

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



UNIVERSITE A.MIRA-BEJAIA  
FACULTE de Technologie  
DEPARTEMENT Génie Electrique



**Mémoire de fin d'étude**  
En vue de l'obtention d'un diplôme de  
**Master**  
Filière : Electromécanique  
Option: Electromécanique

*Thème*

**ELABORATION D'UN SIMULATEUR  
POUR UNE MACHINE OUTIL À  
COMMANDE NUMÉRIQUE**

**Réalisé par:**

- ❖ **KESSAI** Sonia
- ❖ **SAICHE** Saida

**Encadré par:**

**Mr TAIB** Nabil

**Année Universitaire : 2017/2018**

# REMERCIEMENT

Nous remercions en premier lieu Dieu, le tout puissant de nous avoir donné la volonté de mener a terme ce travail qui représente le fruit de plusieurs années d'études.

Nous tenons à exprimer vivement nos profondes gratitudees à monsieur **TAIB NABIL** (maitre de conférence classe A) à l'université de Bejaia, pour nous avoir proposé, dirigé et encadré ce sujet de mémoire de fin de cycle ainsi que nos encadreurs de l'entreprise **DENZER Mr Chikhi mohand cherif & Mr Rachidiou lounis** pour leurs qualités humaines, leurs conseils et leur grande disponibilité tout au long de la durée de notre stage.

Nous tenant aussi à remercier nos enseignants pour nous avoir accompagné durant ces cinq années d'études à l'université et tout particulièrement monsieur **Ouatah El hanafi** (maitre de conférences classe B) pour nous avoir conseillé durant ce travail de mémoire.

Un vive remerciement à nos jury pour avoir accepter de juger notre travail,

et enfin nous tenant particulièrement à remercier nos camarades de **la promotion Electromécanique 2017/2018** pour leur soutien durant ce travail.

# *dedicaces*

*je dédie ce modeste travail en premier lieu a ma très chère et bien aimer mère, ensuite a mon cher père qui m'a tout donné pour réussir dans la vie et sans qui je ne serait pas la.*

*je dédie également ce travail a tout mes chers et vieux amis qui ont su me supporter durant toutes ces années.*

**SONIA**

# dedicaces

Je dédie ce modeste travail :

- ✓ A mes chers parents, pour tous leurs sacrifices, leur amour, leur tendresse, leur soutien et leurs prières tout au long de mes études, pour m'avoir encouragé et permis d'entreprendre ma formation Sans eux, je n'en serais pas là.
- ✓ A mes chères sœurs Lamia, Sara et Amel pour m'avoir épaulé moralement tous les jours dans la construction de ce mémoire.
- ✓ A mon très cher frère Ali et sa femme pour leur appui et leur encouragement.
- ✓ A mes chers neveux : Inès, Meriem et Adem.
- ✓ A toute la famille SAICHE pour leur soutien tout au long de mon parcours universitaire.
- ✓ A tous mes amis et mes chères copines Ibtissam, Ahlam, Sonia, Nadjet...
- ✓ A tout ce qui m'ont aidé de près ou de loin durant toutes mes années d'études.
- ✓ A mon binôme Sonia et toute sa famille.

**SAIDA**

## Sommaire

Présentation de l'entreprise .....	1
Présentation de l'entreprise DENZER technologie .....	7
Introduction générale.....	8
Introduction générale.....	9
Contexte .....	9
Objectifs visés .....	9
Plan du travail.....	10
Chapitre I : .....	11
Généralités sur les MOCN .....	11
I.1 Introduction.....	12
I.2 Historique de la commande numérique .....	12
Figure I.1: la première MOCN en 1952[5]. .....	13
I.3 Définition de la MOCN .....	14
Figure I.3 : cycle d'usinage .....	14
Figure I.4 : les parties qui composent la MOCN.....	15
I.3.1 Partie opératives.....	15
Figure I.5: la partie opérative .....	15
I.3.2 Partie commande.....	16
Figure I.6 : Fonction Originale d'une commande numérique .....	16
I.4 Architecture d'une MOCN .....	17
Figure I.7 : Architecture d'une MOCN.....	17
I.5 Différents types de MOCN .....	17
I.6 Caractéristiques principales résultantes de la structure de la MOCN.....	18
I.7 Classification des MOCN .....	18

I.8 Domaine d'utilisation .....	18
Figure I.8 : Domaine d'utilisation des MOCN [4] .....	19
I.9 Evolutions de la machine à commande numérique.....	20
I.10 Avantages et inconvénients liés aux MOCN .....	20
I.10.1 Avantages.....	20
I.10.2 Inconvénients .....	20
I.11 Flexibilité de la MOCN .....	21
I. 12 Nécessité technique et économique .....	21
I.13 Principaux procédés de fabrication .....	22
I.13.1 Le tour .....	22
Figure I.9: Un tour.....	22
I.13.2 La fraiseuse .....	22
I.13 Axes normalisés .....	23
I.15 Principe de la définition du positionnement et du sens des axes .....	24
I.16 Usinage à 3 axes.....	25
I.17 Usinage a 5 axes.....	25
I.18 Origines.....	26
I.18.1 Origine Machine (OM) .....	26
I.18.2 Origine mesure (Om) .....	26
I.18.3 Origine porte-pièce (Opp).....	26
I.18.4 Origine Programme (OP) .....	27
I.18.5 Point courant ou Point piloté (R) .....	27
I.19 Commande de la MOCN .....	27
I.19.1 Les type de commande.....	27
I.19.2 Directeur de commande .....	29
I.19.3 Différents mode de commande d'une MOCN [5] .....	29

I.19.4 Contribution de CN a la sécurité.....	31
I.1 Conclusion .....	31
Chapitre II :.....	32
Programmation des MOCN.....	32
II.1 Introduction .....	33
II.2 Gcode.....	33
II.2.1 Historique du langage Gcode .....	33
II.2.2 Définition de Gcode.....	34
II.2.3 Programmation de la machine à commande numérique.....	34
II.3 Elaboration d'un programme .....	35
II.4 Structure d'un programme.....	36
II.4.1 Format d'un bloc .....	36
.....	37
Figure II.3 :Format d'un bloc.....	37
II.4.2 Format d'un mot .....	37
II.4.3 Différents mots .....	38
II.4.4 Format d'un programme.....	40
Afin de réaliser un programme on suit les étapes suivantes: .....	40
II.5 Instructions de bases.....	42
II.5.1 La fonction G00.....	42
II.5.2 La fonction G01.....	42
II.5.3 Les fonctions G02 et G03 .....	43
II.5.4 La fonction G40.....	44
II.5.5 La fonction G41 .....	44
II.5.6 La fonction G42.....	45
II.5.7 La fonction G80.....	45

II.5.8 Les fonctions G90 et G91 .....	46
II.5.9 La fonction G92.....	46
II.5.10 La fonction G94.....	47
II.5.11 La fonction G95.....	47
II.5.12 La fonction M00 .....	48
II.5.13 La fonction M01 .....	48
II.5.14 La fonction M02 .....	49
II.5.15 La fonction M03 .....	49
II.5.16 La fonction M04 .....	49
II.6 Conclusion .....	49
Chapitre III : .....	50
Présentation des MOCN et réalisation la maquette .....	50
III.1 Introduction .....	51
III.2 Présentation de notre MOCN .....	51
III.2.1 Aperçu de notre MOCN .....	51
III.2.2 Description des différents composant de la MOCN.....	51
III.3 Principe de fonctionnement de notre MOCN.....	56
III.4 la maquette numérique .....	56
III.4.1 La CAO .....	56
III.4.2 Logiciel SolidWorks .....	57
III.4.3 Réalisation de la maquette.....	58
III.4.3.1 Etape de la réalisation des différentes pièces.....	53

III.4.3.2 Etape de la réalisation des sous-assemblages .....	56
III.4.3.2.1 la table.....	56
III.4.3.2.2 l'axe X.....	57
III.4.3.2.3 l'axe Y.....	58
III.4.3.2.4 l'axe Z.....	58
III.4.3.3 L'étape de la réalisation de l'assemblage final.....	58
III.4.3.3 L'étape de la réalisation de l'assemblage final.....	59
III.5 Conclusion.....	60
Chapitre IV :.....	61
Réalisation du simulateur sur Matlab.....	61
IV.1 Introduction.....	62
IV.2 Définition d'un simulateur.....	62
IV.3 But de notre simulateur .....	62
IV.4 Les étapes de la réalisation du simulateur.....	62
IV.4.1 Teste les déplacements du model simulink dans matlab.....	63
IV.4.1.1 Simscape.....	64
IV.4.1.2 L'export de SolidWorks.....	64
IV.4.1.3 l'import dans Matlab.....	64
IV.4.2 vérification des déplacements des trois axes.....	66
IV.4.3 Teste des fonctions du Gcode .....	66
IV.4.4 Développement et implémentation de l'algorithme de simulation.....	63
IV.5 Conclusion .....	68
Conclusion générale .....	69

Conclusion générale .....	70
---------------------------	----

## Liste des figures

<b>Figure I.1:</b> la première MOCN en 1952[5].	8
<b>Figure I.2:</b> un modèle récent d'une MOCN	9
<b>Figure I.3 :</b> cycle d'usinage	9
<b>Figure I.4 :</b> les parties qui composent la MOCN	10
<b>Figure I.5:</b> la partie opérative	10
<b>Figure I.6 :</b> Fonction Originale d'une commande numérique	11
<b>Figure I.7 :</b> Architecture d'une MOCN	12
<b>Figure I.8 :</b> Domaine d'utilisation des MOCN [4]	14
<b>Figure I.9:</b> Un tour	17
<b>Figure I.10:</b> Une fraiseuse	18
<b>Figure I.11:</b> Schéma simplifié d'une fraiseuse	18
<b>Figure I.12:</b> Axes normalisés d'une MOCN	19
<b>Figure I.13:</b> Règle des trois doigts pour le positionnement du système d'axe	20
figure (a) - figure (b)	22
<b>Figure I.14:</b> Commande manuelle	22
<b>Figure I.15:</b> boîte de vitesse	23
<b>Figure I.16:</b> CN de type Fargo	23
<b>Figure I.17:</b> CN de type Siemens	23
<b>Figure I.18:</b> le déplacement de la machine par positionnement	25
<b>Figure I.19:</b> le déplacement paraxial de la machine	25
<b>Figure I.20:</b> le déplacement continu de la machine	26
<b>Figure II.1 :</b> Classification des méthodes de programmation en fonction des compétences du programmeur [5].	30
<b>Figure II.2 :</b> structure d'un programme	31
<b>Figure II.3 :</b> Format d'un bloc	32
<b>Figure II.4 :</b> Format d'un mot	33

<b>Figure II.5</b> : diagramme d'un programme [5].....	36
A : tournage - B : Fraisage .....	37
<b>Figure II.6</b> : Fonction G00 [11] .....	37
A : tournage - B : Fraisage .....	38
<b>Figure II.7</b> : Fonction G01 [11] .....	38
A : tournage - B : Fraisage .....	38
<b>Figure II.8</b> : Fonction G02[11] .....	38
A : tournage - B : Fraisage .....	38
<b>Figure II.9</b> : Fonction G03 [11] .....	38
<b>Figure II.10</b> : Fonction G40 [11] .....	39
<b>Figure II.11</b> : Fonction G41 [13] .....	39
<b>Figure II.12</b> : Fonction G42 [11] .....	40
<b>Figure II.13</b> : Fonction G80 [11] .....	40
<b>FigureII.14</b> : Fonction G90 [11] .....	40
<b>Figure II.15</b> : Fonction G91 [11] .....	41
<b>Figure II.16</b> : Fonction G90 [11] .....	41
<b>Figure II.17</b> : Fonction G94 [11] .....	42
<b>Figure II.18</b> : Fonction G95 [11] .....	42
<b>Figure II.19</b> : Fonction M00 [11].....	43
<b>Figure II.20</b> : Fonction M01 [11].....	43
<b>Figure II.21</b> : Fonction M02 [11].....	43
<b>Figure II.22</b> : Fonction M03 [11].....	44
<b>Figure III.1</b> : Moteur pas à pas .....	48
<b>Figure III.2</b> : Fin de course .....	49
<b>Figure III.3</b> : Carte Arduino.....	50
<b>Figure III.4</b> : Carte Pilote.....	50

<b>Figure III.5 :</b> Carte shield .....	51
<b>Figure III.6 :</b> La table.....	57
<b>Figure III.7:</b> Axe X .....	57
<b>Figure III.8 :</b> Axes-Y .....	58
<b>Figure III.9 :</b> Axe Z .....	58
<b>Figure III.10 :</b> Les différentes contraintes d'assemblage .....	59
<b>Figure III.11 :</b> Assemblage final de notre MOCN.....	60
<b>Figure IV.1:</b> Etape de l'élaboration du simulateur .....	63
<b>Figure IV.2 :</b> schéma bloc de notre MOCN .....	65
<b>Figure IV.3 :</b> notre MOCN sur Matlab .....	65
<b>Figure IV.4 :</b> Schéma du simulateur .....	66
<b>FigureIV .5 :</b> Appel de la fonction G00 .....	67
<b>FigureIV.6 :</b> Appel de la fonction G02 .....	68

## Liste des tableaux

<b>Tableau1</b> : Signification des fonctions des déplacements.....	34
<b>Tableau2</b> : paramètres d'usinage .....	34
<b>Tableau3</b> : caractéristiques des moteurs pas à pas de notre MOCN .....	48
<b>Tableau4</b> : les pièces qui composent le bâti de notre MOCN.....	53

## Les abréviations

<b>Symbole</b>	<b>Désignation</b>
<b>MOCN</b>	Machine-Outil à Commande Numérique
<b>US Air force</b>	United States Air force
<b>MIT</b>	Massachusetts Institute of Technology
<b>CN</b>	Commande Numérique
<b>CNC</b>	Commande Numérique par Calculateur
<b>MO</b>	Machine-Outil
<b>PP</b>	Porte-Pièce
<b>P</b>	Pièce
<b>PO</b>	Porte-Outil
<b>O</b>	Outil
<b>OM</b>	Origine Machine
<b>Om</b>	Origine mesure
<b>Opp</b>	Origine porte-pièce
<b>OP</b>	Origine Programme
<b>PREF</b>	Prise Référence
<b>DEC</b>	Décalage
<b>R</b>	Point courant ou Point piloté
<b>DCN</b>	Directeur de Commande Numérique
<b>ISO</b>	International System Organisation
<b>EIA</b>	Energy Information Administration
<b>CFAO</b>	Conception et Fabrication Assistée par Ordinateur

---

# **Présentation de l'entreprise**

---

## **Présentation de l'entreprise DENZER technologie**

L'entreprise DENZER est née en 2016 dans la commune d'Akbou, wilaya de Bejaïa, depuis elle s'est étendue en créant un autre siège dans la ville de Bejaia et un autre à Alger, la capitale.

