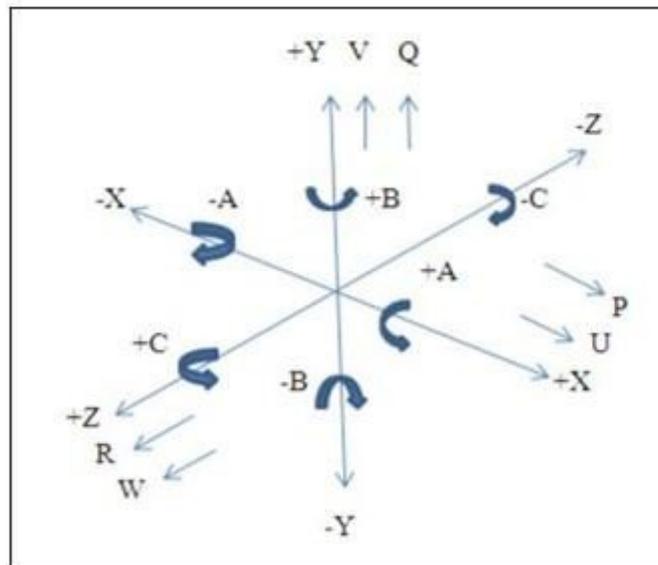


Figure I.12. Axes Primaires Et Axes Additionnels.

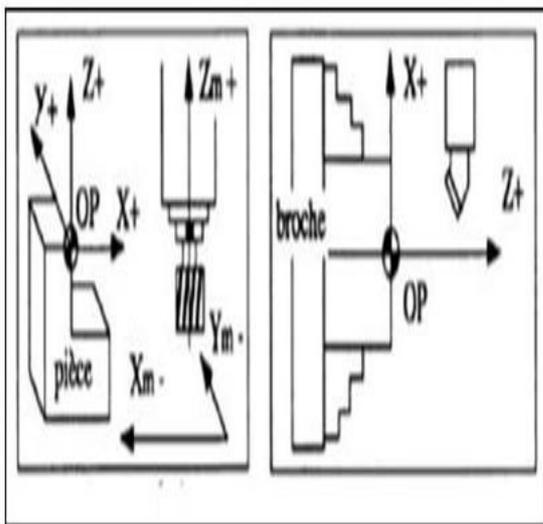


Figure I.13. Axes fraiseuse et tour.

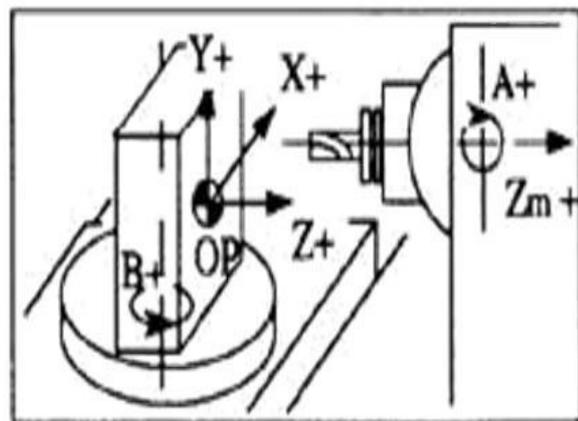


Figure I.14. Axes en centre de fraiseuse.

## I. 8.la structure d'une machine CNC

Ce système peut être divisé en trois parties. Le système **mécanique** qui reçoit des signaux de commande nécessaire au système **électronique** qui aboutit finalement à l'actionnement souhaité des moteurs. Les systèmes **électroniques** obtiennent une commande ou un ensemble de commandes du système logiciel et génère des commandes pour le système **mécanique**.

### I. 8.1. le système électronique

Le système électronique est responsable de la génération du signal de commande pour les moteurs qui guide le mouvement de la trajectoire de l'outil dans chaque direction ou axe. Le système électronique est composé de : L'alimentation, La carte microcontrôleur, La carte de commande des moteurs pas à pas.

### I. 8.2.le système mécanique

Le système mécanique est assemblé de telle manière que le mouvement des 3 axes soit réalisé en utilisant des rails linéaires assemblés avec des roulements linéaires, les moteurs sont montés chacun sur un axe qui est source de mouvement et agit en fonction du signal de commande généré à partir du circuit électronique. Chaque moteur est relié à une vis sans fin de chaque axe qui est chargé de transformer le mouvement du moteur de rotation en mouvement linéaire. Le mouvement contrôlé de chaque axe est obtenu directement par la commande de la rotation du moteur. La vitesse du mouvement de chaque axe peut également être contrôlée par le contrôle direct de la vitesse du moteur, en donnant des signaux de commande nécessaires. Ainsi, la trajectoire de l'outil fixée à l'organe terminal est contrôlée dans chaque axe pour une action sur la pièce à usiner. Pour éviter que les axes sortent de la plage de fonctionnement, on utilise des fins de course pour chaque axe. Une commande manuelle de bouton d'arrêt d'urgence peut également être utilisée pour la machine.

### I. 8.3. Le système de programmation

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

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

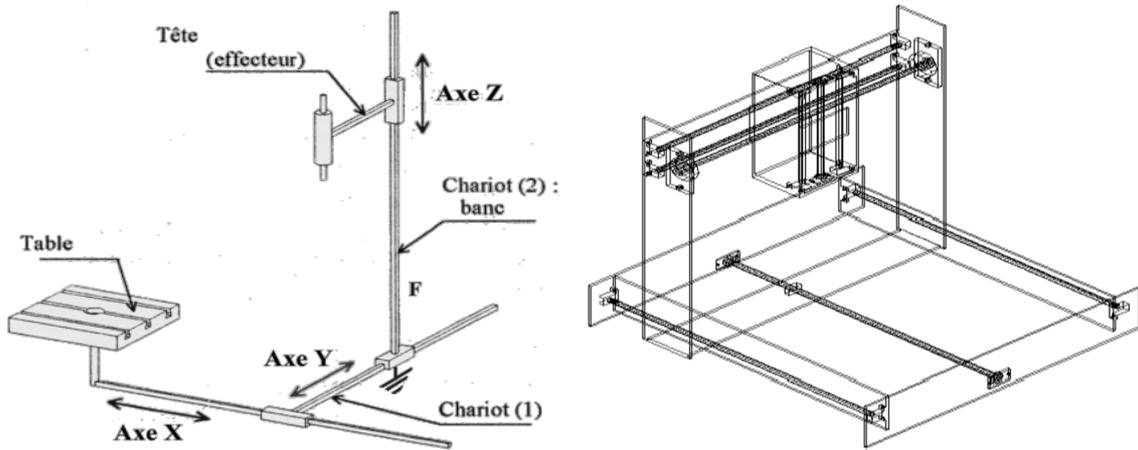


Figure I.15. : structure mécanique d'une machine CNC

### I. 9. Schéma simple d'une machine à commande numérique

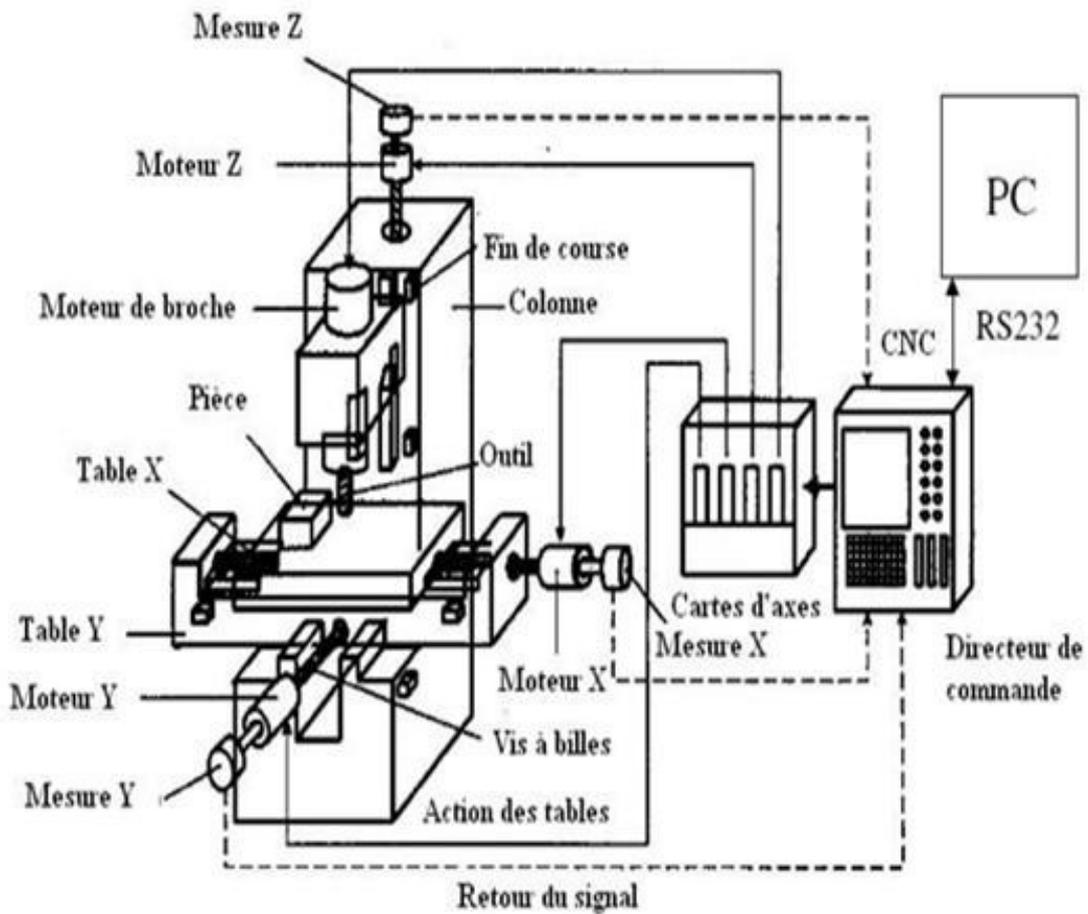


Figure I.16. Architecture d'une machine à commande numérique.

## I. 10. Conclusion

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

## II. 1. Introduction :

La machine CNC peut être divisée en trois parties. Le système mécanique qui reçoit des signaux de commande nécessaire du système électronique qui permet finalement à actionnement souhaité des moteurs. Les systèmes électroniques obtiennent une commande ou un ensemble de commandes du système logiciel et génère des commandes pour le système mécanique. Les principes et les généralités sur des machines à commande numérique avaient été décrit dans le chapitre précédent. Dans ce chapitre nous allons détailler le système électronique qui est responsable de la génération du signal de commande pour les moteurs qui guide le mouvement de la trajectoire de l'outil dans chaque direction ou axe.

Le système électronique est composé de :

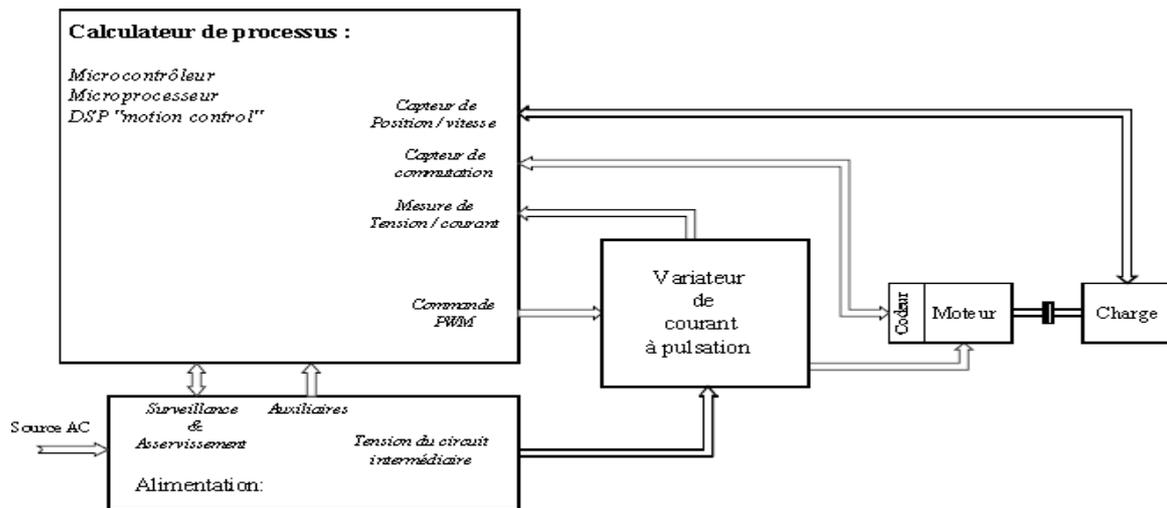
- L'alimentation
- La carte microcontrôleur
- La carte de commande des moteurs pas à pas

## II. 2. Les Actionneurs :

### II. 2. 1. 1. Généralité sur les moteurs pas à pas :

Le moteur pas à pas constitue un convertisseur électromécanique destiné à transformer le signal électrique (impulsion ou train d'impulsions de pilotage) en déplacement (angulaire ou linéaire) mécanique. Au point de vue électrotechnique, le moteur classique ressemble à la machine synchrone, dont le stator (le plus souvent à pôles saillants) porte les enroulements de pilotage et le rotor (presque toujours à pôles saillants) est soit muni d'aimants permanents (structure dite polarisée ou active), soit constitué par une pièce ferromagnétique dentée (structure dite réluctance ou passive). Entre le moteur et son alimentation, sont intercalés trois éléments essentiels (Figure 1) - une unité de calcul, qui élabore les impulsions de commande ;

- un modulateur PWM, qui génère les commandes des contacteurs électroniques de commutation ;
- une électronique de commutation (puissance), qui, à partir d'une alimentation, fournit l'énergie vers les enroulements appropriés du moteur.



**Figure II. 1.** : Schéma bloc de principe du contrôle d'un moteur pas-à-pas

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

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

- d'interaction champ du stator- fer du rotor, lorsque ce dernier présente une structure dentée passive.
- d'interaction champ du stator- champ du rotor, lorsque le rotor comporte un aimant.
- les deux interactions précédentes simultanément.

Les moteurs couramment rencontrés selon le nombre de pas par

tour sont :  $0,9^\circ$  soit 400 pas par tour

$1,8^\circ$  soit 200 pas par tour

$15^\circ$  soit 24 pas par tour

Les moteurs pas à pas présentent une très grande précision et une durée de vie quasi illimitée, l'usure mécanique étant pratiquement inexistante et ils existent en différentes tailles qui varient entre 1cm et plus d'une dizaine de centimètre. Tout dépendra des applications dans ils seront utilisées.