Elle s'est spécialisée dans les nouvelles technologies et l'informatique notamment en créant des applications et des logiciels pour des entreprises dans le besoin, depuis ces domaines d'applications se sont élargies et donc elle s'est intéressée au domaine de l'électronique en développant des systèmes visant à mieux commander des machines à usage domestique, industriel, ou encore administratif à titre d'exemple: des armoires électriques pour commander des ascenseurs et des systèmes qui permettent la gestion des fils d'attente dans les administrations ou encore des bracelets électroniques pour le suivi des malades atteints d'Alzheimer de plus l'entreprise innove dans l'impression 3D, elle a fabriqué une imprimante pour le prototypage rapide, qui porte une option en plus de celles existantes sur le marché actuel.

Notre rôle dans l'entreprise était de créer un simulateur pour une machine-outil à commande numérique en cours de conception qui va permettre de simuler des programmes afin de détecter tout éventuel problème.

Notre stage nous a permis aussi de mettre la main au côté mécanique, électrique et électronique de la machine puisque durant les 2 mois de stage on a travaillé avec deux autres binômes, le premier avait pour rôle de faire le côté conception et réalisation en faisant une étude approfondie de la machine ainsi que les calculs mécaniques et le 2ème avait été chargé de faire la commande de la machine-outil en utilisant une carte Arduino Mega et donc non seulement on a acquis de nouvelles informations durant nos recherches pour notre propre thème mais on a aussi emmagasiné des informations sur tous les autres aspects de la machine.

# **Introduction générale**

## **Introduction générale**

### **Contexte**

Le monde de l'industrie étant en constante évolution, les moyens et les techniques de fabrications doivent suivre le mouvement, dans le domaine de la fabrication mécanique la machine-outil à commande numérique (MOCN) a une très grande importance car elles permettent d'obtenir des surfaces très complexes avec des conditions de coupe optimales ou encore du fait de sa grande largeur d'application industrielle telles que l'aéronautique, l'automobile.

La conception des machines-outils comme pour tout autre projet technologique passe par plusieurs étapes primordiales et nécessaires en commençant par l'étape de la recherche qui déterminera le principe de fonctionnement de cette machine ainsi que ces caractéristiques mécaniques, électroniques et électriques mais aussi définir un cahier des charges adapté aux besoins et aux exigences, cette étape est la base de tout projet de conception, ensuite elle viendra l'étape de la réalisation du simulateur qui validera ou non son bon fonctionnement par des méthodes souvent informatiques et cela en réalisant tout ou une partie du projet de façon virtuelle et ainsi tester ses performance et ces limites, enfin on termine par l'étape de la validation et du développement du produit final.

Dans ce contexte le thème de notre mémoire se basera sur l'étude d'une machine-outil à trois axes en processus de conception dans l'atelier de l'entreprise DENZER afin de réaliser son simulateur.

### **Objectifs visés**

Ce travail a pour objectif de faire une étude plus au moins poussée sur une machine-outil en processus de conception afin de réaliser son simulateur et cela pour plusieurs raisons:

- détecter les limites du programme a base du langage Gcode
- avoir un aspect visuel de la machine pour pouvoir la présenter aux futurs fabricants ou utilisateurs;
- avoir une vision plus large des fonctions préparatoires de type G et les fonctions auxiliaires de type M;
- avoir la possibilité de simuler tout ou une partie d'un programme d'usinage avec un simulateur propre à la machine et sans être obligé d'avoir recourt aux logiciels disponibles dans le marché;

- permettre au futur acquéreur de la machine de s'entraîner et de se familiariser avec le produit avant même de l'avoir en mains;

### **Plan du travail**

Comme nous l'avant déjà précisé la réalisation d'un simulateur de n'importe quel produit nécessite des connaissances approfondies du produit en question et c'est pour cela qu'on va commencer dans le premier chapitre par des généralités sur les MOCN en expliquant leurs principes de fonctionnement, leurs caractéristiques et les différents type de machine qu'existent dans le marché, ensuite dans le 2eme chapitre on va s'intéressé à leur programmation avec le langage Gcode , un langage simple et efficace et qui est utilisé dans pratiquement toutes les MOCN, après le 3eme chapitre sera décomposé en deux parties, dans la première partie on va présenter la machine-outil qui sera le sujet de notre étude et cela en présentant ces différents composants, son principe de fonctionnement et les caractéristiques propre à elle et dans la deuxième partie de ce chapitre on va réaliser l'aspect visuel de notre simulateur en créant une maquette numérique à l'aide du logiciel de conception SolidWorks et enfin dans le dernière chapitre, on va importer cette maquette dans Matlab ou plus précisément dans Simscape afin de réaliser un algorithme qui va nous permettre de simuler un programme Gcode et voir directement les déplacements des 3 axes dans la partie visuel de notre MOCN.

**Chapitre I :**  
**Généralités sur les MOCN**

## I.1 Introduction

Les machines-outils que nous connaissons maintenant diffèrent des modèles anciens de part leur sophistication, leur robustesses ou encore de leur polyvalence, les contraintes du monde de l'industrie leur ont permis d'évoluer et de s'adapter aux exigences de la production moderne. De ce fait dans ce premier chapitre on va commencer par présenter l'évolution historique de la machine à commande numérique après on va expliquer les principales caractéristiques qui définissent une MOCN puis on va s'intéresser à sa commande et tout cela dans le but de mieux cerner le principe de fonctionnement de notre machine.

## I.2 Historique de la commande numérique

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

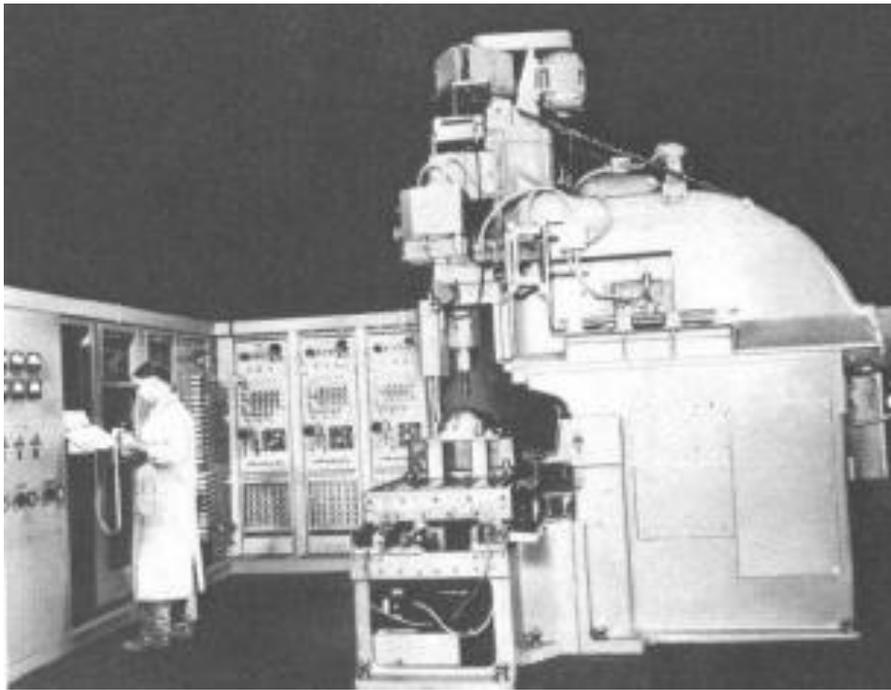
Au printemps 1949, il confie alors au Massachusetts Institute of Technology (MIT) le soin de développer des asservissements capables de piloter une machine qui recevra des instructions intermittentes à partir d'un lecteur de cartes.

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

Il faut encore attendre quelques années de vastes fonds de l'US Air Force et l'appui des chercheurs du MIT pour rendre la première MOCN réellement opérationnelle. Les différentes étapes de développement de la CN sont les suivantes. (La figure ci-dessous représente la première MOCN) [1].

- ❖ 1959 : Bendix acquiert le brevet de Parsons et fabrique la première CN industrielle
- ❖ 1955 : à Font du Lac (Wisconsin), le constructeur américain Giddins & Lewis commercialise la première MOCN.

- ❖ 1959 : apparition de la CN en Europe (foire de Hanovre).
- ❖ 1964 : en France, la Télémécanique Électrique lance la CN NUM 100 conçue à base de relais Téléstatic.
- ❖ 1968 : la CN adopte les circuits intégrés ; elle devient plus compacte et plus puissante.
- ❖ 1972 : les mini-calculateur remplacent les logiques câblées ; la CN devient CNC .
- ❖ 1976 : développement des CN à microprocesseurs.
- ❖ 1984 : apparition de fonctions graphiques évoluées et du mode de programmation conventionnel.
- ❖ 1986 : les CN s'intègrent dans les réseaux de communication, début de l'ère de la fabrication flexible (CIM).
- ❖ 1990 : développement des CN à microprocesseurs 32 bits



**Figure I.1: la première MOCN en 1952[5].**



Figure I.2: un modèle récent d'une MOCN

### I.3 Définition de la MOCN

La MOCN 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, cassette, USB, ...), une MOCN est dite à calculateur lorsque la machine-outil est équipée d'une commande numérique capable de réaliser les calculs des coordonnées des points définissant une trajectoire (interpolation) et donc elle est appelée CNC (Commande Numérique par Calculateur) [2].

Actuellement la plupart des MOCN sont des CNC. (Figure I.3)

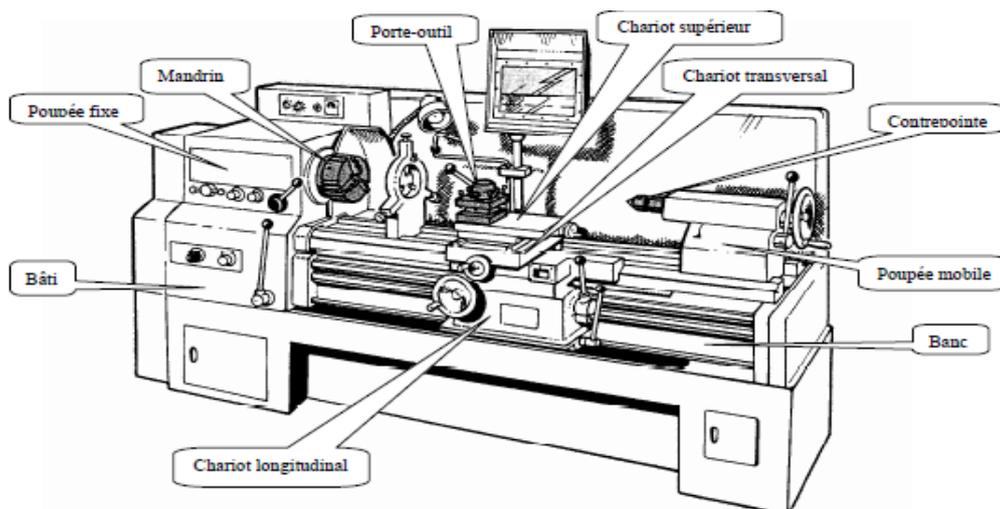
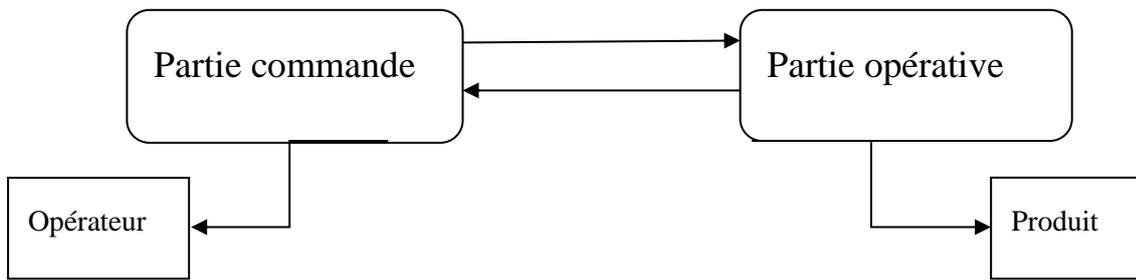


Figure I.3 : cycle d'usinage

Elle se compose de deux parties:



**Figure I.4 : les parties qui composent la MOCN**

### I.3.1 Partie opératives

C'est la partie physique du travail tel que les déplacements selon les 3 axes X, Y, Z elle se compose de :



**Figure I.5: la partie opérative**

- capteurs renseignant à tout moment sur la position du mobile sur chaque axe.
- de moteurs pas à pas qui réalisent le déplacement de l'outil ou de la table
- une dynamo tachymétrique qui assure les mesures de vitesse.
- un support outil
- un socle qui assure l'indépendance de la machine du sol

- une table support pièce qui effectuent ou non des mouvements selon les axes.

La figure ci-dessous représente la partie opérative la MOCN.

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

- Chargement et déchargement (pièce port pièce).
- Chargement et déchargement (outils port outils).
- Intervention manuelles nécessitées par l'usinage et l'entretien.
- Surveillance de commande.

### I.3.2 Partie commande

Elle est constituée d'une armoire dans laquelle se trouve :

- pupitre qui permet de rentrer la commande à l'aide d'un clavier
- le lecteur de donné
- des sorties pour les liaisons avec les périphériques externes
- écran de visualisation de toutes les entrées
- le calculateur
- les cartes électroniques et les mémoires

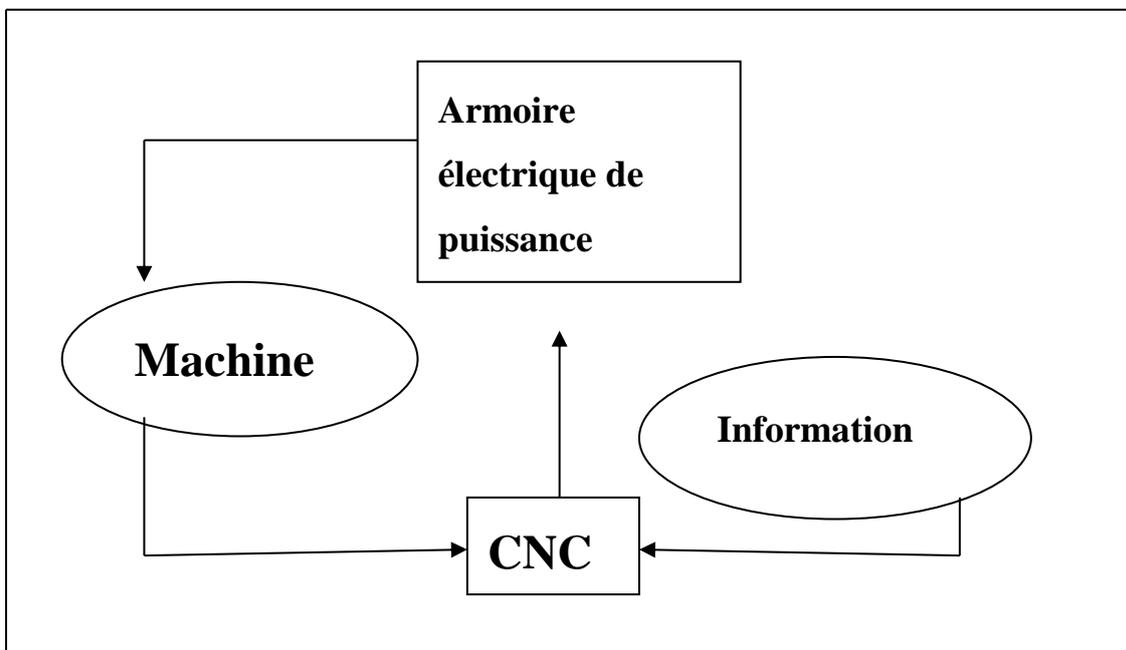


Figure I.6 : Fonction Originale d'une commande numérique

#### I.4 Architecture d'une MOCN

Généralement la MOCN est composée de cinq ensemble organisée en boucle qui sont les suivant: la structure de la (MO), la porte pièce (PP), la pièce (P), le porte-outil (PO) et enfin l'outil (O).

Elles sont positionnées de tel manière à faire coïncider le Pt générateur de l'outil avec la trajectoire programmé par rapport à l'origine programme, choisit judicieusement sur la pièce.

Ces ensembles sont disposés de la manière suivante :



**Figure I.7 : Architecture d'une MOCN**

#### I.5 Différents types de MOCN

On distingue plusieurs types de machines [3]:

- les machines à enlèvement de copeaux : les perceuses, les tours 2 et 4 axes, les centres de tournages 5 axes, les fraiseuses 2 axes 1/2, 3 axes, les centres d'usinage, 3 à 5 axes, les rectifieuses, les affûteuses, 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.6 Caractéristiques principales résultantes de la structure de la MOCN**

La structure de la machine-outil à commande numérique est d'une grande importance car elle permet d'améliorer ses performances qui sont les suivantes :

- puissance et vitesse élevées,
- robuste et bonne résistance à l'usure,
- déplacement rapide, précis, sans saccade,
- accélération et décélération très élevées,
- spécifications métrologiques très serrées,
- frottement et jeu très faibles,
- peu de vibration,
- faible échauffement,

### **I.7 Classification des MOCN**

On peut classer les MOCN selon :

- le système d'entrée des informations
- le mode de fonctionnement du système de mesure
- le mode d'usinage
- le système de fonctionnement de la machine
- le nombre d'axes de la machine (machine à 3 axes, machine à 5 axes ...)

Afin d'améliorer des performances techniques de la MOCN on peut ajouter les éléments suivant:

- un magasin de changement d'outil
- une commande adaptative
- un ordinateur et des supports de communication
- évacuateur de copeaux
- une table de montage
- chargeur de convoyeur de pièce
- Dispositif de contrôle de pièce

### **I.8 Domaine d'utilisation**

La MOCN est utilisé dans de nombreux secteurs industriels (Aéronautique, automobile, Métallurgie, Bois, Textile) et cela est dû au coup qui est assez rentable par rapport aux autres

machines. (Il est représenté dans la figure ci-dessous). Elle est aussi associée à de nouvelles technologies de façonnage (Laser, Électroérosion, Jet d'eau) [1].

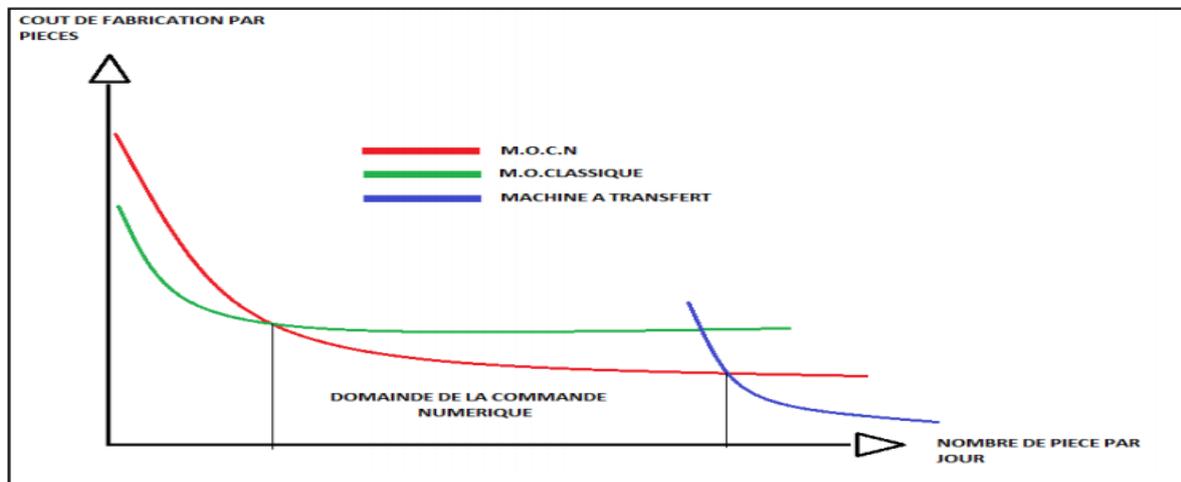


Figure I.8 : Domaine d'utilisation des MOCN [4]

### **I.9 Evolutions de la machine à commande numérique**

Les machines-outils spécialisées (aléseuses-perceuses, fraiseuses) à commande numérique ont évolué pour passer d'une machine à simple fonction au centre d'usinage à commande numérique permettant d'usiner des formes complexes sans démontage de la pièce. Ces centres d'usinage sont généralement équipés de magasins d'outils (tourelles, tables, chaînes) sur lesquels sont disposés les différents outils. Les changements d'outils équipant la (ou les) tête(s) d'usinage sont programmés en fonction de la définition numérique de la pièce.

### **I.10 Avantages et inconvénients liés aux MOCN**

#### **I.10.1 Avantages**

- permet la réalisation d'usinages impossible sur les machines conventionnelles [1]:
  - surface complexes
  - très grand nombre d'opérations
- favorise les très petites séries et les pièces unitaires telles que les prototypes:
  - pièces en cours de conception ou modifiées fréquemment
  - productions à la demande ou juste à temps (réduction de la taille des lots)

Révision:

- machine de meilleure qualité en général
- moins de montage, démontage de la pièce
- fidélité de reproduction: répétabilité (pas d'opérateur humain dans la chaîne de pilotage).

#### **I.10.2 Inconvénients**

- Pour bénéficier de la majorité des avantages précédents, il faut que tout le parc machine de l'entreprise soit des MOCN,
  - Investissement initial plus important.
  - Rentabilité pas immédiate.
- Amortissement impose souvent un travail en 2 ou 3 équipes,
- Programmation et électronique demandant de la qualification,
- Fausse fragilité de l'électronique,
- Equipement annexe : ordinateur, logiciel, banc de réglage des outils, changeur d'outils,
- Changement dans les méthodes de préparation et de fabrication: Réticence du personnel au changement.

### **I.11 Flexibilité de la MOCN**

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

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

La grande souplesse d'utilisation de la CN offre les avantages:

- Changement aisé du programme d'usinage des pièces,
- Réduction des en-cours de fabrication,
- Réduction des outillages et suppression des gabarits,
- diminution du nombre des outils spéciaux et des outils de formes,
- Réduction des temps de réparation et de réglage du poste de travail (la plupart des réglages, en particulier, des outils, étant effectués hors machine),
- Prise en compte rapide des modifications d'usinage (il est plus facile de modifier une ligne de programmation qu'un outillage spécial ou un gabant),
- définition plus rapide et plus fiable des conditions optimales d'usinage
- réduction du nombre de prise de pièces du fait de l'universabilité de la machine,
- Diminution des temps d'attente entre les diverses machines d'un atelier, - Possibilité de réaliser des pièces complexes en gérant des déplacements simultanés sur plusieurs axes [1].

### **I. 12 Nécessité technique et économique**

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

Une fois vérifié et validé, un programme assure la réalisation de 2, 10 ou 1000 pièces identiques avec la même régularité de précision et la même qualité d'usinage, sans que l'habileté de l'opérateur n'intervienne.

il convient, en outre, de souligner que la CN ouvre de nouvelles perspectives en permettant la définition de pièces complexes qu'il est pratiquement impossible de concevoir et de fabriquer sur les MO conventionnelle. [1]

### I.13 Principaux procédés de fabrication

- Perçage, taraudage
- Tournage, alésage
- Fraisage
- Rectification
- Oxycoupage, soudure en continu, par points
- Poinçonnage, cisailage

#### I.13.1 Le tour

Cette machine sert principalement à usiner des pièces de révolution. La pièce est fixée dans le mandrin. Celle-ci est mise en rotation par le moteur de broche. L'outil suit une trajectoire qui interfère avec la pièce. L'outil est muni d'une arête coupante, il en résulte un enlèvement de matière « les copeaux » [5].



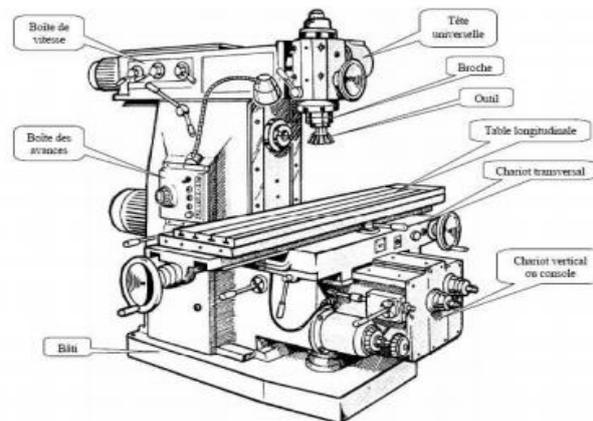
Figure I.9: Un tour

#### I.13.2 La fraiseuse

Cette machine (Figure 10) sert principalement à usiner des pièces prismatiques. La pièce est fixée dans l'étau. L'outil est mis en rotation par le moteur de broche et suit une trajectoire qui interfère avec la pièce. L'outil est muni d'une arête coupante, il en résulte un enlèvement de matière « les copeaux » [5].



**Figure I.10:** Une fraiseuse



**Figure I.11:** Schéma simplifié d'une fraiseuse

### I.13 Axes normalisés

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

Il permet d'obtenir une position, par une instruction numérique, à la résolution du moyen de mesure près [1].