## II. 2. 1. 2. Les différents types de moteurs pas à pas :

On peut classer les moteurs pas à pas en 3 catégories :

- Les moteurs à aimant permanent
- Les moteurs à reluctance variable
- Les moteurs hybrides

### a) Moteurs à reluctance variable

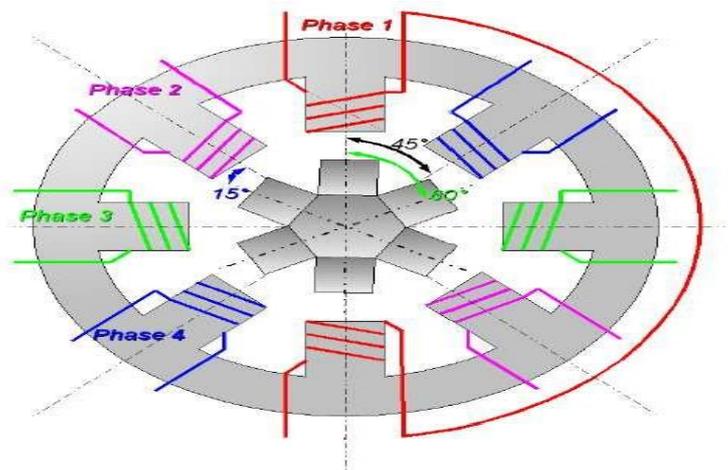
Les moteurs à reluctance variable (moteurs MRV) doivent leur nom au fait que le circuit magnétique qui les compose s'oppose de façon variable à sa pénétration par un champ magnétique.

Ces moteurs sont composés d'un barreau de fer doux et d'un certain nombre de bobines. Lorsqu'on alimente une bobine, elle devient un électroaimant et le barreau de fer cherche naturellement à s'orienter suivant le champ magnétique. On alimente la phase 1, puis la phase 2, puis la phase 3 ... Si on veut changer le sens du moteur, il suffit de changer l'ordre d'alimentation des bobines.

Dans la pratique, le barreau de ferrite a plusieurs dents (dans notre exemple est 6). Dès qu'on alimente la phase 2, il y a une rotation de  $15^\circ$  (i.e.  $60^\circ - 45^\circ = 15^\circ$ ), puis la phase 3, etc. Donc le moteur tourne de  $15^\circ$  dès qu'on alimente une phase.

Il faut 24 impulsions pour faire un tour complet. C'est un moteur

24 pas. Comme le montre la figure suivante :



### Figure II. 2: Moteur a reluctance variable

**Inconvénients** : nécessite au moins trois bobinages, pour obtenir un cycle complet, pas de couple résiduel, c'est-à-dire que hors tension, le rotor est libre, ce qui peut être problématique pour ce genre de moteur. La fabrication est assez délicate, les entrefers doivent être très faibles.

**Avantages du système** : peu coûteux, d'une bonne précision. Dans l'exemple, avec seulement 4 enroulements, on obtient 24 pas (on peut facilement obtenir 360 pas). Le sens du courant dans la bobine n'a aucune importance.

### B. Moteur à aimants permanents

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.

Une façon simple de voir le système, est de placer une boussole entre deux aimants. Suivant la bobine qui est alimentée et le sens du courant, l'aimant va s'aligner avec le champ.

Il utilise le principe de l'action d'un champ magnétique sur un moment magnétique (aimant). On distingue deux (2) types de ces moteurs :

#### 1. MOTEUR UNIPOLAIRE :

Le rotor est constitué par un aimant permanent (en ferrite par exemple) comportant une paire de pôles. Le stator, comprend deux circuits magnétiques décalés de  $90^\circ$ . Les enroulements à point milieu sont alimentés avec une polarité toujours du même signe, d'où le nom d'unipolaire.

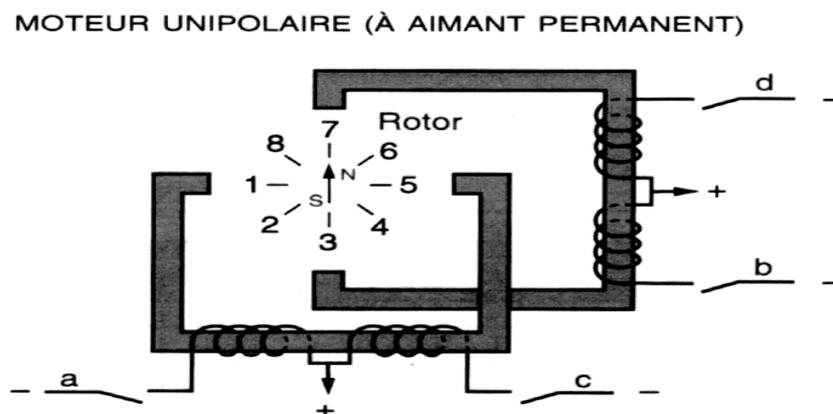


Figure II.3: Moteur unipolaire

Pour inverser le sens de rotation il suffit d'inverser les séquences de commutation.

## 2. Moteur bipolaire

Les enroulements au stator n'ont pas de points milieu, Chaque borne de chaque enroulement est alimentée successivement par une polarité positive puis négative d'où le terme bipolaire, En inversant les polarités des enroulements statoriques, on inverse les pôles nord et sud au stator.

Le nombre de phases est égale au nombre d'enroulement.

Moteur 2 phases, 2 pôles au rotor, alimentation bipolaire.

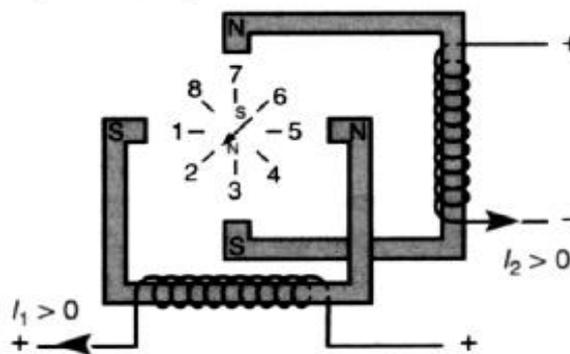


Figure II.4. : Moteur bipolaire

Le sens de rotation dépend du sens du courant et de l'ordre d'alimentation du bobinage.

Caractéristiques principales :

- Faible résolution : nombre de pas / tour peu important ;
- Couple d'utilisation plus élevé par rapport au moteur à reluctance variable ;
- Présence d'un couple résiduel lorsque le moteur est hors tension.

### C. Moteur hybride

C'est un moteur reluctance polarisé, il superpose le principe de fonctionnement des moteurs à aimant permanent et à reluctance variable et combine leurs avantages.

Le rotor est constitué de deux (2) disques dentés décalés mécaniquement. Entre ces deux (2) disques, est inséré un aimant permanent.

Deux modes de fonctionnement peuvent être envisagés selon la valeur

de temps T : Caractéristiques principales

- hors tension, le rotor est maintenu en position

- bonne précision de la position du rotor
- grande vitesse de rotation.

## II. 2. 1. 3. Comparaison des 3 types de moteurs :

<b>Comparaison des performances des 3 types de moteurs pas à pas</b>			
<b>Types de moteurs</b>	Moteur a aimant permanent	Moteur a reluctance variable	Moteurhybride
<b>Résolution (nbr de pas/tour)</b>	Moyenne	Bonne	Elevé
<b>Couple moteur</b>	Elevé	faible	Elevé
<b>Sens de rotation</b>	Il dépend :-du sensdu courant pour le moteur bipolaire -de l'ordred'alimentation des bobines	Il dépend : uniquement de l'ordre d'alimentation des bobines	Il dépend : -du sens du courant - de l'ordre d'alimentation des bobines
<b>Fréquence de travaille</b>	Faible	Grande	Grande

## II. 2. 1. 4. Configuration interne des bobines des moteursLes moteurs à 5 fils :

Le moteur comporte deux (2) bobines à point centrales, on relie le point central à l'alimentation et les autres fils à l'interface de puissance.

### **Les moteurs à 6 fils :**

Avec le moteur à 6fils, on a le choix d'une commande bipolaire en ignorant simplement les connexions centrales, ou une commande unipolaire en reliant les points centraux au (+) de l'alimentation.

### **Les moteurs à 8 fils :**

Avec celui-ci aussi on a le choix des deux commandes bipolaire ou unipolaire. Dans le premiercas on ignore les 4 fils centraux, et dans le deuxième cas on relie les 4 fils ensemble.

**Les moteurs à 4 fils :**

Ce moteur est obligé d'être alimenté soit une bobine à la fois soit les deux au même temps. Donc à tout moment le moteur est à la moitié ou à la totalité de ces bobines alimentées, ce qui a comme avantage de lui donner plus de force.

## II. 2. 2. Les pilotes des moteurs pas à pas :

Les moteurs pas à pas peuvent être commandés soit en utilisant une alimentation en tension constante qui est très utilisée pour les faibles vitesses mais dissipe une puissance assez grande, soit une alimentation en courant constant qui est répondu pour la commande des moteurs bipolaire et permet d'obtenir un grand couple.

Notre choix s'est porté sur l'utilisation d'une alimentation à courant constant pour minimiser la puissance et avoir un grand couple.

La commande d'un moteur pas à pas est assurée par plusieurs pilotes ou « drivers » de moteurs pas à pas et parmi ces drivers, on trouve le A4983 celui qu'on a choisit pour notre conception.

### 1. Commande des moteurs pas à pas avec A4983

Le support de commande de moteur pas à pas A4983 est une carte de platine pour Allegro facile à utiliser.

Le conducteur dispose de courant réglable limitant et cinq résolutions différentes de micro-pas. Il fonctionne 8-35 V et peut fournir jusqu'à 2 A par bobine.

Il permet de contrôler un moteur pas à pas bipolaire jusqu'à 2 A comme courant de sortie par bobine

Voici quelques-unes des principales caractéristiques du conducteur:

- 1) Étape simple et l'interface de commande de direction
- 2) Cinq résolutions de pas différentes: pas entier, demi-pas, quart-pas, huitième-pas, et seizième-pas
- 3) Contrôle de courant réglable vous permet de définir le courant de sortie maximale avec un potentiomètre, ce qui vous permet d'utiliser des tensions supérieures à la tension nominale de votre moteur pas à pas pour obtenir des taux plus élevés d'étape.
- 4) Le contrôle de découpage intelligent qui sélectionne automatiquement le mode de décroissance du courant.
- 5) Surchauffe arrêt thermique et la protection de croisement courant.

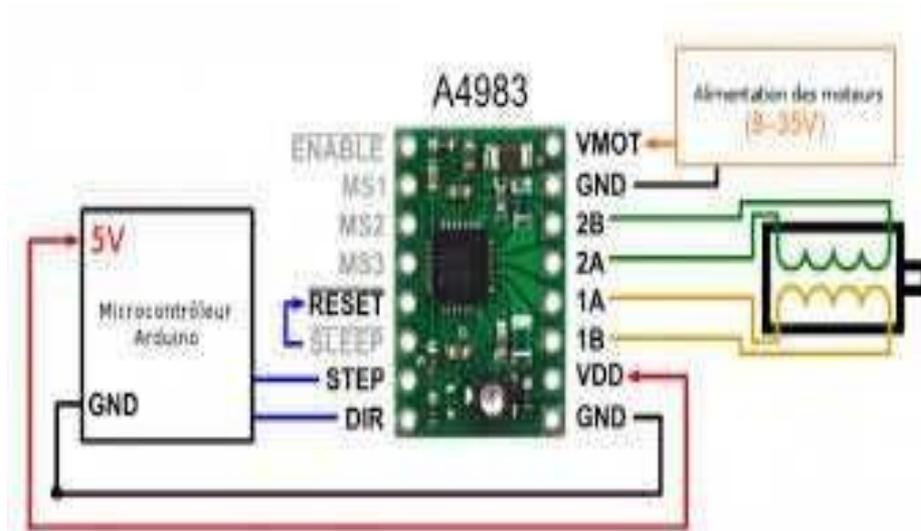


Figure II.4 : A4983

#### a) Les connexions d'alimentation

Le pilote nécessite une tension d'alimentation logique (3 à 5,5 V) connectée entre les broches VDD et GND et une tension d'alimentation du moteur de (8-35 V) devant être connectée aux bornes VMOT et GND. Ces fournitures devraient avoir des condensateurs de découplage appropriés à proximité de la carte, et ils devraient être capables de fournir les courants attendus (pics jusqu'à 4 A pour l'alimentation du moteur).

Les entrées de sélection (MS1, MS2, MS3) permettent la sélection des cinq étapes de résolutions selon le tableau ci-dessous. MS2 et MS3 ont 100kΩ de résistance interne, mais MS1 n'a pas, donc il doit être connecté à l'extérieur. Pour que les modes de micro pas fonctionnent correctement, la limite de courant doit être réglée assez bas de sorte que la limitation de courant se fiance. Sinon, les niveaux actuels intermédiaires ne seront pas correctement entretenus, et les moteurs fonctionneront effectivement dans un mode pas entier.

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

## B) Les entrées de commande

Chaque impulsion à l'entrée d'étape correspond à un micro-pas du moteur pas à pas dans la direction sélectionnée par l'axe DIR. Notons que les STEP et DIR broches ne sont pas tirés à toute tension particulière en interne, de sorte qu'on ne peut pas laisser une de ces broches flottantes dans notre application. Si on a besoin juste d'une rotation dans une seule direction, on peut attacher la broche DIR directement à VCC ou GND.

### 2. Commande des moteurs pas à pas avec L297 :

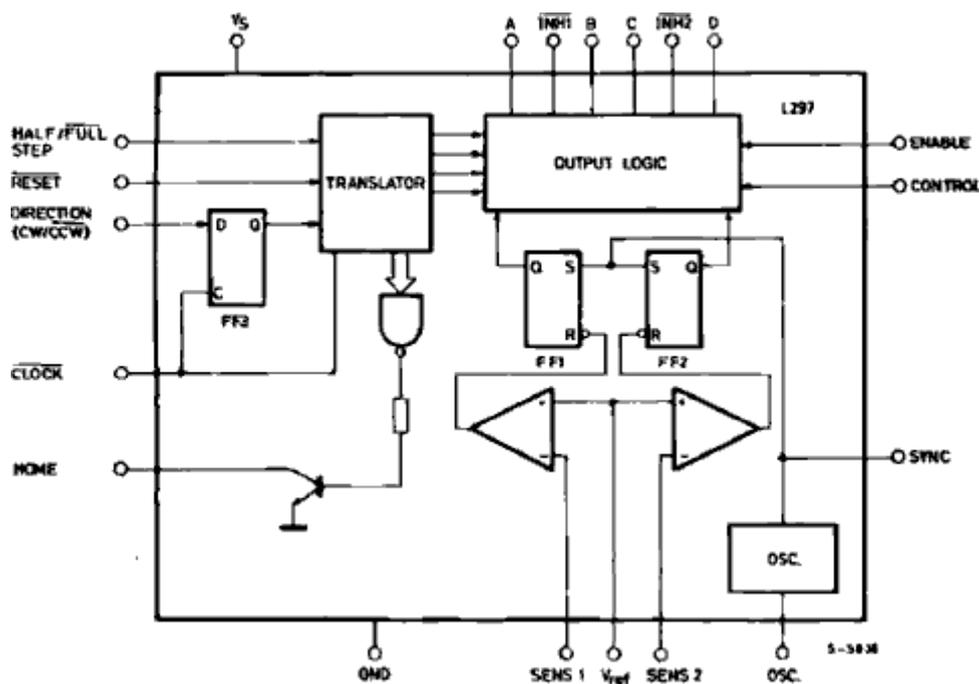
Le L297 est un circuit qui contient toute la circuiterie de commande et de contrôle de moteurs pas à pas unipolaire et bipolaire.