Un axe est constitué de la façon suivante :

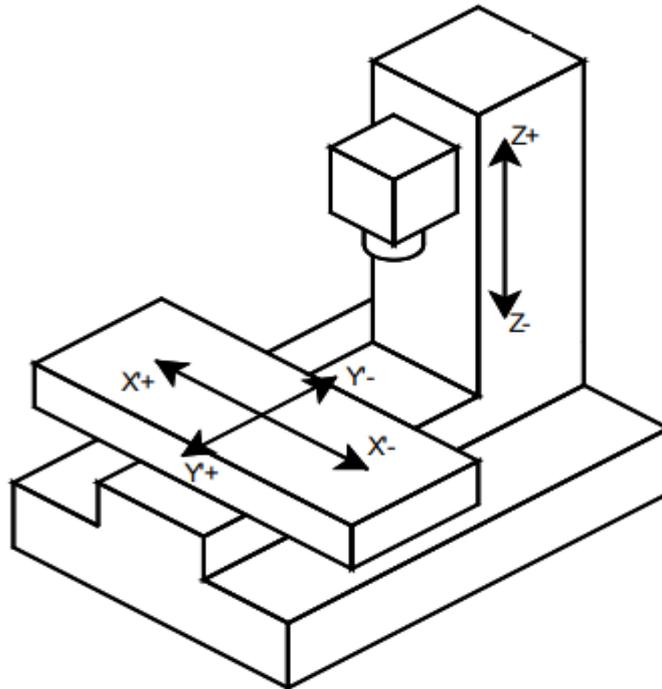
- ❖ Un chariot mobile sur glissières.
- ❖ Un système de transmission vis-écrou (vis à billes).
- ❖ Un moteur et un réducteur.

- ❖ Un dispositif de mesure de vitesse.
- ❖ Un dispositif de mesure de position.

Chaque axe de déplacement est donc assujéti à un asservissement en boucle fermée, dont le principe consiste à mesurer continuellement la position réelle du mobile et à la comparer avec la grandeur d'entrée, ou position de consigne, que délivre la CN pour atteindre la nouvelle position programmée; dès que l'écart entre les deux mesures s'annule, le mobile s'arrête.

Le déplacement de la table ou de l'outil d'un point à un autre implique la connaissance :

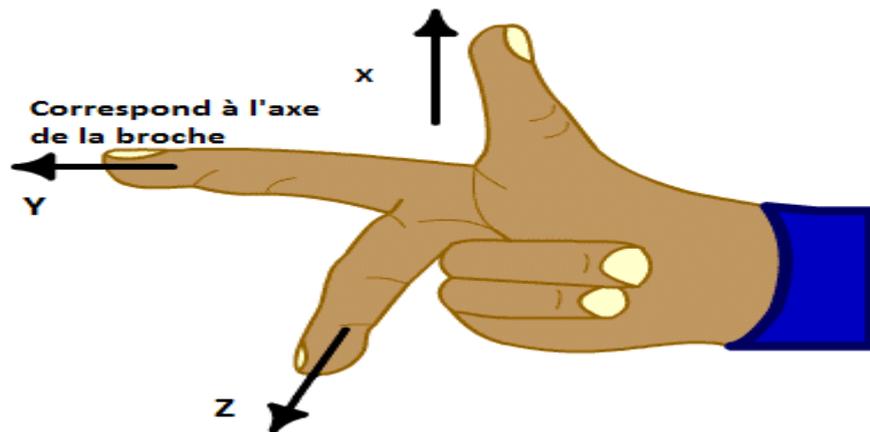
- ❖ De l'axe (X, Y, Z,...) sur lequel le déplacement doit s'effectuer ;
- ❖ Des coordonnées du point à atteindre ;
- ❖ Du sens de déplacement (+ ou -) ;
- ❖ De la vitesse de déplacement (d'avance) de la table ou de l'outil.



**Figure I.12:** Axes normalisés d'une MOCN

### I.15 Principe de la définition du positionnement et du sens des axes

Pour positionner le système d'axes, il est simple d'utiliser la règle des trois doigts. L'axe du majeur étant, par convention, parallèle à celui de la broche de travail. Les axes additionnels présents sur certains centres d'usinage, définissant les mouvements secondaires parallèles aux axes principaux (X, Y et Z), sont repérés par U, V et W. [6]



**Figure I.13:** Règle des trois doigts pour le positionnement du système d'axe

Compte au sens des axes, on considère que la pièce est fixe par rapport au bâti et que l'outil à toutes les mobilités, le sens positif de l'axe est tel que lorsque l'outil s'éloigne de la pièce, les dimensions de la pièce croissent, lorsque qu'il n'est pas possible de définir le sens avec cette méthode, choisir le sens de travail principal, sur les machines comportant des pièces en rotation (tour, rectifieuse), l'axe X de mouvement est radial et son sens positif correspond à l'accroissement de la distance entre l'outil et l'axe de rotation de la pièce, disposant du sens de deux axes, le sens du troisième axe est tel que le repère formé soit direct [7].

### I.16 Usinage à 3 axes

La technique la plus employée dans la fabrication des pièces mécanique est sans doute l'usinage à 3 axes, relativement simple dans son exécution, elle fait recours à des machines-outils classiques comme la fraiseuse ou la tour qui permettent un travail de la matière sur 3 axes (X, Y et Z). La machine-outil procède alors à l'enlèvement des copeaux suivant trois directions de base correspondant aux axes d'une surface plane. Tout à fait adaptée aux pièces peu profondes, cette technique basée sur l'utilisation d'outils longs présente cependant de grosses limites quand il s'agit de traiter une pièce profonde avec des cavités étroites. Le travail peut alors devenir très laborieux, avec une finition souvent peu aboutie.

### I.17 Usinage à 5 axes

Comme son nom l'indique, l'usinage 5 axes fait recours à un outil qui se déplace dans cinq différentes directions correspondant aux 3 axes linéaires X, Y et Z, auxquels s'ajoutent deux axes, A et B, autour duquel tourne l'outil. Un tel parcours permet d'approcher la pièce dans toutes les directions et de la traiter sur cinq côtés, en une seule opération.

Contrairement à l'usinage 3 axes, cette technique est parfaitement adaptée aux pièces profondes et aux matériaux durs avec lesquels elle garantit un niveau de précision élevé grâce

à l'utilisation d'outils d'usinage plus courts. La vitesse d'usinage est également plus grande, tandis que les vibrations d'outils sont réduites.

Par ailleurs, alors que l'usinage 3 axes autorise des opérations plus simples pour les dispositifs, l'usinage 5 axes nécessite de plus longs préparatifs mais moins de manipulation de la matière.

### **I.18 Origines**

Les programmes d'usinage sont réalisés à partir d'une origine appelée "origine programme" (OP) définie par le programmeur lui-même, le programme a pour but de commander les déplacements de la machine afin de réaliser la pièce souhaitée à partir d'une pièce vierge, ces déplacements sont réalisés dans un repère orthonormé (x,y,z) normalisé basé sur la structure de la machine .

#### **I.18.1 Origine Machine (OM)**

L'origine machine coïncide avec la position du point piloté (point courant ou point référence R) au moment où les axes de la machine ont en butée d'initialisation. L'origine machine n'a aucune influence sur le réglage ou la programmation de la machine, en effet elle dépend essentiellement de la technologie de mesure adoptée (mesure incrémental ou mesure absolue).

#### **I.18.2 Origine mesure (Om)**

C'est un point préférentiel défini sur chaque axe qui est l'origine absolue de la mesure. Les coordonnées de ce point peuvent être introduites voire modifiées par des paramètres machines spécifiques.

#### **I.18.3 Origine porte-pièce (Opp)**

Indépendante du système de mesure, cette origine est définie par un point de la pièce sur lequel on est capable de se positionner. Elle est déterminée par rapport à l'origine mesure par le paramètre Prise Référence PREF.

#### **I.18.4 Origine Programme (OP)**

Indépendante du système de mesure, c'est l'origine du trièdre qui a servi à établir le programme. Elle est définie par rapport à l'origine porte-pièce par le décalage DEC1 (ou DEC3 dans cas d'un centre d'usinage).

### **I.18.5 Point courant ou Point piloté (R)**

C'est un point de référence outil et appartient à la machine. Ce point est défini comme l'intersection de l'axe de l'alésage de logement de l'outil, avec la face de la tourelle dans le cas d'un tour ou, avec le plan de jauge de logement outil dans la broche de centre d'usinage. Dans des cas particuliers le plan de jauge est confondu avec la face de la broche. On peut lire à chaque instant la position de ce point par rapport à l'origine programme (OP) ou par rapport à l'origine mesure (OM).

## **I.19 Commande de la MOCN**

### **I.19.1 Les type de commande**

#### **I.19.1.1 La commande manuelle**

Un opérateur est chargé du déplacement de l'outil selon la trajectoire d'usinage souhaité, les mouvements des axes sont générés par les manivelles. Les deux figures ci-dessous représentent les deux commandes manuelles



figure (a)



figure (b)

Figure I.14: Commande manuelle

Des moteurs permettent aussi de choisir des vitesses d'avance suivant les axes de déplacement. Le choix de ces vitesses s'effectue par l'intermédiaire d'une boîte de vitesse mécanique.



Figure(a)



Figure(b)

Figure I.15: boîte de vitesse

### I.19.1.2 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 [4].



Figure I.16: CN de type Fargo



Figure I.17: CN de type Siemens

### **I.19.2 Directeur de commande**

Le directeur de commande (DCN) de contournage équipe pratiquement toutes les MOCN actuelles (tours, fraiseuses, centre de tournage ou de fraisage).

Il peut commander et contrôler :

- Des usinages linéaires entre deux points préalablement programmés; avec des trajectoires parallèles, perpendiculaires ou inclinées par rapport aux axes de la machine (commande paraxiale)
- L'usinage de toutes les formes dès l'instant que celles-ci sont définies mathématiquement (commande en contournage).

Sur un tel dispositif et grâce à son interpolateur, les vitesses d'avance et de coupe, ainsi que les trajectoires sont contrôlées en permanence; on distingue trois types d'interpolateur [5]:

- Les interpolateurs linéaires,
- Les interpolateurs circulaires,
- Les interpolateurs paraboliques utilisés dans l'industrie automobile et aéronautique.

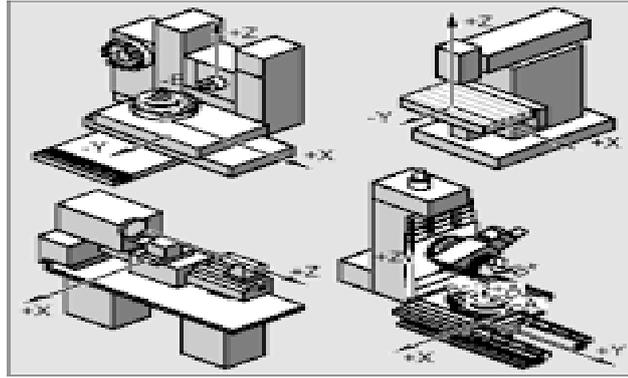
### **I.19.3 Différents mode de commande d'une MOCN [5]**

L'usinage par enlèvement de matière se résume à la conduite d'un mobile (outil ou pièce) suivant un déplacement déterminé, par un ordre (humain ou numérique). Ce déplacement peut être linéaire, angulaire ou circulaire en fonction des possibilités d'asservissement des mouvements. C'est donc naturellement que l'on a classé les MOCN suivant le mode déplacement des tables support de pièce [5].

Trois générations des MOCN ont été développées dans l'industrie :

#### **I.19.3.1 Machine à déplacement par positionnement (point à point)**

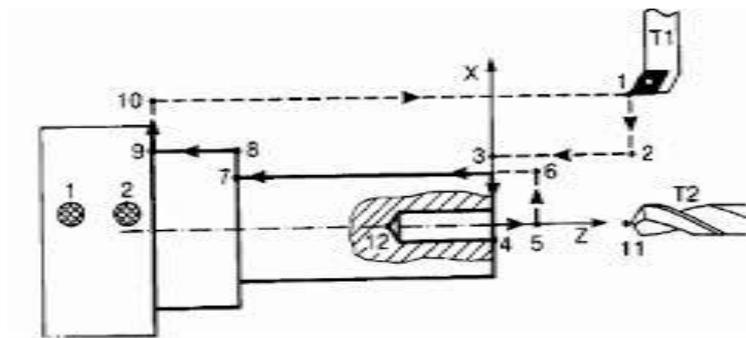
Amène le mobile à un point ou successivement à plusieurs points définis par des coordonnées numériques, dans un plan.



**Figure I.18 :** le déplacement de la machine par positionnement

### I.19.3.2 Machine a déplacement paraxial

C'est une machine à commande numérique Point à Point permettant d'usiner durant les déplacements parallèles aux axes.



**Figure I.19:** le déplacement paraxial de la machine

### I.19.3.3 Machine à déplacement continu (countournage)

C'est une machine à Commande Numérique intégrale développée grâce aux formidables possibilités de l'interpolation (combinaisons simultanées de déplacement du mobile sur deux ou plusieurs axes), qui permettent l'usinage de formes toujours plus complexes. L'interpolateur est le calculateur de la Commande Numérique.

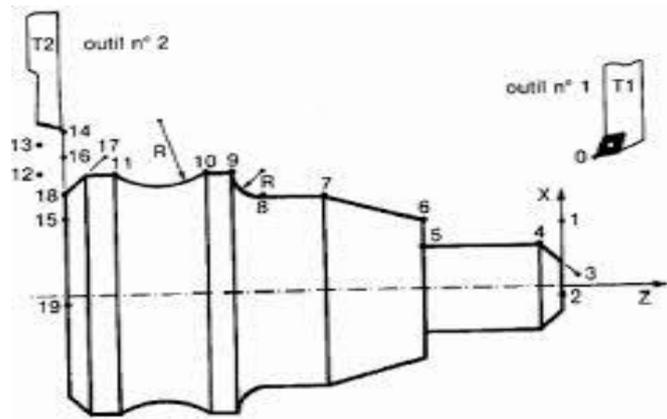


Figure I.20: le déplacement continu de la machine

#### I.19.4 Contribution de CN a la sécurité

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

- en premier lieu, parce qu'elle connaît très précisément l'enveloppe de travail dans laquelle doivent évoluer les outils (possibilité de mémorisation des courses maximale des organes mobiles),
- ensuite, parce qu'elle permet une simulation graphique hors usinage des nouveaux programmes nouvellement créés, pour vérification et détection des risques éventuels de collision,
- enfin, parce qu'en exerçant une surveillance permanente de l'usinage en cours, elle peut décider d'interrompre l'usinage d'une pièce et alerter l'opérateur en cas d'incident,
- les constructeurs prévoient des dispositifs de protection très élaborés (contre les projections de copeaux ou de liquide d'arrosage, notamment) qui ne s'impose pas nécessairement sur une machine outils conventionnelle [1].

#### I.1 Conclusion

Dans ce chapitre on a essayé de se familiariser avec la machine à commande numérique en faisant une recherche bibliographique détaillée sur tout ce qui concerne cette dernière afin de mieux cerner ses caractéristiques principales et son mode de fonctionnement pour mieux appréhender le chapitre prochain.

**Chapitre II :**  
**Programmation des MOCN**

## II.1 Introduction

Après avoir défini dans le premier chapitre les caractéristiques et les différents types des MOCN, on va s'intéresser maintenant à leur programmation à l'aide du langage G-code, et cela en commençant par son historique puis on va expliquer la manière de programmer avec ce langage en précisant la structure et le format de chaque ligne de code, enfin on s'est intéressé à certaines fonctions présentes dans le G-code qui sont les plus répétitives et qui sont pour la plupart indispensables à la programmation des MOCN.

## II.2 Gcode

### II.2.1 Historique du langage Gcode

Le Gcode 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 des 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 particularités des machines qu'ils utilisent, et les systèmes de CFAO doivent se limiter au plus petit dénominateur commun des machines qu'ils commandent.

Beaucoup de fabricants ont essayé de contourner cette difficulté à rester compatible en suivant la route tracée par 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 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 [8].

### **II.2.2 Définition de Gcode**

Le Gcode est le langage utilisé pour contrôler une machine à commande numérique. Il s'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 "commande" 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. Le Gcode n'est pas utilisé exclusivement pour des fraiseuses à CN, mais aussi pour des tours, des imprimantes 3D et des lasers de découpe [9].

### **II.2.3 Programmation de la machine à commande numérique**

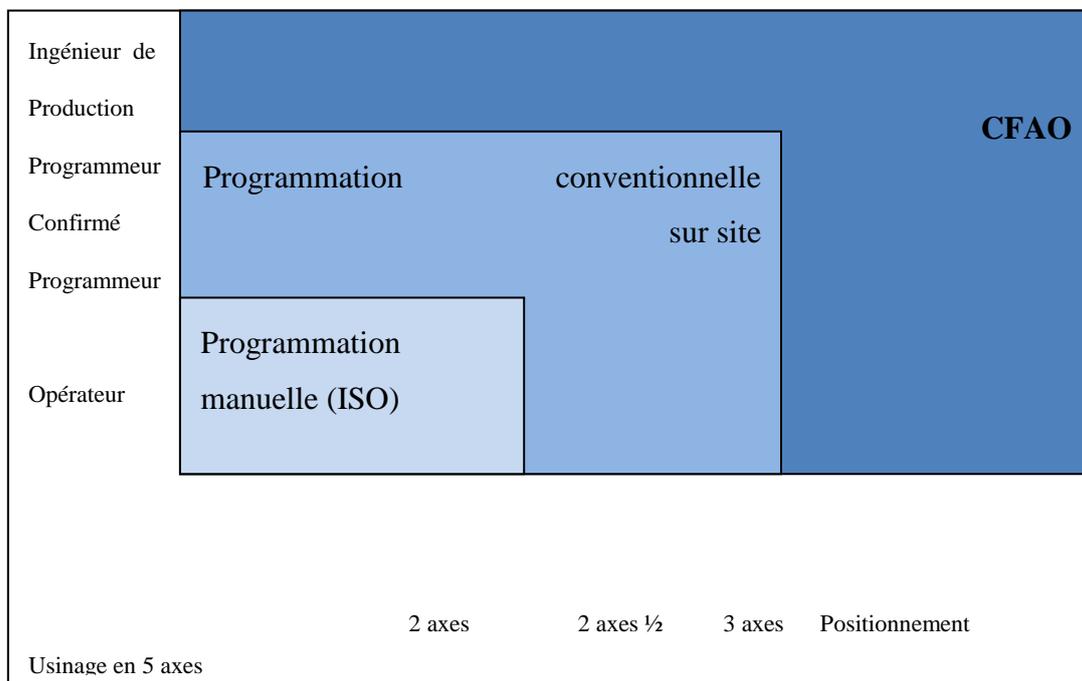
La programmation de commande numérique (CN) permet de spécifier et de définir des séquences d'instructions qui permettent de guider et de piloter des machines-outils à commande numérique. Cette programmation est actuellement fortement automatisée à partir de plans réalisés en CAO. [8]

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 numérotation n'intervient pas dans l'ordre de déroulement du programme donc il est conseillé de les numéroter dans l'ordre d'écriture [8].

Un programme est une transcription dans un langage compréhensible par le Directeur de Commande Numérique(DCN) d'une machine, une pièce sera exécutée par ses opérations d'usinage.

On peut programmer de différentes manières :

- La programmation manuelle,
- La programmation assistée :
  - Soit conversationnelle par le DCN
  - Soit avec un logiciel de F.A .O (fabrication assistée par ordinateur).



**Figure II.1** : Classification des méthodes de programmation en fonction des compétences du programmeur [5].

### II.3 Elaboration d'un programme

Il est considérable de faire soigneusement une organisation et une planification des opérations d'usinages. Plus notre préparation aura été explicite plus la programmation sera direct et rapide et moins nous aurons d'erreurs dans le programme terminé .

Les principales étapes de la création sont :

- Définir le déroulement des opérations d'usinage.
- Définir la gamme de fabrication.
- Traduire les opérations dans le langage de programmation.

- Regrouper toutes les opérations en un programme.

#### II.4 Structure d'un programme

Un programme CN se compose d'une suite de blocs CN. Chaque bloc contient les données pour l'exécution d'une opération d'usinage. Il est divisé en 3 domaines : Entête de programme, corps de programme et fin de programme.

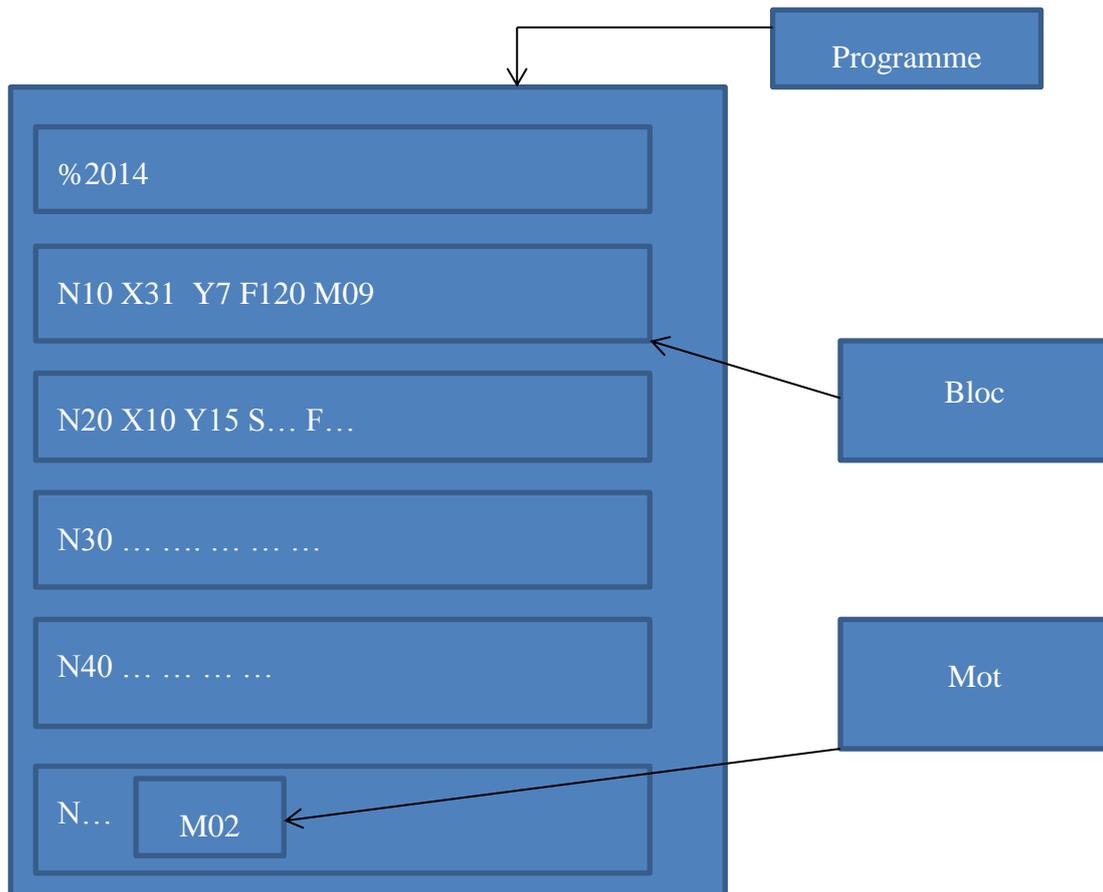
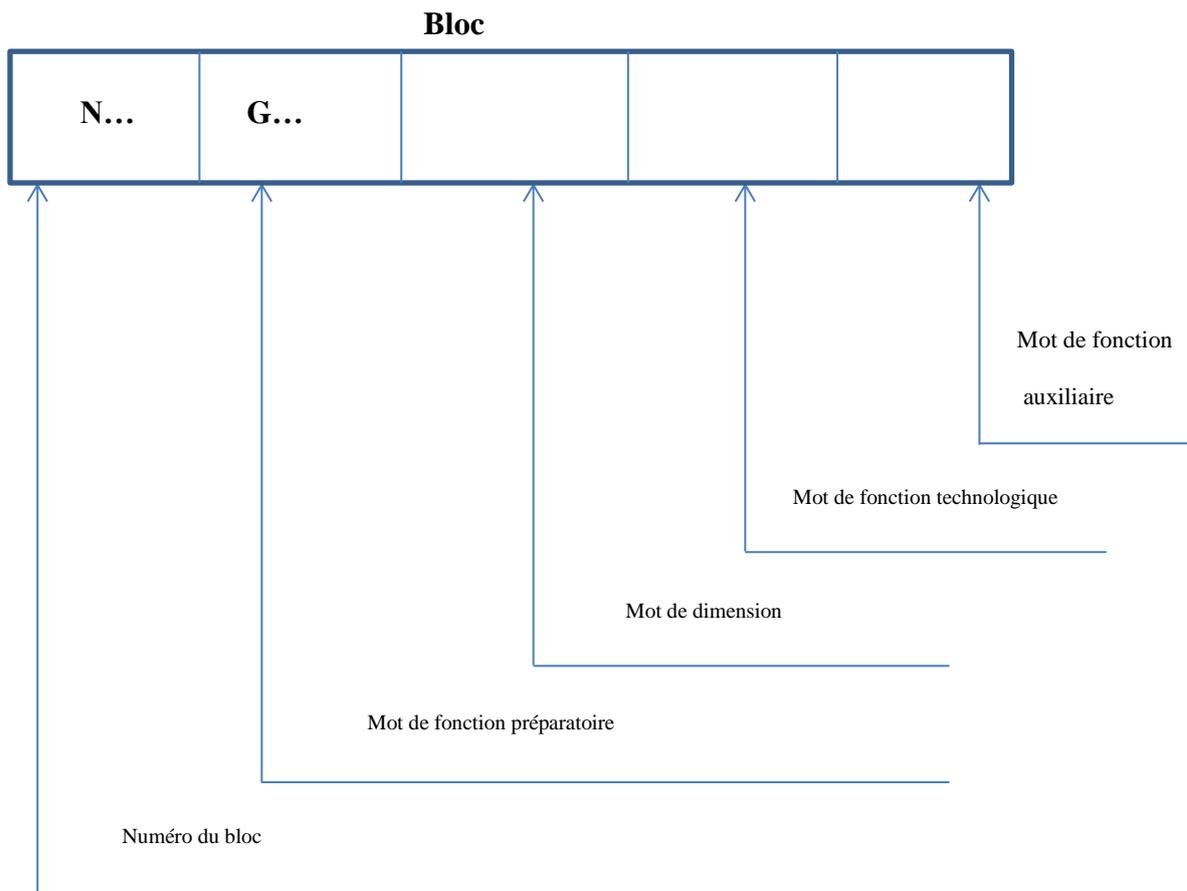