Utilisé souvent avec un driver double pont tel que le L298, l'ensemble forme une interface idéale pour le contrôle d'un moteur pas à pas par microprocesseur ou microcontrôleur.

Il est possible de contrôler un moteur pas à pas avec un L297 en lui associant quatre transistors darlington ou mosfet

#### a) Constitution du L297 :

Le cœur du L297 est le block appelé "translator" sur le schéma-block. C'est lui qui génère les séquences appropriées pour l'alimentation des phases en mode demi-pas ou pas entier. .

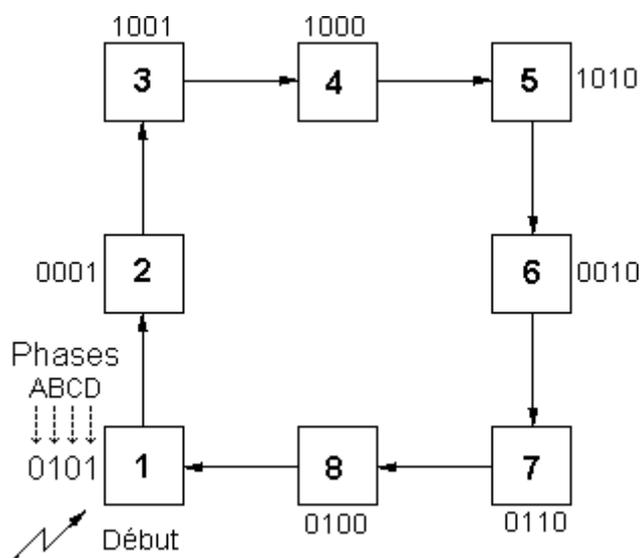


**Figure II. 6. :** schéma Blok du L297

Ce block est contrôlé par deux entrées de mode, le mode direction (CW / CCW) et le mode pas / demi-pas (HALF/ FULL) et l'entrée clock dont les impulsions d'horloge qu'on lui applique font évoluer le séquenceur vers le prochain pas.

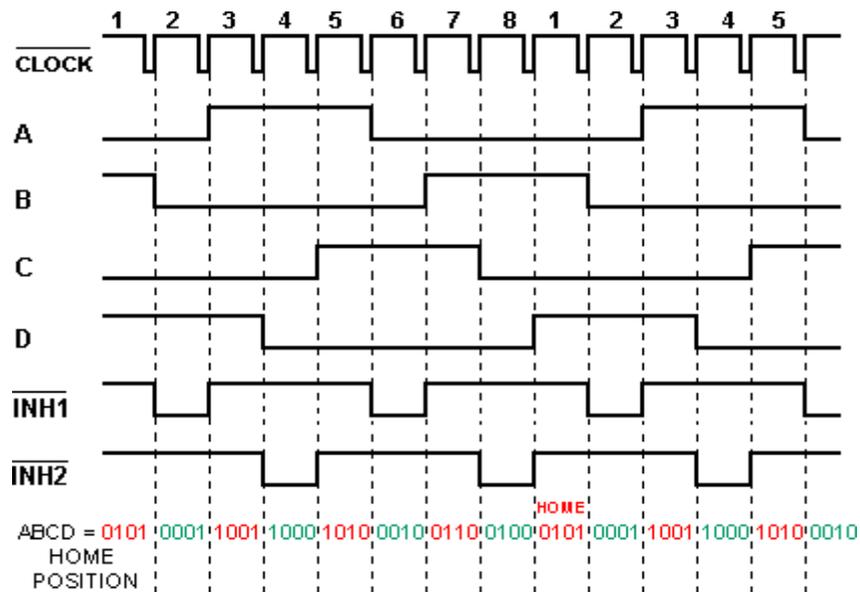
Quatre sorties sont fournies par le séquenceur et attaquent le block logique qui contient les fonctions "inhibe" et "chopper".

La structure interne du séquenceur consiste en un compteur 3 bits (plus une logique combinatoire) qui génère les huit étapes de base (en code gray) comme nous le voyons sur la figure suivante.

**Figure II. 7. :** schéma de code gray

Sur ce diagramme nous sommes en mode demi-pas (Half) et en sens horaire (CW). Les trois séquences de commande possibles sont extraites directement de cette séquence principale.

Le chronogramme des sorties en mode demi pas est visible ci-après.



Nous remarquons deux signaux supplémentaires, INH1 et INH2, générés dans cette séquence. Leurs usages sont expliqués un peu plus en avant dans le texte.

Le circuit de commande de moteur pas à pas L297 est initialement prévu pour être utilisé avec un driver L298 ou L293E.

Il reçoit les signaux de contrôle en provenance d'un système à base de microprocesseur et génère tous les signaux nécessaires pour l'étage de puissance.

Ce circuit inclut deux régulations à découpage type PWM (modulation de largeur d'impulsion) pour réguler le courant des enroulements du moteur. Avec un étage de puissance adapté le L297 commande un moteur pas à pas bipolaire à aimant permanent, un moteur unipolaire à quatre phases à aimant permanent ou un moteur à reluctance variable à quatre phases. De plus il est capable de gérer le mode demi pas et pas complet. Le L297A est une version spéciale pour le positionnement des têtes de lecteur de disquette. Le circuit intégré L298N est un étage de puissance (driver) pour les moteurs à courant continu.

Toutefois, le L297 est utilisable avec toute sorte d'étage de puissance et peut s'affranchir de la commande des composants discrets tels que des transistors de puissance ; il fournit jusqu'à 20 mA sur ces sorties. Le schéma suivant représente une configuration typique du circuit de commande de moteur pas à pas L297 et le double pont L298 pour former une interface avec un microprocesseur.

## II. 2.3. Les capteurs :

## II. 2. 3. 1. Définition des capteurs :

Un capteur est un organe de prélèvement d'information qui élabore à partir d'une grandeur physique, une autre grandeur physique de nature différente (très souvent électrique). Cette grandeur représentative de la grandeur prélevée est utilisable à des fins de mesure ou de commande.

## II. 2. 3. 2. Les différents types de capteurs :

### a) Capteur actif :

Fonctionnant en générateur, un capteur actif est généralement fondé dans son principe sur un effet physique qui assure la conversion en énergie électrique de la forme d'énergie propre à la grandeur physique à prélever, énergie thermique, mécanique ou de rayonnement. Les plus classiques sont :

- Effet thermoélectrique : Un circuit forme de deux conducteurs de nature chimique différente, dont les jonctions sont à des températures  $T_1$  et  $T_2$ , est le siège d'une force électromotrice  $e(T_1, T_2)$ .
- Effet piézo-électrique : L'application d'une contrainte mécanique a certains matériaux dits piézo-électrique (le quartz par exemple) entraîne l'apparition d'une déformation et d'une même.

Charge électrique de signe différent sur les faces opposées.

- Effet d'induction électromagnétique : La variation du flux d'induction magnétique dans un circuit électrique induit une tension électrique.
- Effet photo-électrique : La libération de charges électriques dans la matière sous l'influence d'un rayonnement lumineux ou plus généralement d'une onde électromagnétique dont la longueur d'onde est inférieure à un seuil caractéristique du matériau.
- Effet Hall : Un champ  $B$  crée dans le matériau un champ électrique  $E$  dans une direction perpendiculaire.
- Effet photo voltaïque : Des électrons et des trous sont libérés au voisinage d'une jonction PN illuminée, leur déplacement modifie la tension a ses bornes.

### b) Capteur passif :

Il s'agit généralement d'impédance dont l'un des paramètres déterminants est sensible à la grandeur mesurée. La variation d'impédance résulte :

- Soit d'une variation de dimension du capteur, c'est le principe de fonctionnement d'un grand nombre de capteurs de position, potentiomètre, inductance à noyaux mobile, condensateur à armature mobile.

- Soit d'une déformation résultant de force ou de grandeur s'y ramenant, pression accélération (armature de condensateur soumise à une différence de pression, jauge d'extensomètre liée à une structure déformable).

L'impédance d'un capteur passif et ses variations ne sont mesurables qu'en intégrant le capteur dans un circuit électrique, couplé avec une alimentation qui représente son conditionneur.

### II. 2. 3. 3. Quelques exemples sur les capteurs :

Les Capteurs de Lumière, les Capteurs de Force, les Capteurs de Son, les Capteurs de Vitesse, les Capteurs Météorologiques, et les Capteurs de Position.

Afin d'éviter les dépassements de la machine et avoir plus de sûreté on va utiliser des contacteurs de fin de course, qui peuvent aussi être appelés "Détecteur de position" ou "Interrupteur de fin de course". Ils coupent ou établissent un circuit lorsqu'ils sont actionnés par un mobile.

La détection s'effectue par contact d'un objet extérieur sur le levier ou un galet. Ce capteur peut prendre alors deux états :

- Enfoncé (en logique positive l'interrupteur est fermé).
- Relâché (en logique positive l'interrupteur est ouvert).

## II. 2. 4. Les microcontrôleurs :

### II. 2. 4. 1. Définition :

Ce sont des circuits programmables capable d'exécuter un programme et qui possède des circuits d'interface intégrés avec le monde extérieur.

Les microcontrôleurs sont apparus quand les technologies d'intégrations ont suffisamment progressées et quand les applications domestiques ou industrielles avaient besoin de systèmes « intelligents » ou tout au moins programmables.

Les microcontrôleurs se caractérisent par un plus haut degré d'intégration, une plus faible consommation électrique, une vitesse de fonctionnement plus faible (de quelques mégahertz jusqu'à plus d'un gigahertz<sup>1</sup>) et un coût réduit par rapport aux microprocesseurs polyvalents utilisés dans les ordinateurs personnels.

Par rapport à des systèmes électroniques à base de microprocesseurs et autres composants séparés, les microcontrôleurs permettent de diminuer la taille, la consommation électrique et le coût des produits. Ils ont ainsi permis de démocratiser l'utilisation de l'informatique dans un grand nombre de produits et de procédés.

Les microcontrôleurs sont fréquemment utilisés dans les systèmes embarqués, comme les contrôleurs des moteurs automobiles, les télécommandes, les appareils de bureau, l'électroménager, les jouets, la téléphonie mobile, etc.

Les microcontrôleurs améliorent l'intégration et le coût (lié à la conception et à la

réalisation) d'un système à base de microprocesseur en rassemblant ces éléments essentiels dans un seul circuit intégré. Un microcontrôleur est donc un composant autonome, capable d'exécuter le programme contenu dans sa mémoire morte dès qu'il est mis sous tension. Selon les modèles et les conditions de fonctionnement, les microcontrôleurs peuvent avoir besoin de quelques composants externes (quartz, quelques condensateurs, etc), mais ceci reste très limité.

#### II. 2. 4. 2. Composants intégré :

Un microcontrôleur se compose de :

- un processeur (CPU), avec une largeur du chemin de données allant de 4 bits pour les modèles les plus basiques à 32 ou 64 bits pour les modèles les plus évolués ;
- de la mémoire vive (RAM) pour stocker les données et variables ;
- de la mémoire morte (ROM) pour stocker le programme. Différentes technologies peuvent être employées : EPROM, EEPROM, mémoire flash (la plus récente) ;
- souvent un oscillateur pour le cadencement. Il peut être réalisé avec un quartz, un circuit RC ou encore une PLL ;
- des périphériques, capables d'effectuer des tâches spécifiques. On peut mentionner entre autres :
  - les convertisseurs analogiques-numériques (CAN) (donnent un nombre binaire à partir d'une tension électrique),
  - les convertisseurs numériques-analogiques (CNA) (effectuent l'opération inverse),
  - les générateurs de signaux à modulation de largeur d'impulsion (MLI, ou en anglais, PWM pour *Pulse Width Modulation*),
  - les timers/compteurs (compteurs d'impulsions d'horloge interne ou d'événements externes),
  - les chiens de garde (watchdog),
  - les comparateurs (comparent deux tensions électriques),
  - les contrôleurs de bus de communication (UART, I<sup>2</sup>C, SSP, CAN, FlexRay, USB, Ethernet, etc.).



**FIGURE II. 9. : pic microcontrôleur**

Ces PIC intègrent des quantités de composants très différentes.

Le fonctionnement des périphériques peut être paramétré et commandé par le programme et/ou les entrées-sorties. Les périphériques peuvent générer une interruption qui contraint le processeur à quitter le programme en cours pour effectuer une routine de traitement de l'interruption, lorsque l'événement qui la déclenche survient.

Les microcontrôleurs peuvent généralement se placer dans un état de sommeil, dans lequel ils présentent une très faible consommation électrique. Un signal envoyé par l'un de leurs périphériques (timer, broche d'entrée-sortie, watchdog, etc.) permet de les faire sortir de cet état de sommeil.

Certains microcontrôleurs ont un nombre très restreint de broches, si bien qu'une broche donnée peut correspondre à plusieurs périphériques internes. La fonction choisie doit alors être sélectionnée par logiciel.

Le choix des périphériques à intégrer dans un microcontrôleur est délicat. Les fabricants doivent réaliser un compromis entre des besoins contradictoires : utiliser des fréquences élevées, réduire la taille du circuit, apporter des fonctionnalités nombreuses, élaborer une architecture flexible, assurer des coûts modérés, etc.

#### II. 2. 4. 3. Les avantages des microcontrôleurs :

- Diminution de l'encombrement du matériel et du circuit imprimé.
- Simplification du tracé du circuit imprimé (plus besoin de tracer de bus !).
- Augmentation de la fiabilité du système nombre de composants connexions composants/ Supports et composant circuit imprimé.
- Intégration en technologie MOS, CMOS, ou HCMOS diminution de la consommation.
- Le microcontrôleur contribue à réduire les coûts à plusieurs niveaux: moins cher que les composants qu'il remplace Diminution des coûts de main d'oeuvre (conception et montage).
- Environnement de programmation et de simulation évolués.

#### II. 2. 4. 4. Les défauts des microcontrôleurs :

- Le microcontrôleur est souvent surdimensionné devant les besoins de l'application.
- Investissement dans les outils de développement.
- Écrire les programmes, les tester et tester leur mise en place sur le matériel qui entoure le microcontrôleur.
- Incompatibilité possible des outils de développement pour des microcontrôleurs de même marque.
- Les microcontrôleurs les plus intégrés et les moins coûteux sont ceux disposant de ROM programmables par masque.

- Fabrication uniquement en grande série >1000.