Figure II.2 : structure d'un programme

##### II.4.1 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 [8].

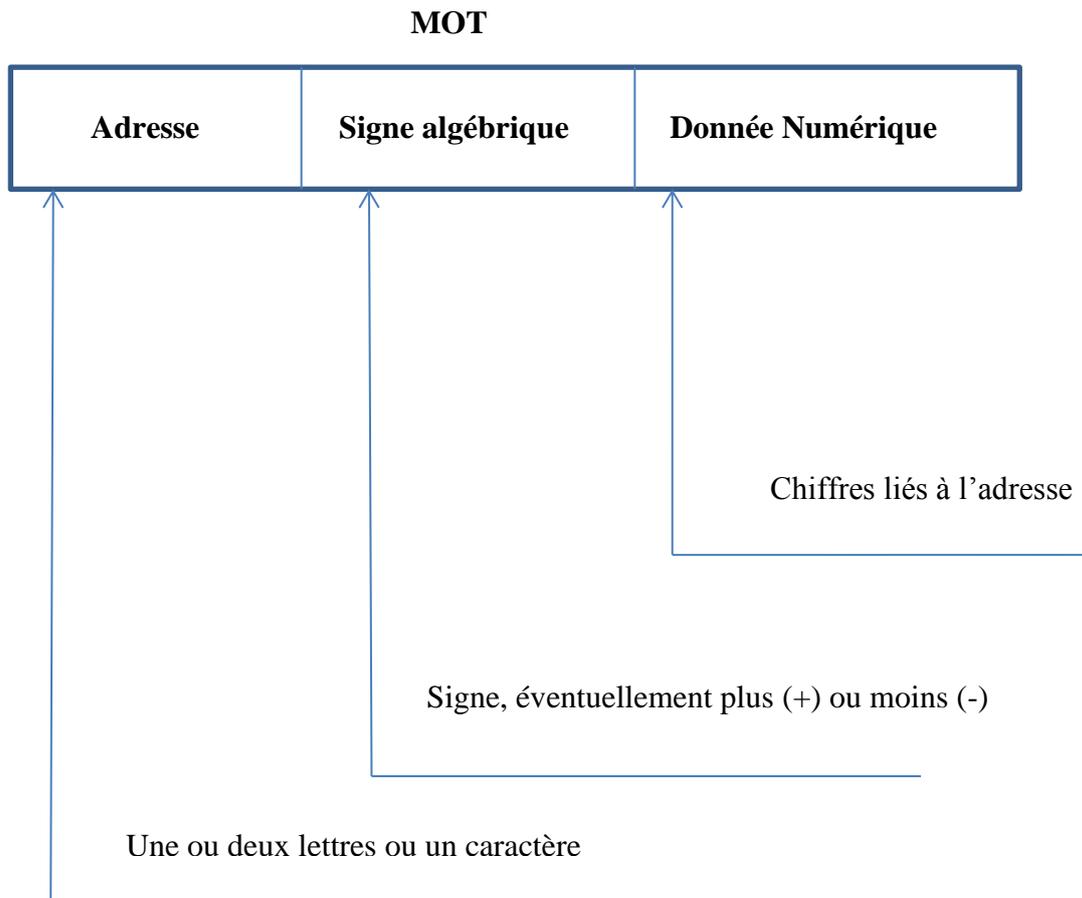


**Figure II.3 :Format d'un bloc**

#### **II.4.2 Format d'un mot**

Une instruction ou une donnée à transmettre au système de commande est défini par un mot. Il existe deux types de mots :

- ❖ Mots définissant des dimensions
- ❖ Mots définissant des fonctions



**Figure II.4 : Format d'un mot**

### II.4.3 Différents mots

Un mot est composé d'une lettre et d'une valeur numérique, entière ou décimale. Dans un bloc, il peut y avoir plusieurs mots. Il est fréquent d'avoir une erreur si plusieurs mots du groupe de fonctions auxiliaires sont entrés sur le même bloc [10].

- ❖ Fonctions préparatoires (G), fonctions d'appel de mode d'interpolation (G0, ...), cycles machine.

❖ **Coordonnées de points (X, Y, Z, I, J, K, A, B, C, U, V, W)****Tableau1** : Signification des fonctions des déplacements

<b>La lettre</b>	<b>La fonction</b>
<b>X</b>	position absolue/relative
<b>Y</b>	position absolue/relative
<b>Z</b>	position absolue/relative
<b>I</b>	coordonnée de centre de cercle sur X
<b>J</b>	coordonnée de centre de cercle sur Y
<b>K</b>	coordonnée de centre de cercle sur Z
<b>A</b>	position angulaire (rotation autour de l'axe X)
<b>B</b>	position angulaire (rotation autour de l'axe Y)
<b>C</b>	position angulaire (rotation autour de l'axe Z)
<b>U</b>	position relative en tournage
<b>V</b>	position relative en tournage
<b>W</b>	position relative en tournage

❖ **Vitesses, avances, outils, ... (S, F, T, ...)****Tableau2** : paramètres d'usinage

<b>La lettre</b>	<b>La fonction</b>
<b>S</b>	vitesse de rotation de la broche ou vitesse de coupe
<b>F</b>	avance de travail en millimètres/minute ou en millimètres/tour

<b>T</b>	sélection d'un outil
<b>H</b>	numéro du correcteur de longueur (fraisage)
<b>D</b>	numéro du correcteur de rayon (fraisage)
<b>N</b>	numéro de bloc

**Remarque :**

Le numéro de bloc n'est pas obligatoire à chaque ligne avec les commandes ISO Fanuc, il désigne juste un endroit du programme où l'on peut ordonner un saut d'opération ou un départ d'opération, le plus souvent avant un changement d'outil.

- Fonctions auxiliaires (M) qui permettent d'enclencher la lubrification, de changer d'outil, ou de déclencher des accessoires. Permet aussi de contrôler des entités externes à la machine proprement dit (magasin auxiliaire, refroidisseur, compteur ETC).
- Des machines différentes peuvent utiliser le même code pour effectuer des fonctions différentes: même les machines qui utilisent le même contrôleur CNC. Certains ont déclaré que le G-code "Fanum" est standard, ce qui est faux. C'est simplement l'un des plus simples et des plus répandu.

**II.4.4 Format d'un programme**

Afin de réaliser un programme on suit les étapes suivantes:

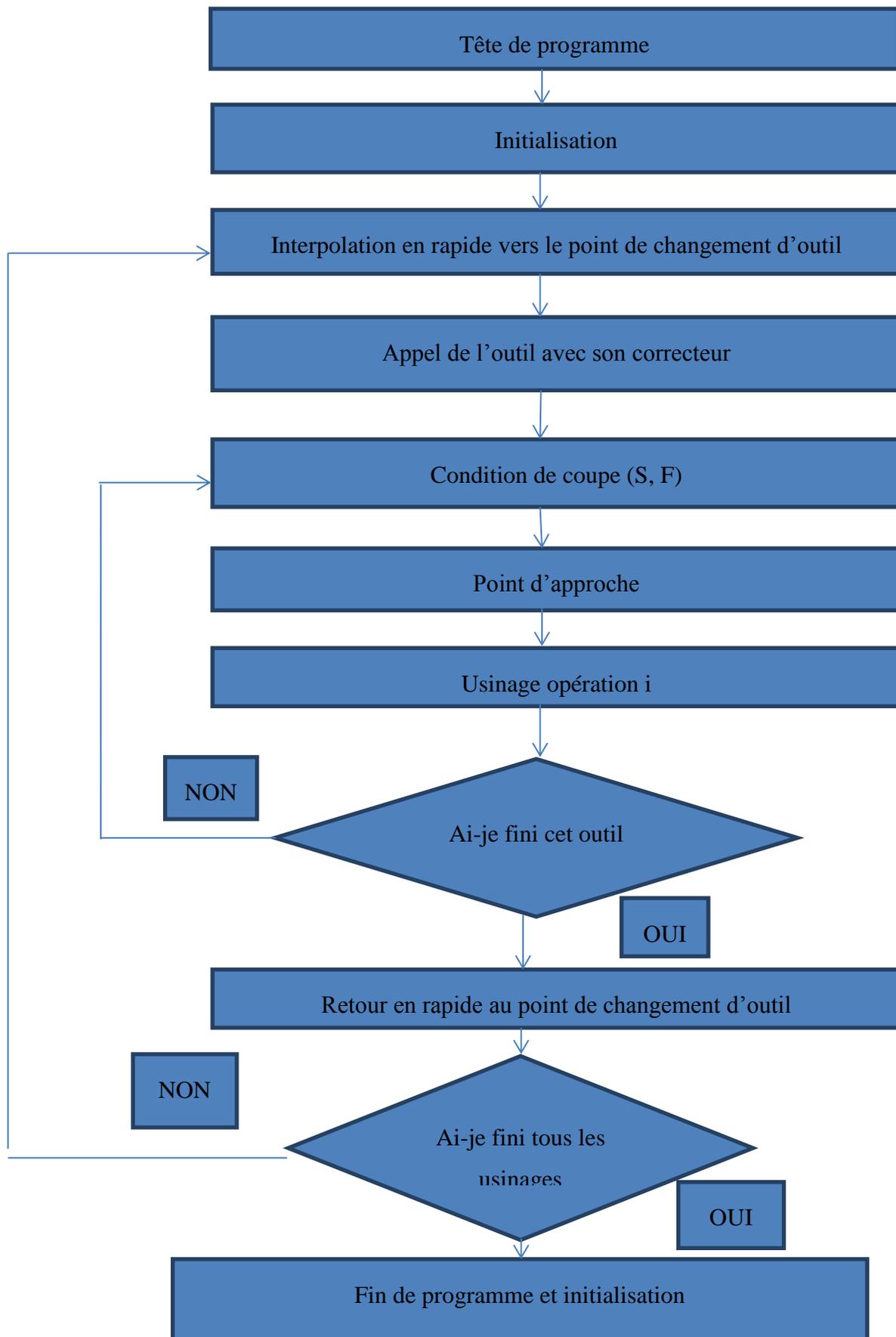


Figure II.5 : diagramme d'un programme [5].

## II.5 Instructions de bases

Dans le langage Gcode certaines fonctions telle que G00, G01, G94 ..., ont une importance capitale du fait qu'elles sont présentes pratiquement dans chaque programme de ce fait, on s'est intéressé [11,12]

### II.5.1 La fonction G00

- Le point programmé est atteint en effectuant une trajectoire linéaire à vitesse rapide.
- La trajectoire est la résultante de tous les déplacements d'axes programmés dans le bloc.
  - Axes primaires X, Y, Z
  - Axes secondaires U, V, W
  - Axes rotatifs A, B, C
- Interpolation linéaire, positionnement rapide. La commande G00 permet le déplacement de l'outil dans le système de pièce jusqu'à la position spécifiée grâce à une commande incrémentielle ou absolue à une vitesse transversale rapide.

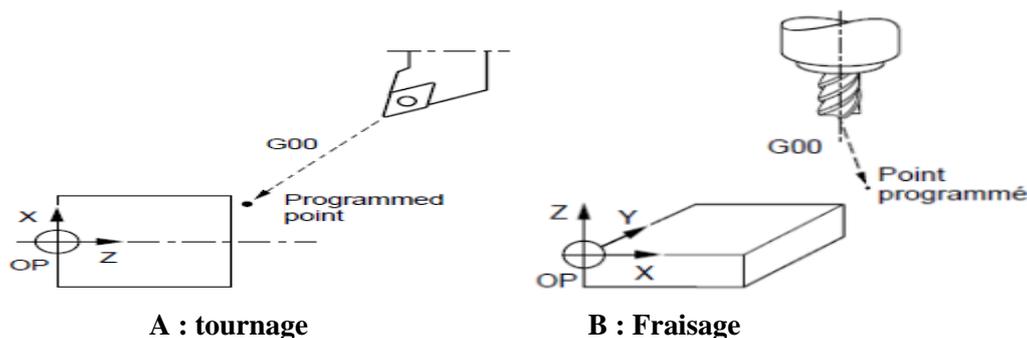
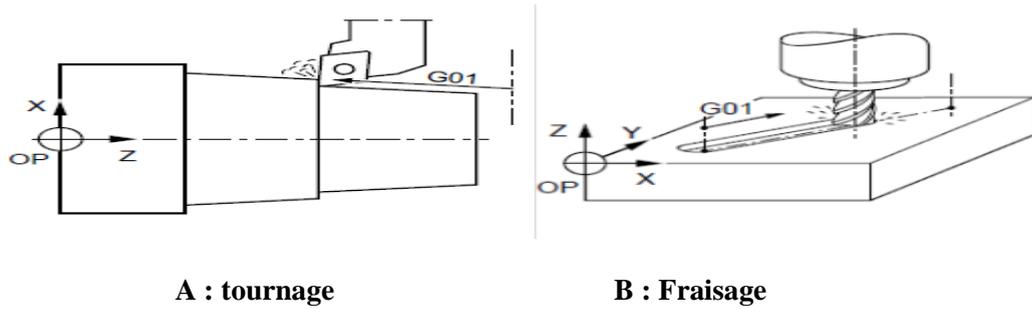


Figure II.6:Fonction G00 [11]

### II.5.2 La fonction G01

- Le point programmé est atteint en effectuant une trajectoire linéaire à vitesse d'avance programmée.
- La trajectoire est la résultante de tous les déplacements d'axes programmés dans le bloc.
  - Axes primaires X, Y, Z
  - Axes secondaires U, V, W
  - Axes rotatifs A, B, C
- Interpolation linéaire (avance de coupe) .Le déplacement de l'outil est suivant une ligne droite le long d'une pièce jusqu'à la position précise à la vitesse d'avance spécifiée dans F. il n'est pas nécessaire de programmer l'avance dans chaque bloc. Au long de la trajectoire de l'outil l'avance F programmée est mesurée. Si F n'est pas programmée, l'avance est considérée comme étant 0.L'avance suivant chaque axe est calculée.



A : tournage

B : Fraisage

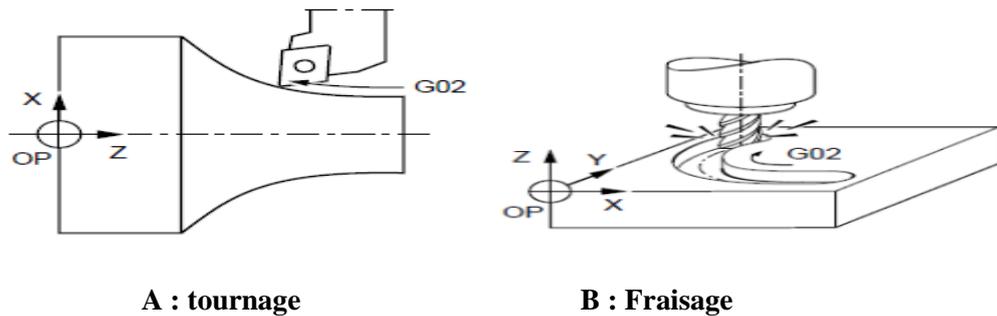
Figure II.7:Fonction G01 [11]

### II.5.3 Les fonctions G02 et G03

L'interpolation circulaire permet de vérifier et d'examiner à chaque instant la position de l'outil pendant l'usinage des arcs de cercles dans le plan.

**G02** : interpolation circulaire sens horaire à vitesse d'avance programmée.

- La position du point programmée est atteinte en décrivant une trajectoire circulaire.



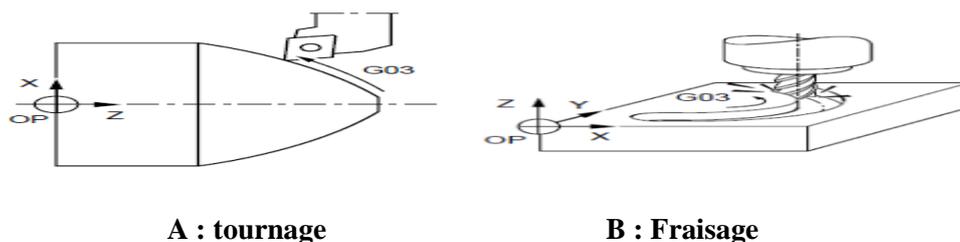
A : tournage

B : Fraisage

Figure II.8:Fonction G02[11]

**G03** : interpolation circulaire sens antihoraire à vitesse d'avance programmée.

- La position du point programmée est atteinte en décrivant une trajectoire circulaire.



A : tournage

B : Fraisage

Figure II.9:Fonction G03 [11]

### II.5.4 La fonction G40

- Compensation d'outil de coupe ou de rayon de pointe d'outil : Annulation
- Compensation d'outil de coupe tridimensionnelle : Annulation
- Le pilotage du point de la coupe théorique de l'outil. La correction de rayon n'est plus appliquée à l'outil. (en tournage)
- Le pilotage du centre de l'outil : les trajectoires programmées sont appliquées au centre de l'outil. (en fraisage)

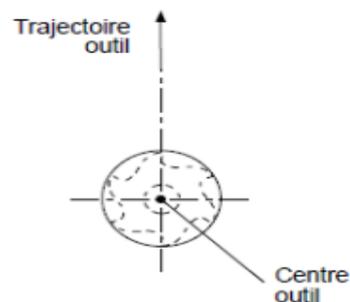


Figure II.10 : Fonction G40 [11]

### II.5.5 La fonction G41

Correction de rayon à gauche du profil à usiner.

- les trajectoires outil programmées sont corrigées (décalées à gauche) d'une valeur égale au rayon d'outil (R) déclaré par le correcteur D.

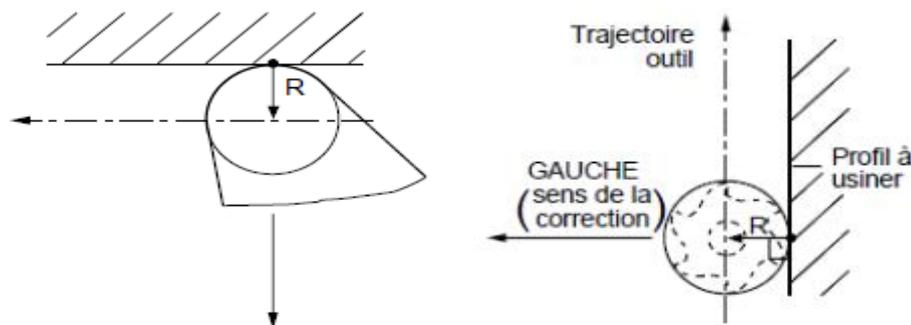


Figure II.11 : Fonction G41 [13]

### II.5.6 La fonction G42

Correction de rayon à droite du profil à usiner.

- Les trajectoires outil programmées sont corrigées (décalées à droite) d'une valeur égale au rayon d'outil (R) déclaré par le correcteur D.

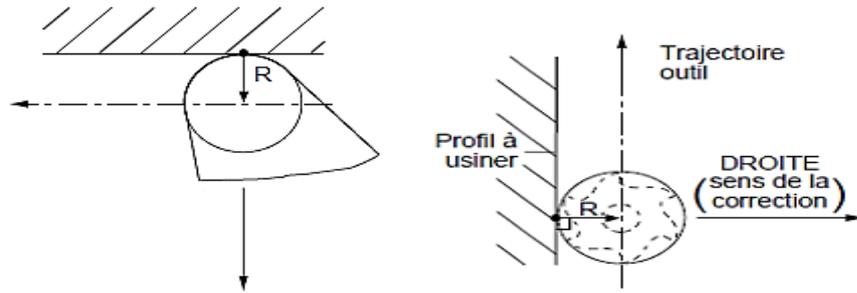


Figure II.12: **Fonction G42** [11]

**II.5.7 La fonction G80**

Annulation de cycle d’usinage.

- La fonction permet la révocation de cycles d’usinage.

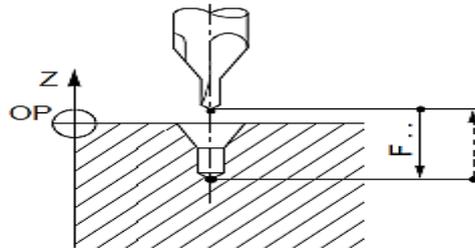


Figure II.13 : **Fonction G80** [11]

**II.5.8 Les fonctions G90 et G91**

On distingue deux types de commandes de déplacement de l’outil : les commandes absolues et les commandes relatives. Les deux codes sont utilisés pour sélectionner le mode absolu et relatif respectivement.

**G90** : programmation absolue par rapport à l’origine programme.

- les valeurs des coordonnées de la position d’arrivée sont programmées.

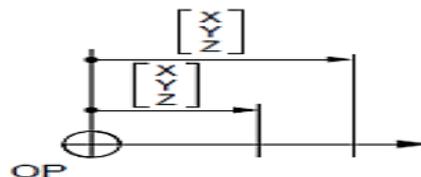


Figure II.14: **Fonction G90** [11]

**G91** : programmation relative par rapport au point de départ du bloc.

- La valeur programmée sur un axe est repérée par rapport à la dernière position programmée.
- La valeur est égale au déplacement à réaliser.

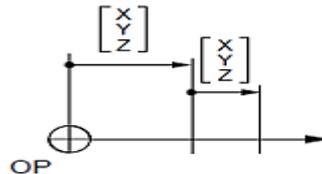


Figure II.15: Fonction G91 [11]

### II.5.9 La fonction G92

Présélection de l'origine programme.

- La fonction affectée d'un ou plusieurs axes et leurs valeurs définit la position courante du mobile par rapport à la nouvelle origine programme.
- Définition du système de coordonnées pièce ou limitation de la vitesse maximale de la broche.
- ces commandes peuvent être utilisées sans paramètres additionnels, permet de définir les nouvelles positions des axes (X, Y, Z) et la programmation du point zéro absolu, en réinitialisant la position actuelle aux valeurs spécifiées.

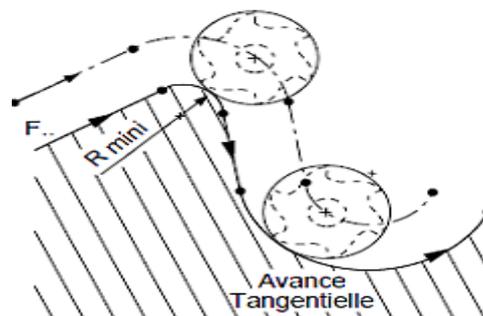


Figure II.16: Fonction G90 [11]

### II.5.10 La fonction G94

Vitesse d'avance exprimée en millimètre pouce ou degré par minute

- la vitesse d'avance s'exprime en millimètre ou pouce par minute sur les axes linéaires, en degré/ minute sur les axes rotatifs programmés seuls.
- Avance par minute, la valeur de l'avance par minute est programmée et spécifiée avec le code F . le G94 est un code modal, il reste actif jusqu'à ce que le G95 soit programmé.

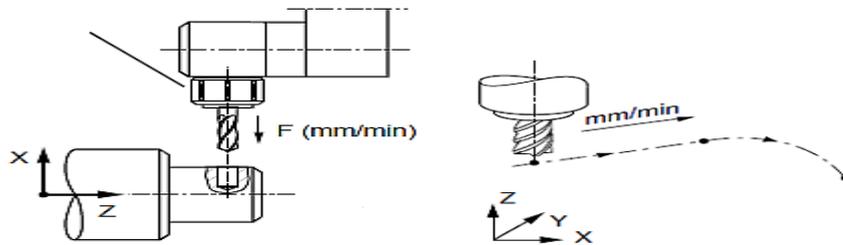


Figure II.17: Fonction G94 [11]

### II.5.11 La fonction G95

Vitesse d'avance exprimée en millimètre ou en pouce par tour

- La vitesse d'avance s'exprime en millimètre ou pouce par tour de broche.
- Avance par tour, la valeur de l'avance de l'outil par tour de broche est programmé et spécifier avec le code F. le G95 est un code modal, il reste actif jusqu'à ce que le G94 soit programmé.

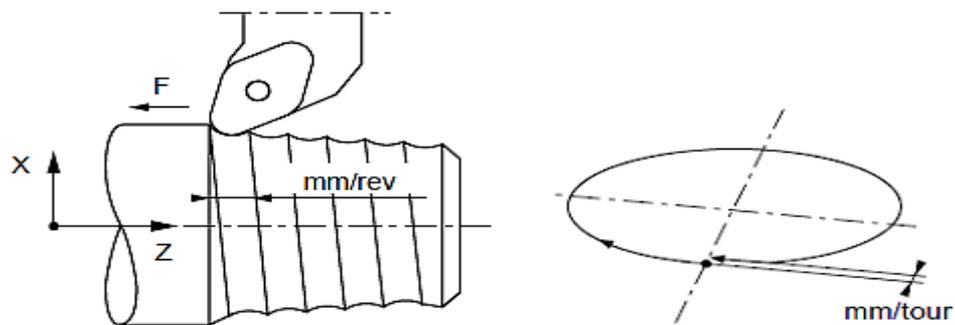


Figure II.18:Fonction G95 [11]

### II.5.12 La fonction M00

La fonction provoque un arrêt dans le déroulement du programme en cours.

Après intervention ou contrôle, le cycle est relancé par l'opérateur.

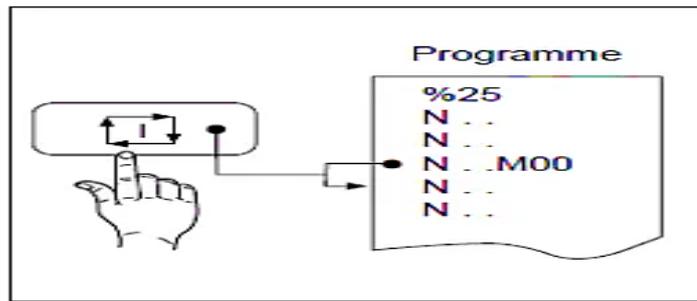


Figure II.19: Fonction M00 [11]

### II.5.13 La fonction M01

La lecture de M01 validé par l'opérateur provoque un arrêt dans le déroulement du programme.

Après intervention ou contrôle, le cycle est relancé par l'opérateur.