## II. 2. 4. 5. Familles de microcontrôleurs :

- la famille Atmel AT91 ;
- la famille Atmel AVR (utilisée par des cartes Arduino) ;
- le C167 de Siemens/Infineon ;
- la famille Hitachi H8 ;
- la famille Intel 8051, qui ne cesse de grandir ; de plus, certains processeurs récents utilisent un cœur 8051, qui est complété par divers périphériques (ports d'E/S, compteurs/temporisateurs, convertisseurs A/N et N/A, chien de garde, superviseur de tension, etc.) ;
- l'Intel 8085, à l'origine conçu pour être un microprocesseur, a en pratique souvent été utilisé en tant que microcontrôleur ;
- le Freescale 68HC11 ;
- la famille Freescale 68HC08 ;
- la famille Freescale 68HC12 ;
- la famille FreescaleQorivva MPC5XXX ;
- la famille des PIC de Microchip ;
- la famille des dsPIC de Microchip ;
- la famille des ST6, ST7, STM8, ST10, STR7, STR9, STM32 de STMicroelectronics ;
- la famille ADuC d'AnalogDevices ;
- la famille PICBASIC de ComfileTechnology ;
- la famille MSP430 de Texas Instruments ;
- la famille 8080, dont les héritiers sont le microprocesseur Zilog Z80 (désormais utilisé en tant que contrôleur dans l'embarqué) et le microcontrôleur Rabbit ;
- la famille PSoC de Cypress ;
- la famille LPC21xxARM7-TDMI de Philips ;
- la famille V800 de NEC ;
- la famille K0 de NEC.

## II. 2. 4. 6. En conclusion :

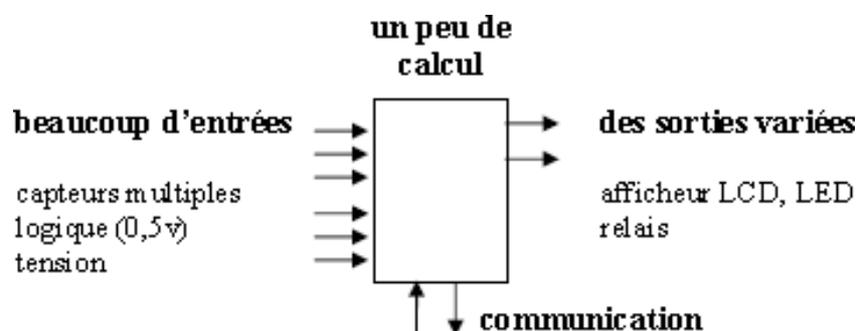


Figure II. 10. : Rôle de microcontrôleur

Le microcontrôleur présente l'avantage des  $\mu$ processeurs mais limités aux applications ne nécessitant pas trop de puissance de calcul (architecture courante 8bits)

Il existe plusieurs architectures de microcontrôleurs de 4 à 128 bits pour les applications demandant une certaine puissance de calcul (injecteurs automobile)

Dans notre conception on a utilisé un microcontrôleur de la famille ATmel AVR (utilisée par des cartes Arduino), il existe plusieurs microcontrôleurs de cette même famille parmi eux l'ATMega328 qui est utilisé pour les cartes Arduino UNO, qu'on a choisis pour notre conception et qu'on va présenter dans ce qui suit.

## II. 2. 5. Arduino UNO :

### II. 2. 5. 1. Vue d'ensemble :

Le modèle UNO de la société ARDUINO est une carte électronique dont le cœur est un microcontrôleur ATMEL de référence ATMega328. Le microcontrôleur ATMega328 est un microcontrôleur 8bits avec une vitesse d'horloge de 16Mhz de la famille AVR dont la programmation peut être réalisée en langage C. L'intérêt principal des cartes ARDUINO (d'autres modèles existant) est leur facilité de mise en œuvre. ARDUINO fournit un environnement de développement s'appuyant sur des outils open source. Le chargement du programme dans la mémoire du microcontrôleur se fait de façon très simple par port USB. En outre, des bibliothèques de fonctions "clé en main" sont également fournies pour l'exploitation d'entrées- sorties courantes : gestion des E/S TOR, gestion des convertisseurs ADC, génération de signaux PWM, exploitation de bus TWI/I2C, exploitation de servomoteurs...

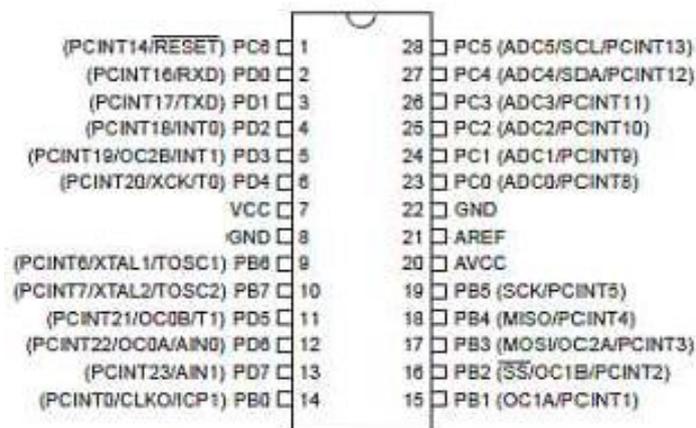


Figure II. 11. :Brochage de l'ATMega328

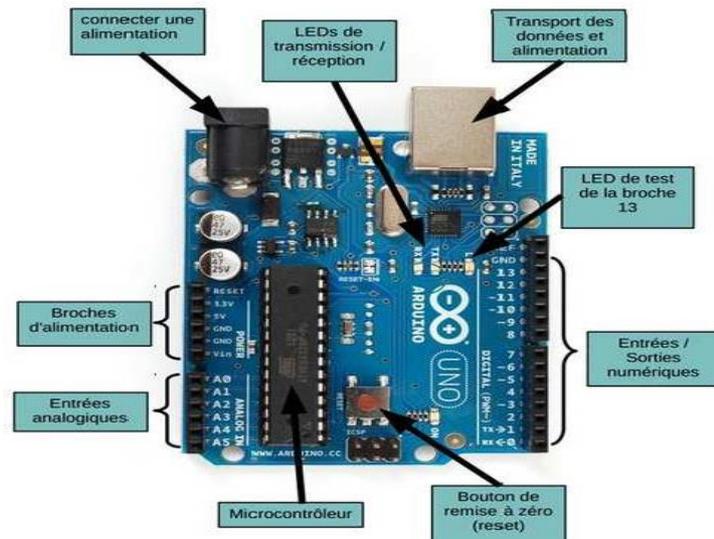


Figure II. 11. :Carte Arduino UNO

### II. 2. 5. 2. Pourquoi Arduino :

- **Pas cher** : les cartes Arduino sont relativement peu coûteuses comparativement aux
- **multiplateforme** : le logiciel Arduino, écrit en Java, tourne sous les systèmes d'exploitation Windows, Macintosh et Linux. La plupart des systèmes à microcontrôleurs sont limités à Windows.
- **un environnement de programmation clair et simple** : l'environnement de programmation Arduino (le logiciel Arduino) est facile à utiliser pour les débutants, tout en étant assez flexible pour que les utilisateurs avancés puissent en tirer profit également.
- **logiciel Open Source et extensible** : le logiciel Arduino et le langage Arduino sont publiés sous licence open source, disponible pour être complété par des programmes expérimentés.
- **matériel Open source et extensible** : les cartes Arduino sont basées sur les microcontrôleurs Atmel ATMEGA8, ATMEGA168, ATMEGA 328, etc... Les schémas des modules sont publiés sous une licence Creative Commons, et les concepteurs de circuits expérimentés peuvent réaliser leur propre version des cartes Arduino, en les complétant et en les améliorant. Même les utilisateurs relativement inexpérimentés peuvent fabriquer la version sur plaque d'essai de la carte Arduino, dans le but de comprendre comment elle fonctionne.

### II. 2. 5. 3. Alimentation :

Arduino UNO peut être alimentée via la connexion USB ou avec une alimentation externe. La source d'alimentation est automatiquement sélectionnée. La plage de tension recommandée est de 7 à 12 volts, les br

suivantes :

**VIN** : la tension d'entrée de la carte arduino, quand c'est une source d'alimentation externe (par opposition à 5 volts de la connexion USB ou tout autre source d'énergie réglementés).

**5V** : l'alimentation régulée utilisée pour alimenter le microcontrôleur et les autres composants sur la carte. Cela peut provenir soit de VIN via un régulateur de bord, ou par l'intermédiaire d'un port USB ou d'une autre alimentation 5V régulée.

**3.3V** : une alimentation de 3.3 Volts générée par un circuit intégrée FTDI, avec un courant maximal de 50 mA.

**GND** : broche de masse.

#### II. 2. 5. 4. La mémoire :

Le ATmega328 a 32Ko de mémoire flash pour stocker le code dont 0.5 Ko utilisé par le bootloader, 2 Ko de SRAM et 1 Ko de mémoire EEPROM qui peuvent être lues et écrites avec la bibliothèque EEPROM .

#### II. 2. 5. 5. Les entrées et sorties :

**Broche numérique** : Chacune des 14 broches numériques de la carte UNO (numérotées de 0 à 13) peut être utilisée soit comme une entrée numérique, soit comme une sortie numérique, en utilisant les instructions `pinMode()`, `digitalWrite()` et `digitalRead()` du langage Arduino. Ces broches fonctionnent en 5V. Chaque broche peut fournir ou recevoir un maximum de 40mA d'intensité et dispose d'une résistance interne "rappel au plus" (pull-up) (déconnectée par défaut) de 20-50 KOhms. Cette résistance interne s'active sur une broche en entrée à l'aide de l'instruction `digitalWrite(broche, HIGH)`.

**Broche analogique** : La carte Uno dispose de 6 entrées analogiques (numérotées de 0 à 5), chacune pouvant fournir une mesure d'une résolution de 10 bits (c-à-d sur 1024 niveaux soit de 0 à 1023) à l'aide de la très utile fonction `analogRead()` du langage Arduino. Par défaut, ces broches mesurent entre le 0V (valeur 0) et le 5V (valeur 1023),

De plus, certaines broches ont des fonctions spécialisées :

- Communication Série: Port série (USART) Broches 0 (RX) et 1 (TX). Utilisées pour recevoir (RX) et transmettre (TX) les données série.
- Interruptions Externes: Broches 2 et 3. Ces broches peuvent être configurées pour déclencher une interruption sur une valeur basse, sur un front montant ou descendant, ou sur un changement de valeur. Voir l'instruction `attachInterrupt()` pour plus de détails .
- Timers/Counters: Timer0 et Timer2 (comptage 8 bits), Timer1 (comptage 16bits), on peut les utilisés comme temporisateur ou compteur.
- WatchdogTimer : programmable.
- Impulsion PWM (largeur d'impulsion modulée): Broches 3, 5, 6, 9, 10, et 11. Fournissent une impulsion PWM 8-bits à l'aide de l'instruction `analogWrite()`.
- SPI (Interface Série Périphérique): Broches 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Ces broches supportent la communication SPI (Interface Série Périphérique) disponible avec la librairie pour communication SPI. Les broches SPI sont également

connectées sur le connecteur ICSP qui est mécaniquement compatible avec les cartes Méga.

- I2C: Broches 4 (SDA) et 5 (SCL). Supportent les communications de protocole I2C (ou interface TWI (TwoWire Interface - Interface "2 fils"), disponible en utilisant la librairie Wire/I2C (ou TWI - Two-Wire interface - interface "2 fils") .
- LED: Broche 13. Il y a une LED incluse dans la carte connectée à la broche 13. Lorsque la broche est au niveau HAUT, la LED est allumée, lorsque la broche est au niveau BAS, la LED est éteinte.
- Reset : bouton de réinitialisation du microcontrôleur.

#### II. 2. 5. 6. Communication :

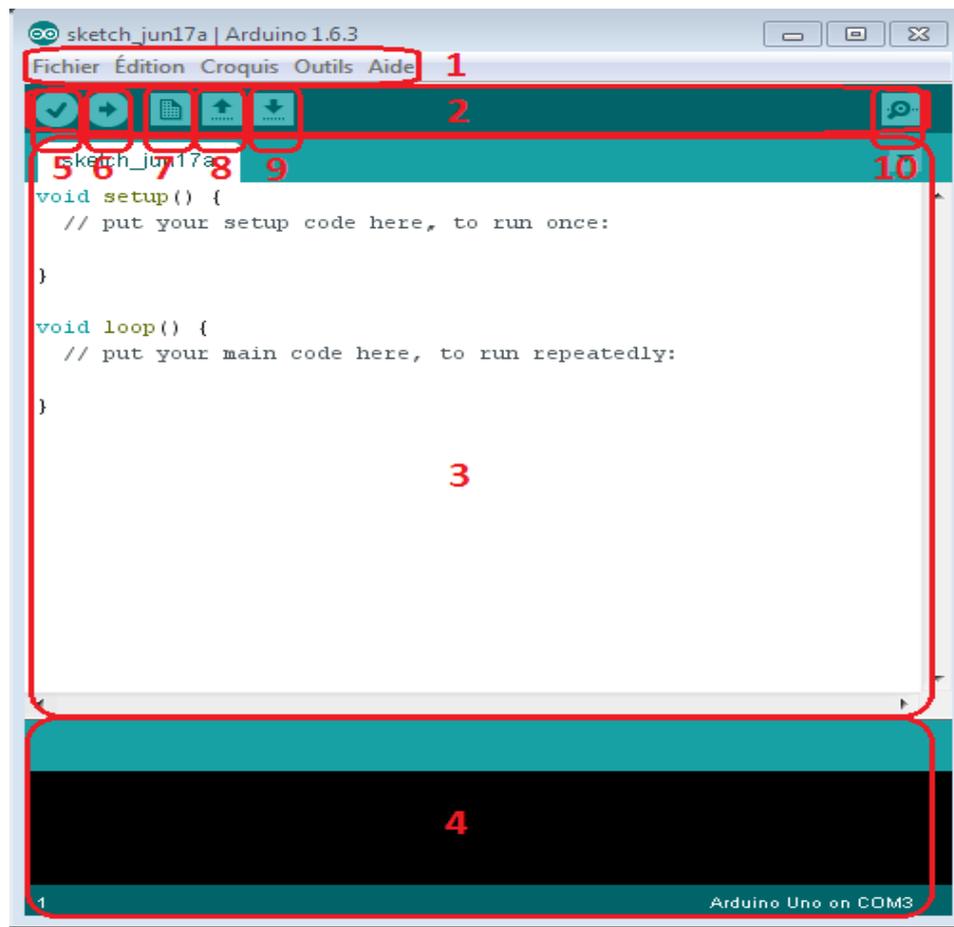
Arduino UNO peut communiquer avec un ordinateur, ou un autre arduino, ou d'autres microcontrôleurs, L'ATmega 328 dispose d'une UART (Universal Asynchronous Receiver Transmitter) pour communication série.

Le logiciel Arduino inclut une fenêtre terminal série (ou moniteur série) sur l'ordinateur et qui permet d'envoyer des textes simples depuis et vers la carte Arduino. Les LEDs RX et TX sur la carte clignotent lorsque les données sont transmises via la connexion USB vers l'ordinateur.

Le logiciel Arduino inclut la librairie Wire, qui simplifie l'utilisation du bus I2C, et la librairie pour communication SPI, pour utiliser la communication SPI (Interface Série Périphérique).

#### II. 2. 5. 7. Programmation :

Pour programmer la carte Aduino UNO, nous avons utilisé le logiciel Arduino C prévu à cet effet. C'est un logiciel libre et gratuit développé pour Windows, Mac et Linux. Ce logiciel présente une interface épurée et permet de développer facilement des programmes qui peuvent être transférés sur la carte en branchant celle-ci sur un port USB. Le langage utilisé par le logiciel est basé sur le C, auquel s'ajoutent des instructions spécifiques au langage Arduino. Le logiciel dispose d'une banque d'exemples très riche qui permet d'utiliser des morceaux de codes pré-écrit.



- Figure II. 12. : Présentation du logiciel Arduino 1.6.3

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

**Le cadre numéro 2** : il contient les boutons qui vont nous servir lorsque l'on va programmer la carte Arduino.

**Le cadre numéro 3** : ce bloc va contenir le programme que nous allons écrire.

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

**Bouton 5** : Ce bouton permet de vérifier le programme, il actionne un module qui cherche les erreurs dans le programme.

**Bouton 6** : Charge le programme dans la carte Arduino

**Bouton 7** : Crée un nouveau fichier

**Bouton 8** : Ouvre un fichier

**Bouton 9** : Enregistre le fichier

**Bouton 10** : Ouvre le moniteur série.

**Exemple** : : commander un moteur pas à pas pour faire 400 pas en avance :

```
#define Enable 6 // Activation du driver/pilote
#define pinStep 7 // Signal de PAS (avancement)

#define pinDir8 //
Direction void
setup(){
Serial.begin(9600);
Serial.println("Test
A4988"); pinMode(
Enable, OUTPUT );
pinMode( pinDir ,
OUTPUT );
pinMode( pinStep ,
OUTPUT );} void
loop(){
int i = 0;

digitalWrite( pinDir , HIGH); // Direction avant // Avance de
200 pasfor( i=0; i < 400; i++){
Serial.println( i );
digitalWrite(
pinStep, LOW );
digitalWrite(
pinStep, HIGH );
delay (1); }
Serial.println("Axe bloqué +
attendre 5 sec");delay( 5000 );
Serial.println("Debloce axe");

digitalWrite( Enable, HIGH); //
logique inverséedelay(5000);
digitalWrite( Enable, LOW); //
logique inverséeSerial.println("Fin
de programme");
while( true );

}
```

## II. 3. Commande PWM (Pulse Width Modulation) :

### II. 3. 1. Définition :

PWM signifie Pulse Width Modulation, modulation par largeur d'impulsion.

La commande par PWM permet de produire simplement une variation de la valeur moyenne d'une tension.

La commande "PWM" est un signal numérique où la tension peut prendre deux valeurs seulement.

Dans certains cas très spécifiques (onduleurs à MLI par exemple) on fabrique un troisième niveau en inversant la tension du niveau haut.

Le signal de cette commande est carré et Le niveau bas correspond généralement à 0 Volt. La période est notée  $T$  ; la durée de l'impulsion (pour laquelle la tension est celle de l'état haut) est appelée  $t_h$ .

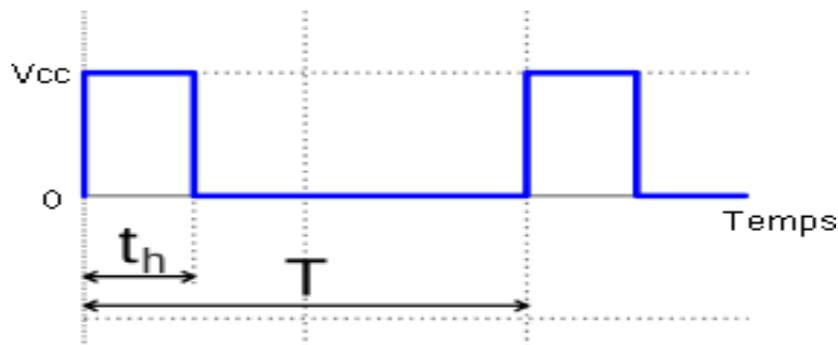


FIGURE II. 13. : le signal de VCC par rapport au temps

Si la période change, le signal n'est plus vraiment périodique au sens strict.

On appelle alors  $T$  la pseudo période.

### II. 3. 2. INTÉRÊT :

Le principal intérêt de la technique PWM est de limiter la chauffe des composants électroniques.

Par exemple : une lampe de 20 Watts allumée au maximum consomme 20 W. Si par une commande de gradation elle est allumée au quart de sa puissance, elle consomme 5 W. Le composant analogique devrait alors dissiper 15 W, ce qui implique un énorme radiateur. En PWM, la puissance fournie est soit maximale, soit nulle. Lorsqu'elle est maximale, pendant un quart du temps par exemple, il n'y a pas besoin de dissiper de puissance résiduelle. Lorsqu'elle est nulle, il n'y a pas besoin de dissiper non plus de puissance, car elle n'est pas fournie du tout.

Un autre intérêt du PWM est que la tension appliquée au moteur pendant  $t_h$  est  $V_{cc}$ . Celle-ci est suffisante pour vaincre les frottements et faire tourner le moteur. La tension moyenne appliquée au moteur est proportionnelle au rapport cyclique, ce qui permet d'avoir des consignes de vitesse faibles :

$$V_{MOY} = \frac{t_h \times V_{cc}}{T}$$

### II. 3. 3. FRÉQUENCE :

La commande d'actionneurs de puissance par PWM est très liée à la notion de fréquence. Pour que l'impression d'une valeur moyenne constante d'allumage apparaisse, il faut que l'alternance d'allumage/extinction soit suffisamment rapide pour qu'elle ne se remarque pas.

Selon les utilisations la fréquence du PWM va de 100 Hz (100 cycles par seconde) à 200 kHz.

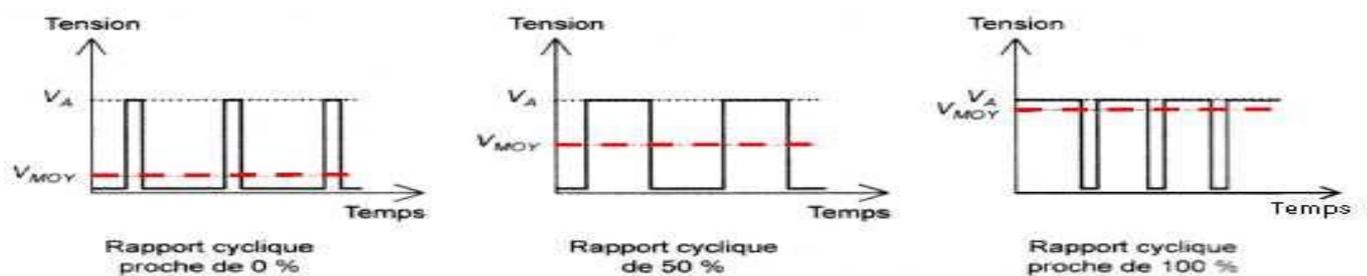
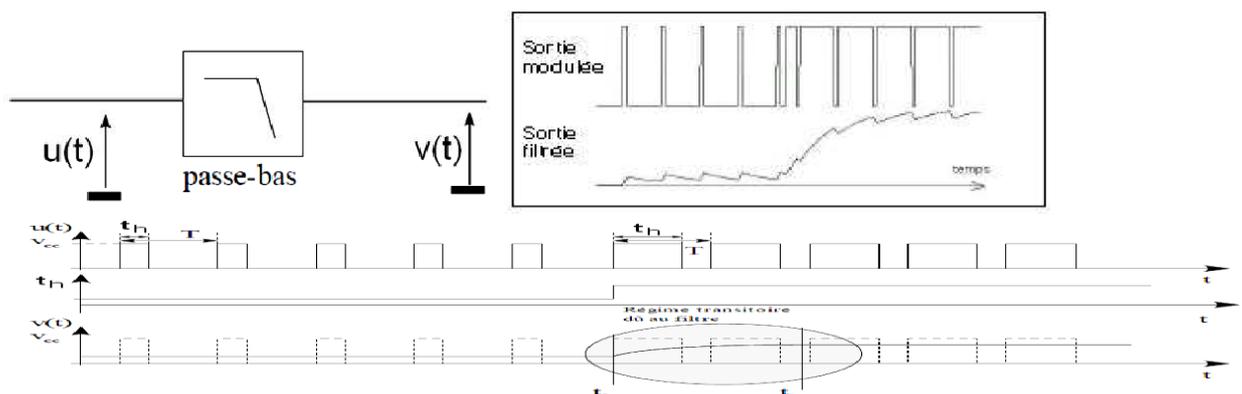


FIGURE II. 14. : Changements de tension par rapport au temps

#### Remarques :

Même si le signal n'est nullement une tension analogique, il peut le devenir si on lui fait subir un filtrage de type passe-bas.

Filtrer ce signal carré permet d'obtenir la tension moyenne. La fréquence de coupure du filtre doit être plus faible que la fréquence du signal carré. Autrement dit la



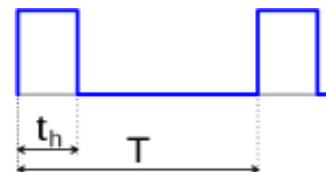
fréquence porteuse doit être significativement supérieure à la fréquence de coupure du filtre, pour que l'ondulation résiduelle soit faible.

### II. 3. 4. . RAPPORT CYCLIQUE :

On appelle rapport cyclique le rapport :  
 $\alpha = 100 \times t_h / t$  exprimé en pourcentage.

Si  $t_h = 0$  alors  $\alpha = 0\%$  et la tension moyenne de sortie est nulle.

Si  $t_h = T$  alors  $\alpha = 100\%$  et la tension moyenne de sortie est égale à  $V_{cc}$ .



### II. 4. Conclusion :

Dans ce chapitre nous avons présentés les composant électronique utilisés dans notre projet à savoir : le microcontrôleur, LaunchPad MSP430, le driver de commande de moteurs leA4988, l'arduino, l'alimentation et le moteur pas à pas type Nema17.

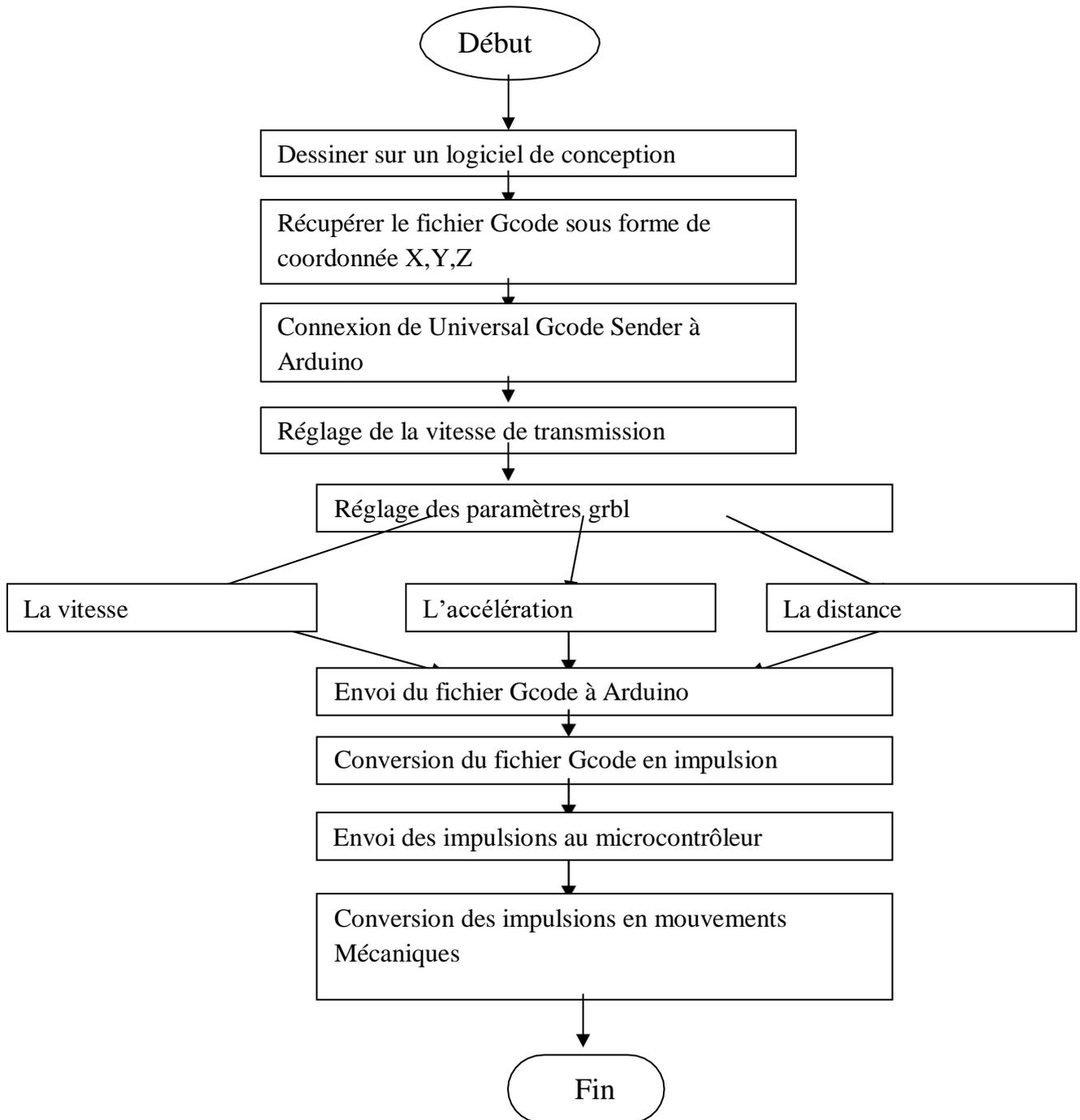
Nous avons réussi à contrôler les moteurs pas à pas par la commande PWM et GPIO. C'est notre premier pas vers la commande de notre machine CNC.





### III. 1. Organigramme

Il représente la partie programmation d'une machine CNC



## III. 2. G-code :

### III. 2. 1. Historique du langage G-code :

Le G-code est un langage de programmation, était développé par l'EIA au début des années 1960, et finalement normalisé par l'ISO en février 1980 sous la référence RS274D/ (ISO 6983).

Compte tenu de l'absence de développement ultérieur, une grande variété de configurations de machines-outils, et du peu de demande pour une réelle interopérabilité, peu de contrôleurs à commande numérique respectent ce standard. Des extensions et variantes ont été ajoutées indépendamment par divers fabricants, ce qui fait que les opérateurs doivent connaître les différents dialectes et particularité des machines qu'ils utilisent, et les systèmes de CFAO (Conception et Fabrication Assistées par Ordinateur) doivent se limiter au plus petit dénominateur commun des machines qu'ils commandent.

Beaucoup de fabricants ont essayé de contourner cette difficulté et rester compatible en suivant la route tracée par la firme Fanuc. Malheureusement, Fanuc n'est pas conforme à la norme RS- 274 ou à ses précédents standards, et a été lent à ajouter de nouvelles fonctionnalités et à utiliser la puissance croissante des ordinateurs. Par exemple, ils ont transformé la commande g70/g71 en g20/g21 ; ils ont utilisé des parenthèses pour les commentaires, ce qui a causé des problèmes lors de l'introduction des calculs mathématiques ; ils n'ont commencé à utiliser les nanomètres que récemment (ce qui requiert 64 bits) ; ils ont introduit les nurbs pour compenser le faible débit des blocs depuis la mémoire (au lieu de mettre en place un cache).

Depuis l'établissement de la norme ISO 6983, et les technologies évoluant rapidement, de nombreuses extensions ont été ajoutées pour tenir compte des nouveautés et des nouvelles capacités des machines-outils. Ces extensions, bien que souvent utiles chez des constructeurs différents de directeur de commande numérique, n'entrent pas dans la norme et compliquent la tâche des logiciels de fabrication assistée par ordinateur, qui doivent créer les lignes de ce langage pour un DCN particulier. Pour analyser la syntaxe des programmes ISO, des éditeurs de logiciels ont mis au point des solutions de simulation d'usinage qui ont permis d'émuler le comportement

d'un DCN sur un PC. Cela permet de mettre au point les programmes sans mobiliser la machine- outil à commande numérique. On peut citer des solutions comme Ncsimul, Roboris, Ncview, etc... A côté de l'ISO, des langages propres aux constructeurs de DNC ainsi que les interfaces de programmation conversationnelle sont destinées à simplifier la programmation.

Le principe est simple : on va écrire une suite de lignes comportant quelques instructions et commandes qui vont demander à la machine de réaliser des mouvements dans l'espace en 3 dimensions. On va pour cela utiliser un repère cartésien dont les 3 axes de références sont les axes X-Y-Z.

### III. 2. 2. Définition de G-code :

Le G-code est le langage plus utilisé pour contrôler une machine à commande numérique. Ils'agit bien d'un langage de programmation, qui nous sert donc à programmer les mouvements que la machine va effectuer, et le fichier contenant la suite d'instruction s'appelle, en toute logique, un programme. Il s'agit de simple fichier texte, humainement lisible, au même titre que du code en C, Pascal ou Basic. Il se compose d'un certain nombre de "commandes" spécifiques indiquant à la machine quel type de mouvement elle doit exécuter (droite, arc de cercle, etc....), et l'indication de coordonnées sur les axes X, Y et Z.

### III. 2. 3. Les usages de G-code

Le G-code Développé à l'origine pour des machines-outils par enlèvement de matière, il est désormais utilisé dans un domaine très vaste de la fabrication, avec des adaptations :

- ❖ Usinage par enlèvement de matière : tournage, fraisage, perçage, gravure, défonçage ;
- ❖ Découpe avec : couteau, laser, jet d'eau, plasma, flamme ou oxydation.
- ❖ Poinçonnage
- ❖ Impression 3D : par dépôt de matière, durcissement d'une résine, solidification de poudre.

## III. 3. Programmation de la machine à commande numérique

Un programme CN comporte des caractères obligatoires de début et fin. L'affectation des blocs situés entre ses caractères est dans l'ordre de les écrire et leurs numérotations n'intervient pas dans l'ordre de déroulement du programme donc il est conseillé de les numéroter dans l'ordre d'écriture [25]. On peut programmer manuellement ou assistée par un logiciel de FAO

**III. 3. 1. Structure d'un programme :** Un programme CN se compose d'une suite de blocs. Chaque bloc se compose des mots et chaque mot contient les données et les ordres l'exécution d'une opération requis. Il est divisé en 3 parties : tête de programme, corps de programme et fin de programme.

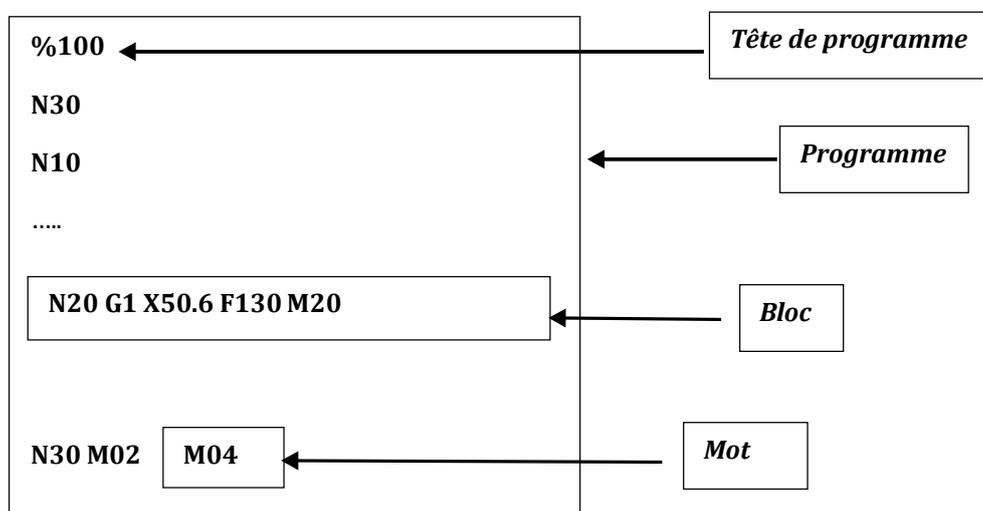
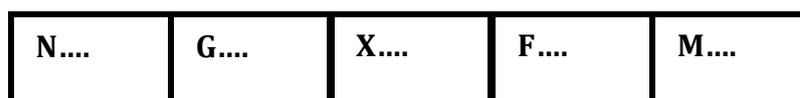


Figure III. 1. : Structure d'un programme.

### III. 3. 2. Format d'un bloc :

On définit une ligne d'instruction composée de mots codés par un bloc à transmettre au système de commande. Le format de bloc définit la syntaxe des mots de fonction et de dimension composant chaque bloc de programme.



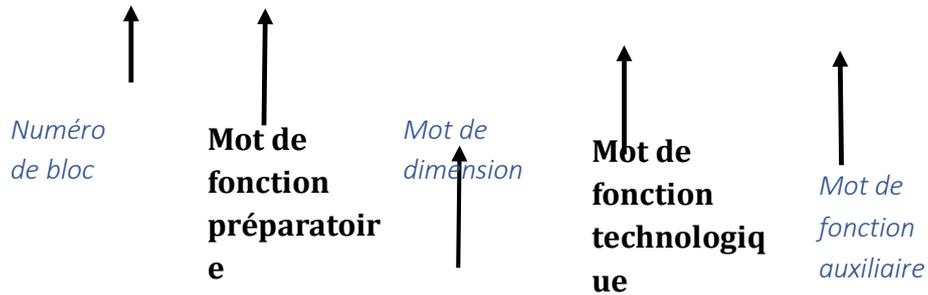


Figure III. 2. : Format d'un bloc

### III. 3. 3. Format d'un mot :

Une instruction ou un ordre à transmettre au système de commande est défini par un mot. Un mot est composé d'une lettre et d'une valeur numérique, entière ou décimale.

<b>Adresse</b> <i>(1 ou 2 lettre / 1 caractère)</i>	<b>Signe Algébrique</b> <i>(+ / -)</i>	<b>Données</b> ----- <i>(Chiffres lié à l'adresse)</i>
--	---	--

Figure III. 3. : Format d'un mot

On distingue deux types de mot :

- Mots définissant des dimensions :

X	+	5.3
---	---	-----

- Mots définissant des fonctions :

G	0	2
---	---	---

#### III. 3. 3. 3. Les différents formats des mots :

Dans un bloc, les mots peuvent avoir plusieurs fonctions, chaque fonction de mot est affectée en tant qu'adresse identifiée par un code de lettre dans le programme. Il existe différents fonction des mots, dans ce qui suit nous décrivons les fonctions de base les plus utilisés dans les programmes :

- **Fonctions préparatoires (G) :** Commande G-Code Standard, Déplacement d'un ou plusieurs axes, cycles machine.

### Les principales fonctions du G-code :

Les principales fonctions du G-code sont résumées dans le tableau III.1 suivant.

<b>G00</b>	Déplacement rapide
<b>G01</b>	Interpolation linéaire
<b>G02</b>	Interpolation circulaire (sens horaire)
<b>G03</b>	Interpolation circulaire (sens anti-horaire)
<b>G04</b>	Arrêt programme et ouverture carter (pour nettoyer) (temporisation - suivi de l'argument F ou X en secondes)
<b>G10/G11</b>	Écriture de données/Effacement de données
<b>G17</b>	Sélection du plan X-Y
<b>G18</b>	Sélection du plan X-Z
<b>G19</b>	Sélection du plan Y-Z
<b>G20</b>	Programmation en pouces
<b>G21</b>	Programmation en mm
<b>G28</b>	Retour à la position d'origine

<b>G31</b>	Saute la fonction (utilisé pour les capteurs et les mesures de longueur d'outil)
<b>G33</b>	Filetage à pas constant
<b>G34</b>	Filetage à pas variable
<b>G40</b>	Pas de compensation de rayon d'outil
<b>G41</b>	Compensation de rayon d'outil à gauche
<b>G42</b>	Compensation de rayon d'outil à droite
<b>G90</b>	Déplacements en coordonnées absolues
<b>G91</b>	Déplacements en coordonnées relatives
<b>G94/G95</b>	Déplacement en Pouces par minute/Pouce par tour
<b>G96/G97</b>	Vitesse de coupe constante (vitesse de surface constante) /Vitesse de rotation constante ou annulation de G96.
<b>M00/M01</b>	Arrêt du programme, arrêt optionnel ou avec condition
<b>M06</b>	Changement d'outil
<b>M30</b>	Fin du programme, réinitialisation, rembobinage

Tableau III. 1. : Les principales fonctions du G-code.

### III. 4. Le fichier GRBL :

Grbl : est un firmware de haute performance conçue pour contrôler le mouvement des machines praticable sur un Arduino. Il a été adapté pour être utilisé dans certaines projets comme les machine CNC et les imprimantes 3D, Le programme est écrit en C hautement optimisé utilisant chaque fonctionnalité intelligente de l'AVR-puces pour atteindre un timing précis et le fonctionnement asynchrone. Il est capable de maintenir plus de 30 kHz de la stabilité, la rapidité des impulsions de commande libres.

En 2009, Simen Svale Skogsrud a honoré la communauté open-source en écrivant et en libérant les premières versions de Grbl à tout le monde. Depuis 2011, Grbl va de l'avant comme

un projet Open-source mené par la communauté sous la direction pragmatique de Sonny Jeon Ph.D. (chamnit).

Open-source mené par la communauté sous la direction pragmatique de Sonny Jeon Ph.D. (chamnit).

Le projet a d'abord été inspiré par l'interprète Arduino Gcode par Mike Ellery.

### III. 4. 1. Schéma de Pin pour Grbl :

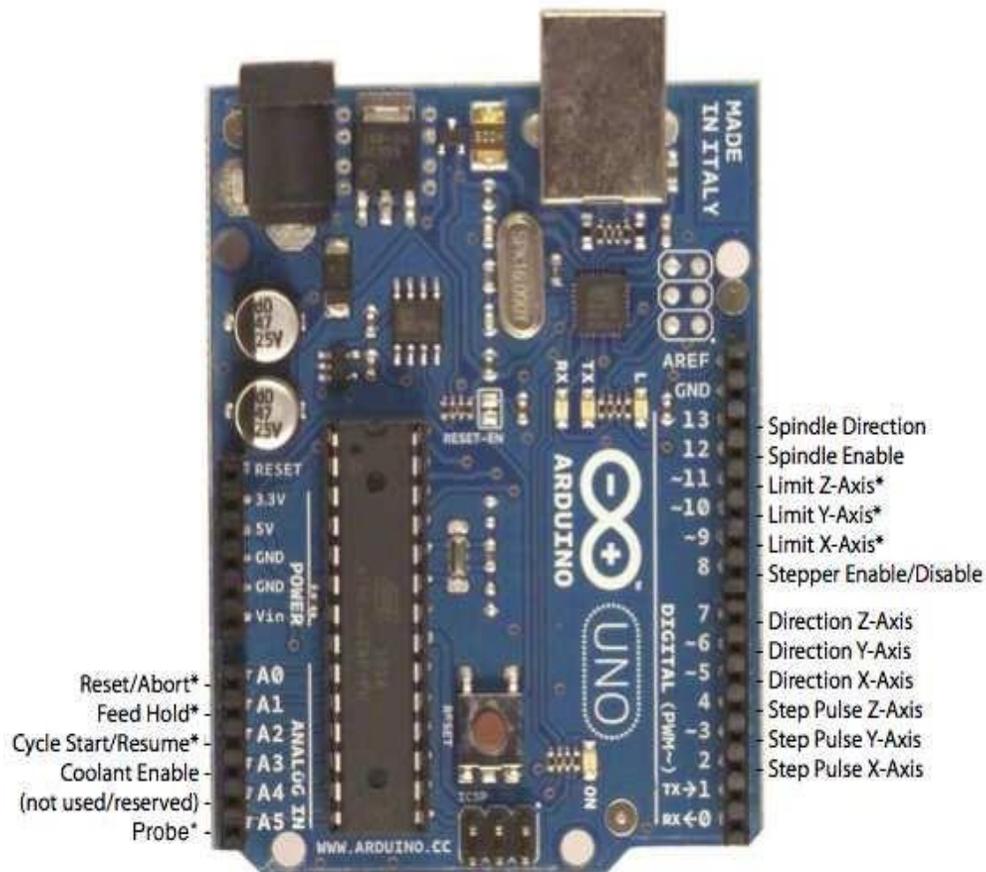


Figure III.4 : Schéma de Pin pour Grbl.

### III. 4. 2. Fonctionnement de grbl :

Après avoir flashé Grbl à l' Arduino, la connexion est assez simple. Il faut utiliser l'Arduino IDE lui-même pour se connecter à Grbl, ou D'autres programmes de port série, tels que DIVA ou PuTTY,

Avec le Xloader on peut modifier les paramètres comme suit:

- Sélectionner le fichier HEX.
- Sélectionner la configuration dans L'Arduino.
- Sélectionner le port COM correct.
- Gardez la vitesse de transmission est que ... (Il va changer pour chaque type de carte.

### III. 4. 3. Le branchement :

La broche 'step » pour les axes X, Y et Z sont broches sur les pins 2, 3 et 4. La broche "Dir» pour les axes X, Y et Z broches sur les pins 5, 6 et 7.

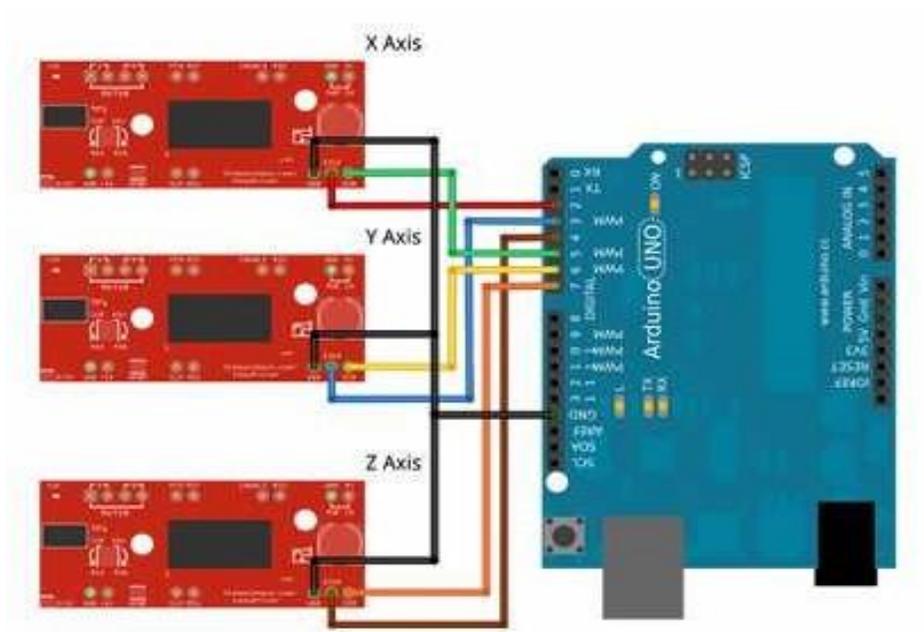


Figure III.5 : branchement des pins STEP et DIR sur arduino

### III. 4. 4. Installation GRBL dans la carte ARDUINO:

- Tout d'abord télécharger GRBL.
- Copiez le dossier décompressé dans: \Documents\Arduino\libraries
- Une fois terminé, redémarrer Arduino.
- Rendez-vous ensuite dans fichiers -> exemples -> grbl -> grblUpload, voir figure III. 6.

- Une fois le sketch ouvert, téléchargez-le sur votre carte Arduino Uno par une cliquesur le bouton qui est on cercle rouge. Voir la figure III. 7.

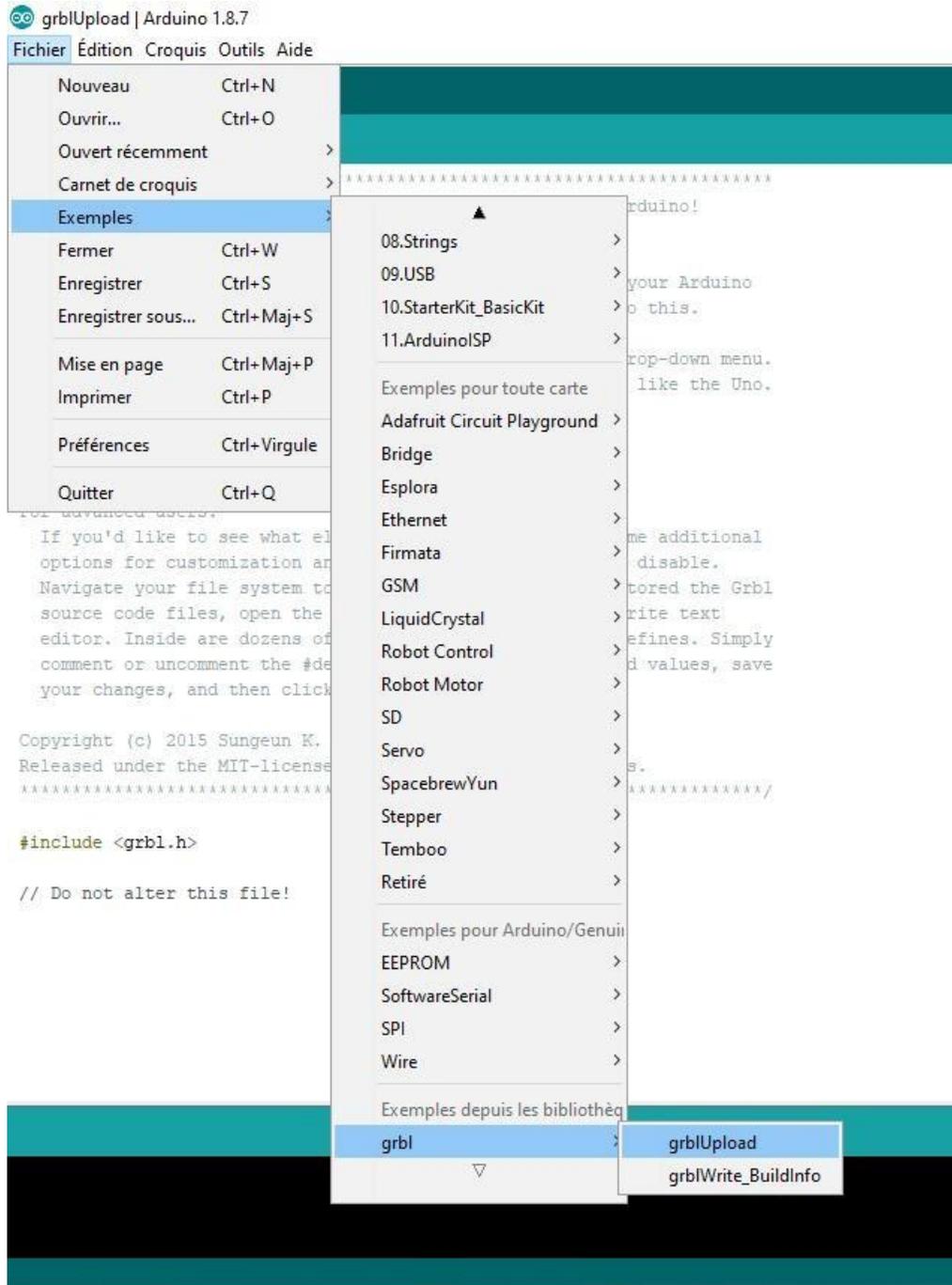


Figure III. 6. : 1ere étape de l'installation.

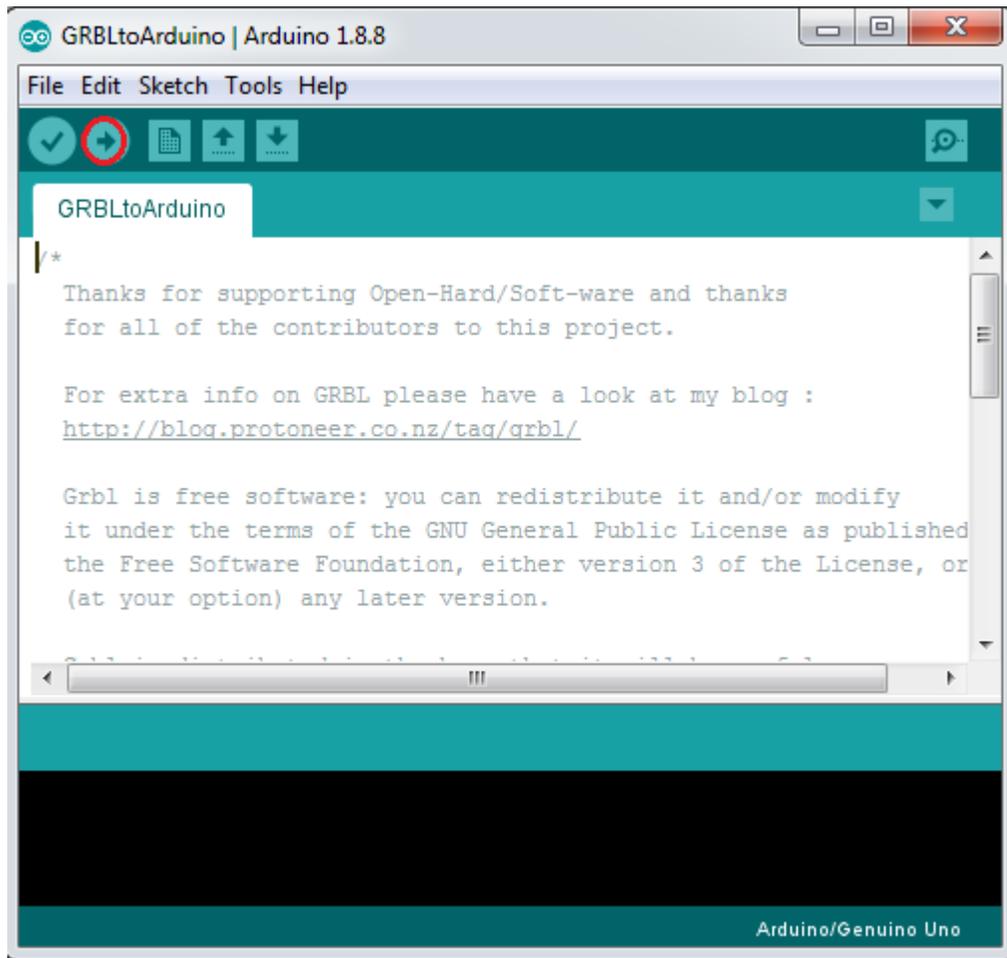


Figure III. 7. : 2eme étape de l'installation.

#### III. 4. 5. Universal G-Code Sender :

C'est logiciel qui fera l'interface avec ordinateur et qui permettra d'envoyer le G-Code à la carte Arduino qui lui-même sera interprété par GRBL.

Une fois installé, vous devrez aller dans « Settings » -> « Firmware Setting » et sélectionner « GRBL ». Ainsi le logiciel saura comment piloter la carte Arduino .

Il y'a deux modes utilisées dans ce programme soit mode manuel ou mode automatique.

- Les deux cadres en rouge représentent le mode manuel de logiciel G-code Sender. Voir figureIV.6.
- Le cadre en verre représente le mode automatique de logiciel G-code Sender. Voir figureIII. 8.

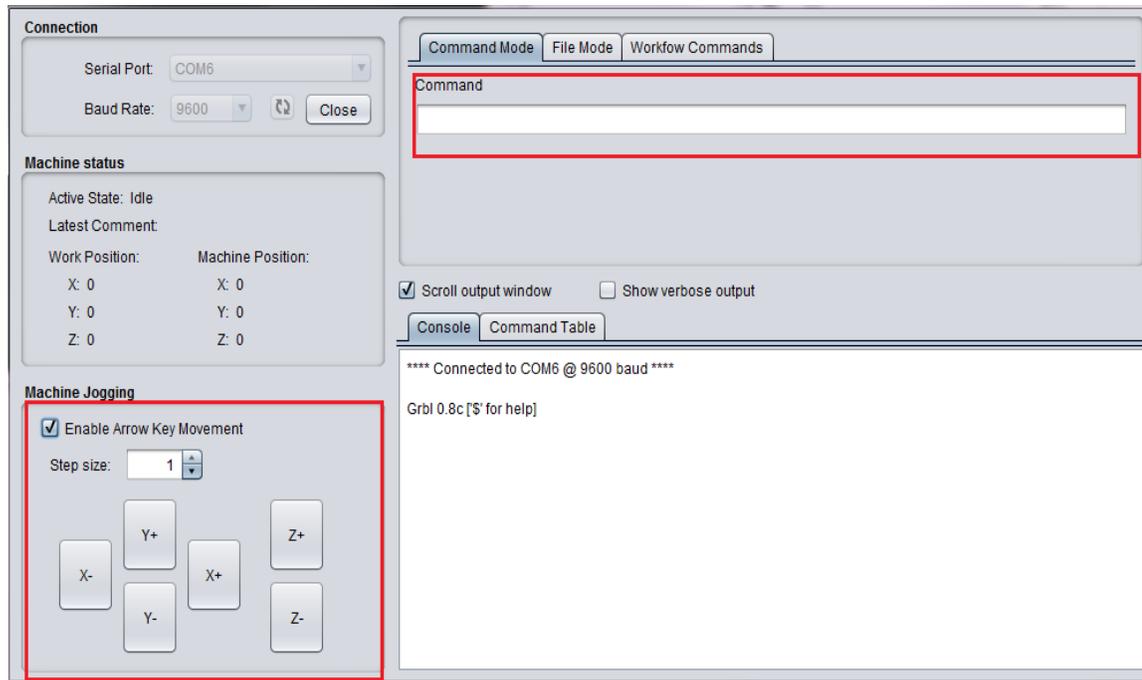


Figure III. 8. : mode manuel.

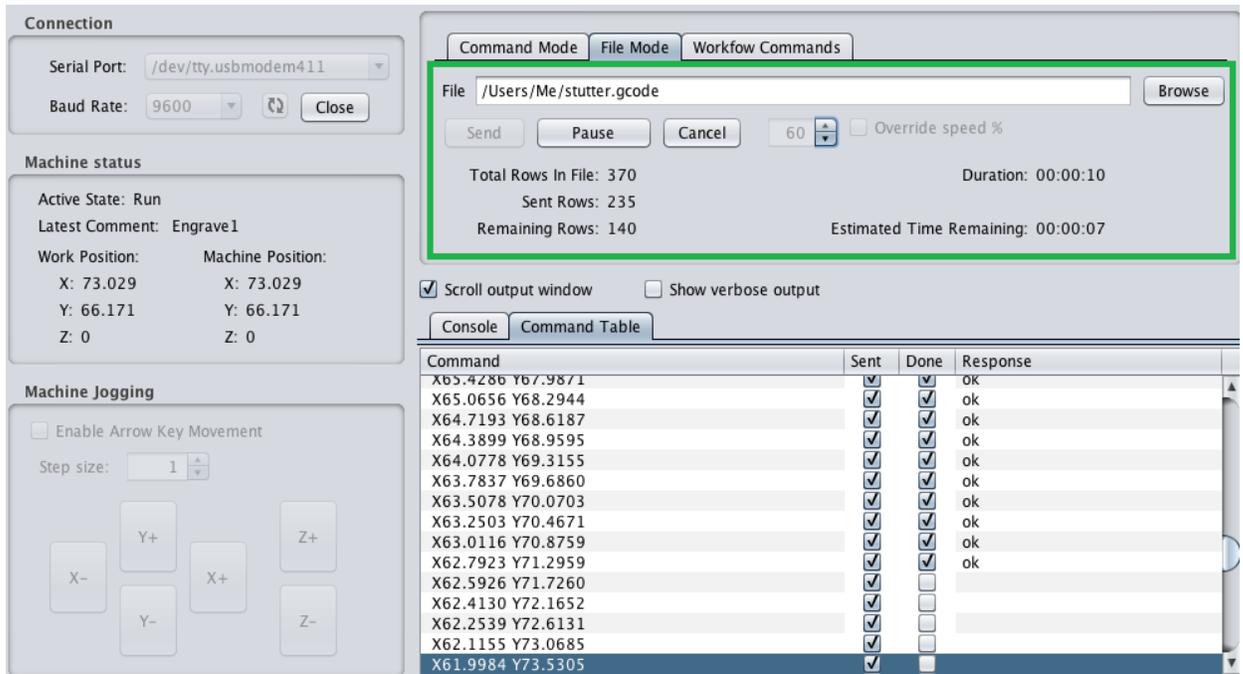


Figure III. 9. : Mode automatique.

### III. 5. Inkscape :

Inkscape est un logiciel libre, d'édition de graphismes vectoriels, Il permet de créer des graphismes de manière très précise et contient également de nombreux effets. le dessin est enregistré sous forme d'équations de courbes (chemins).

La figure III.10. Représente l'interface de logiciel Inkscape.

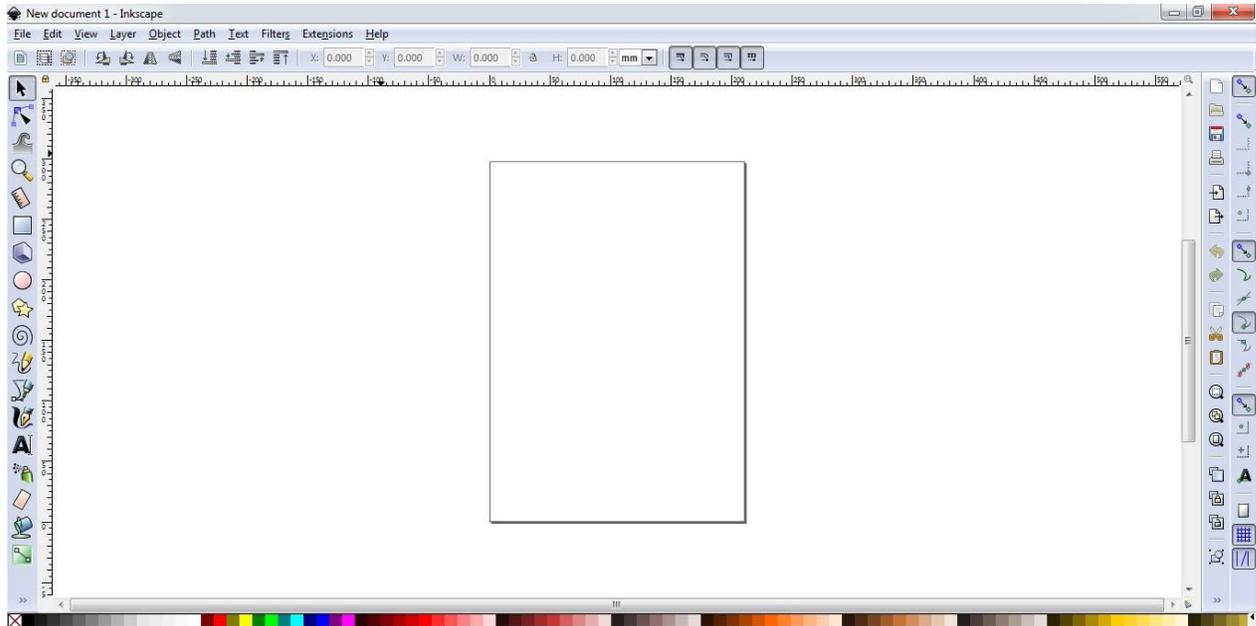


Figure III. 10. : interface de logiciel inkscape.

"G-code Tools" est une extension de Inkscape permettant (entre autre) de générer du G-code à partir des chemins d'une image vectorielle.

### III. 6. Mise en marche de la machine :

Pour tester de la machine on suit les étapes suivantes:

- Création du fichier G-code de la pièce à graver, avec le logiciel Inkscape qui permet de générer un fichier G-code. la forme à graver sur une plaque en forex est un rectangle.

La figure III. 11. représente exemple de la gravure qui on veut la graver.

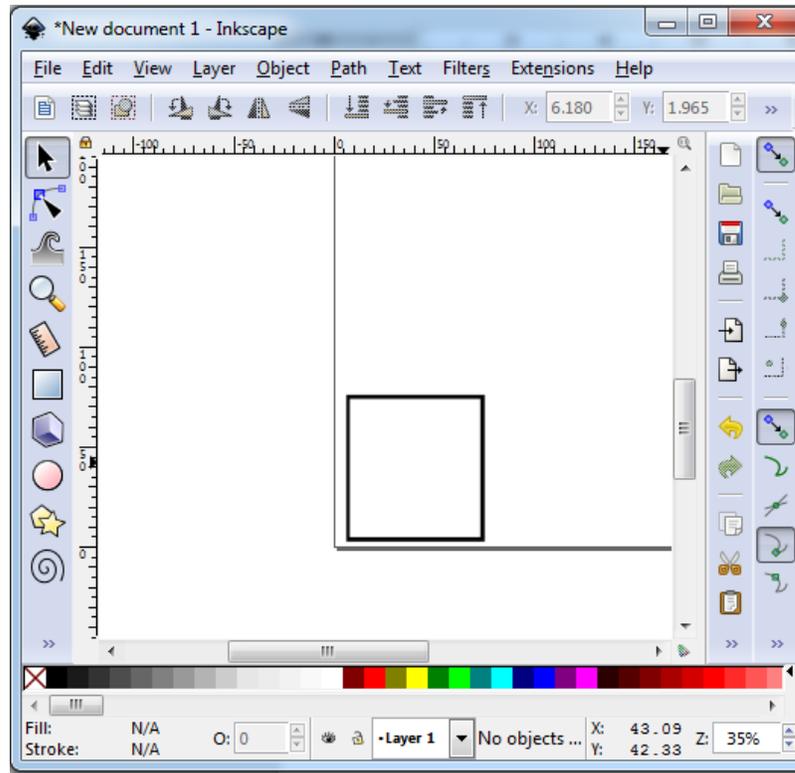


Figure III. 11. : exemple à graver

la figure III.12. Représente le fichier gcode qui est obtenu a l'aide de logiciel inkscape.

```

File Edit Format View Help
G90
G21
G17 G64 P0.001 M3 S3000
F100.0
G0 Z0
G0 X0 Y0.000
G1 Z-10
G1 X-40
G1 Y-40
G1 X0
G1 Y0
G0 Z10
M5
M2
    
```

Figure III. 12. : G-code de la pièce à graver.

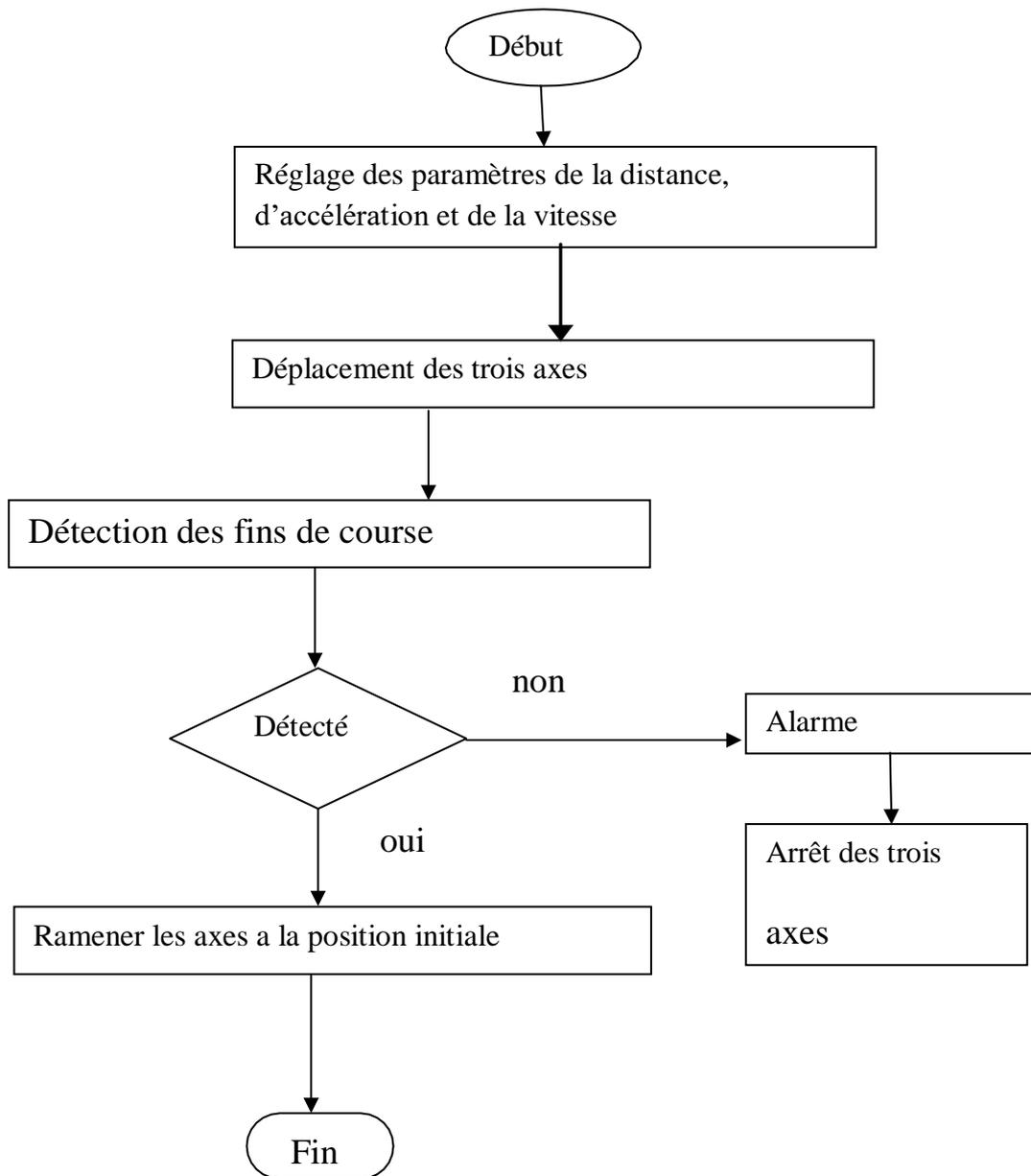
- Brancher L'arduino à l'ordinateur via un port USB.
- Configurer et téléverser le programme Grbl dans l'Arduino via port USB.
- Configurer l'interface de logiciel G-code sender . Choix de (port USB, la vitesse ...).
- Initialiser les axes au point zéro manuellement.
- Importer le fichier G-code de la pièce à graver dans l'interface de commande de logicielg-code sender.
- Mise sous tension de la boîte d'alimentation.
- Allumage de la broche.
- Envoi du fichier G-code et lancement de l'usinage.

### III. 7. Utilisation du firmware grbl :

#### III. 7. 1. paramétrages de la machine CNC :

Pour le premier démarrage de la machine on doit la paramétrer afin qu'elle détecte les contacteurs de fin course et positionner les axes à leurs positions initiales.

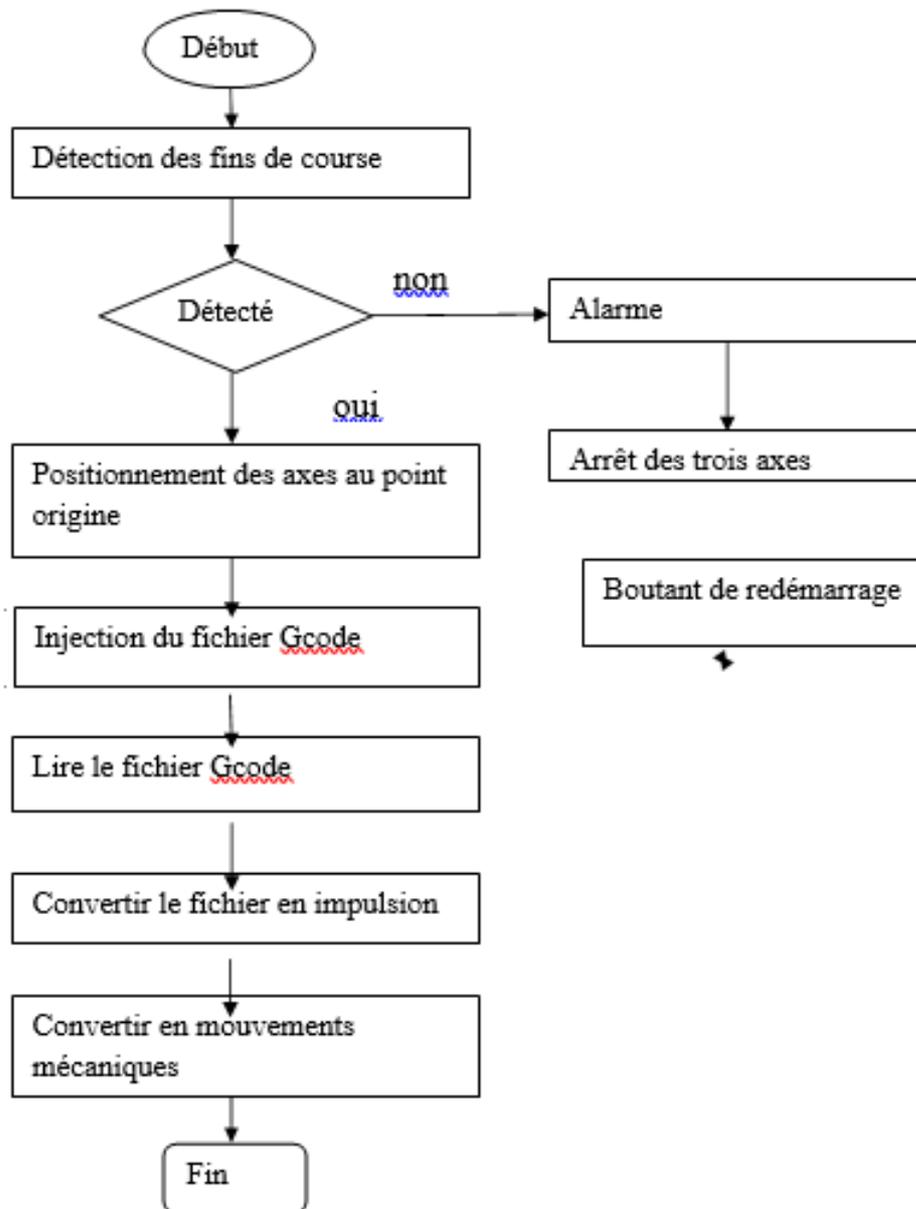
L'organigramme suivant nous montre les étapes à suivre



### III. 7. 2. Initialisation de la machine :

A chaque démarrage de la machine l'initialisation effectue la détection des fins de course X,Y,Z dans le cas échéant une alarme sera retenue et déclenche un arrêt des trois axes.

L'organigramme suivant nous montre les étapes à suivre



### III. 8. Conclusion :

Dans ce chapitre nous avons montré les étapes de conception que nous avons répartie en trois parties partie mécanique, partie électrique et partie informatique (programmation)

Et nous avons essayé d'expliquer la commande des trois moteurs avec une carte de commande basée sur Arduino et les circuits de puissance **A4983** et nous avons fait une étude sur les choix des moteurs et des circuits de puissance à utiliser.

Après cela nous proposons de valider nos résultats via la conception de la machine via SolidWorks et les mise en plan des pièces nécessaires d'un prototype.

Dans le chapitre suivant nous présentons la conception de la machine via SolidWorks et les mise en plan des pièces nécessaires du prototype suivant la démarche de conception présentée précédemment.