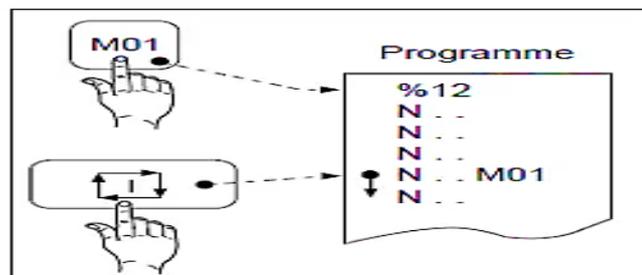


Figure II.20: Fonction M01 [11]

### II.5.14 La fonction M02

La fonction M02 termine un programme

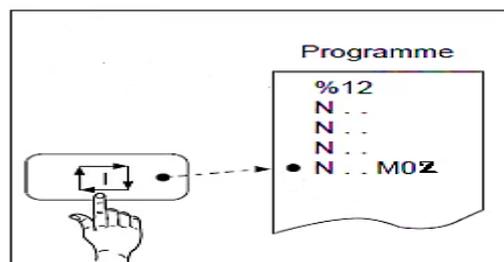


Figure II.21: Fonction M02 [11]

### II.5.15 La fonction M03

Rotation de la broche sens horaire.

La commande permet la mise en rotation de la broche à la vitesse programmée.

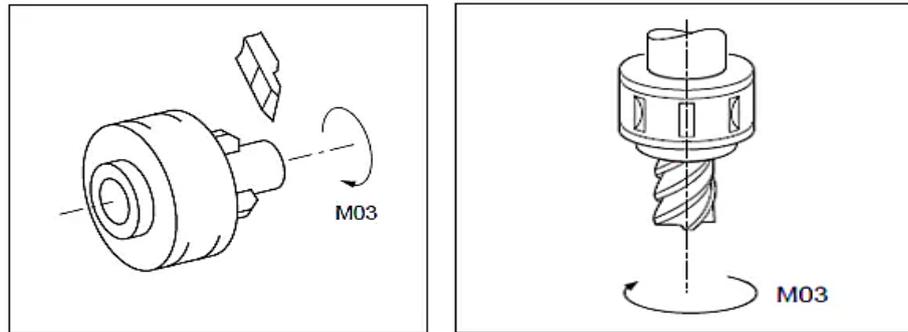


Figure II.22: Fonction M03 [11]

### II.5.16 La fonction M04

Rotation de la broche sens anti horaire.

La commande permet la mise en rotation de la broche à la vitesse programmée.

### II.6 Conclusion

Dans ce chapitre on s'est intéressé au système de programmation des machines-outils à commande numériques à l'aide du langage de programmation Gcode et à la manière de concevoir ce programme en mettant en avant les fonctions les plus utilisés de façon général dans tous les type de MOCN par la suite, dans le chapitre suivant on va présenter plus précisément notre machine -outil qui sera le sujet de notre simulateur.

**Chapitre III :**  
**Présentation des MOCN et réalisation la**  
**maquette**

### **III.1 Introduction**

Dans les deux chapitres précédents on a présenté les MOCN en général ainsi que leur programmation afin de mieux se familiariser et cerner leur principe de fonctionnement, chose indispensable afin de réaliser un simulateur et donc dans ce chapitre on va s'intéresser plus précisément à la machine qui sera le sujet de notre simulateur et pour cela on va commencer par présenter un aperçu général ainsi que son principe de fonctionnement et ces différents composants, puis après on va réaliser son modèle en 3D avec le logiciel Solidworks qu'on aura préalablement présenté en précisant les différentes étapes de la conception et l'assemblage de ce modèle numérique, modèle qui sera le support visuel de notre simulateur.

### **III.2 Présentation de notre MOCN**

#### **III.2.1 Aperçu de notre MOCN**

Notre machine sera une MOCN à trois axes (X,Y,Z), la structure de la machine sera en aluminium avec des dimensions et toutes ses caractéristiques bien détaillées dans un tableau des annexes. Elle sera alimentée avec une source de 12V et 20A, elle sera aussi équipée de trois fins de courses et de 4 moteurs pas à pas, un moteur pour le déplacement selon l'axe X, un autre selon l'axe Z et deux pour le déplacement selon l'axe Y, le choix de maître deux moteurs pour l'axe Y est un choix stratégique qui permet de mieux contrôler le déplacement de cette axe ainsi que de renforcer sa stabilité.

L'outil qu'on a utilisé pour notre CNC est disponible dans le marché au complet avec un moteur intégré mais il ne dispose pas d'un magasin de changement d'outil et donc cette étape se fera manuellement.

#### **III.2.2 Description des différents composants de la MOCN**

La réalisation de notre machine à commande numérique a nécessité l'intégration de différents éléments choisis soigneusement afin de renforcer le bon fonctionnement et le déroulement des opérations.

##### **III.2.2.1 Moteur pas à pas**

Les moteurs pas à pas sont des transducteurs électromécaniques qui assurent la conversion des signaux électriques digitaux ou impulsifs en mouvement de rotation ou de translation de type incrémental.

Les moteurs pas à pas, utilisés pratiquement dans tous les composants d'un système informatique, sont très mal connus, surtout en ce qui concerne leur mode de commande. Ils sont utilisés aussi bien dans les lecteurs de disquettes, les disques durs, que dans les scanner et les imprimantes 3D conçue pour prototype rapide.

Le moteur pas à pas est actuellement le principal élément intermédiaire entre les dispositifs de traitement d'information et le mode électromécanique extérieur. Par ailleurs, ses capacités à contrôler la position et la vitesse, par un train d'impulsion de commande assurent à ce convertisseur des applications comme [13,14] :

- La traction des robots mobiles,
- Le fonctionnement en moteur couple de garde puissance,
- L'indexage rotatif ou linéaire.

Dans sa version classique, le moteur pas à pas est alimenté à partir d'une source (de courant ou de tension) continue et le contrôle de la vitesse ou / et de la position s'effectue en boucle ouverte.

Le pilotage en boucle ouverte, qui constitue un des principaux avantages du moteur pas à pas, aussi bien du point de vue économique (cout et fiabilité d'installation) que fonctionnel (commande naturelle par « tout ou rien » à partir d'une horloge séparé ou intégrée dans un (microprocesseur), présente un certain nombre d'inconvénients comme :

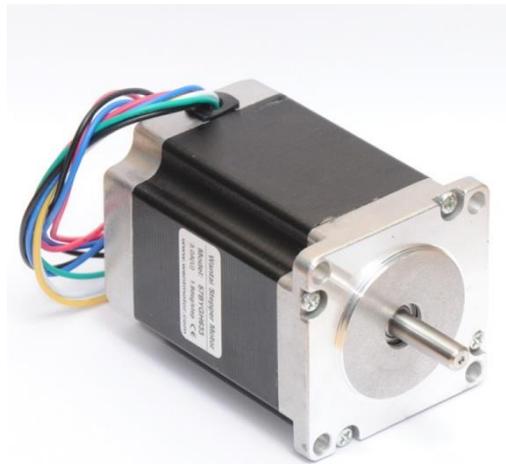
- La limitation du couple de démarrage,
- Des instabilités de fonctionnement à certaines fréquences,
- Des accélérations relativement modestes,

Le moteur pas à pas est un convertisseur électromécanique qui assure la transformation d'un signal électrique impulsionnel en un déplacement mécanique (angulaire ou linaire). Sa structure de base se présente sous la forme de deux pièces séparées mécaniquement, le Stator et le Rotor. L'interaction électromagnétique entre ces deux parties assure la rotation [8].

- les moteurs qu'on a utilisés dans notre MOCN ont les caractéristiques suivantes:

**Tableau3** : caractéristiques des moteurs pas à pas de notre MOCN

Vitesse de pas	1.8°
Longueur du moteur	76 mm
Profil de l'arbre	Rond
Diamètre de l'arbre	6.35mm
Tension nominale	3v
Courant nominal	1 A
La résistance de phase	1Ω
Couple de maintien	19 kg.cm
Poids du moteur	1030g

**Figure III.1:** Moteur pas à pas

### III.2.2.2 Commutateur de fin de courses

Ils servent de mécanisme qui nous indique les limites de déplacement des axes de la MOCN. Ils sont généralement placés aux extrémités.

Lorsque l'un des axes se déplace à une limite, l'intercepteur s'active et la machine s'arrête.



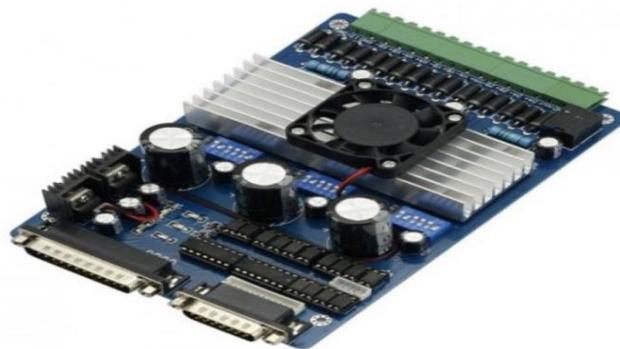


**Figure III.3 : Carte Arduino**

Il existe plusieurs types de carte Arduino mais dans notre cas on a utilisé une carte Arduino Mega qui possède un microcontrôleur ATmega1280 pour I/O additionnel et mémoire.

#### **III.2.2.4 Carte pilote "drivers" pour moteur pas-à-pas**

Les pilotes des moteurs pas-à-pas sont des modules qui permettent de contrôler un moteur en fonction des instructions reçues de la carte de pilotage (dans notre cas la carte arduino), en d'autre termes c'est une carte auxiliaire intermédiaire entre le système de contrôle et les moteurs, il est important de faire attention au voltage et à l'ampérage et donc à l'impulsion fournie par le moteur afin d'éviter toutes dégradations du matériel, ils peuvent être utilisés pour des applications multiples telles que les imprimantes 3D, les CNC, les robots et les machines à étiquettes.



**Figure III.4: Carte Pilote**

### II.2.2.5 Carte shield

C'est une carte électronique qui se connecte dans soudage sur une carte Arduino et qui lui permis d'augmenter ces performances et d'élargir ses capacités et ces fonctions, il existe plusieurs type pour différentes applications de carte shield telle que : afficheur graphique, ethernet et carte SD, GPS, carte de prototypage.



**Figure III.5:** Carte shield

### III.3 Principe de fonctionnement de notre MOCN

Le déplacement selon les 3 axes se fait grâce à l'actions de 4 moteurs pas-à-pas, un pour l'axe X , un autre pour l'axes Z et deux pour l'axes Y, les moteurs sont commandés avec une carte arduino qui est programmée avec le langage C, ils sont donc relié à cette dernière par l'intermédiaire d'une carte driver, le mouvement de cette MOCN est un mouvement rectiligne crée par l'interaction entre les vis sans fin et les 4 roues, qui sont elles-même entrainées par l'arbres des moteur pas-à-pas.

### III.4 la maquette numérique

#### III.4.1 La CAO

##### III.4.1.1 Définition

La CAO peut être définie comme étant l'ensemble des outils logiciels et les techniques informatiques qui permettent d'aider le concepteur dans la conception et la mise au point d'un produit. Un système de CAO permet de représenter et d'étudier le fonctionnement d'un objet sans l'avoir fabriquer réellement, c'est-à-dire en mode virtuel [15].

##### III.4.1.2 Historique que de CAO

La CAO est née aux Etats-Unis à la fin des années 1950, quand Général(Motors) et le (Massachusetts Institute of technology) ont imaginé de dialoguer avec un ordinateur autrement qu'avec des cartes perforées, des bandes magnétiques ou des rubans de papier perforé.

Vers les années 1965, les premières idées d'utilisation de ce nouvel outil ont été de s'en servir pour la définition et le contrôle de pièces de carrosseries.

Vers les années 1970, un certain nombre de travaux portant sur les techniques de représentation et de manipulation de formes complexes, dans lesquelles l'objet est constitué de surfaces, ont abouti.

Enfin, au début des années 1980, les bureaux d'études mécanique ont commencé à s'équiper de moyens de CAO, là encore grâce à une avancée notable des logiciels, du matériel et des prix : en effet, il a été possible d'utiliser de nouveaux objets mathématiques, les solides, permettant l'accès aux propriétés de masse des pièces modélisées (centre de gravité, moment d'inertie, etc...) qui intéressent fortement le mécanicien [16].

#### **II.4.1.3 Pourquoi la CAO**

Le but de la CAO est de créer un modèle en 2D ou bien 3D afin d'avoir une vue globale de la structure de l'objet en question et d'étudier son comportement, son fonctionnement avant qu'il soit réalisé, en pratique tous ou presque peut être pris en compte lors de la conception de la maquette que ça soit le poids, la taille ou le type de matériaux dont l'objet sera construit et dans le cas d'une structure complexe formée par plusieurs pièces, un assemblage peut être réalisé pour aboutir la forme voulu.

### **III.4.2 Logiciel SolidWorks**

#### **III.4.2.1 Définition**

Le logiciel de conception mécanique SolidWorks est un outil de conception de modélisation volumique paramétré, basé sur des fonctions, qui tire parti des fonctionnalités de Windows, connu pour sa convivialité. Vous pouvez créer des modèles volumiques 3D entièrement intégrés avec ou sans contraintes tout en utilisant des relations automatiques ou définies par l'utilisateur pour saisir l'intention de conception [17].

#### **III.4.2.2 Pourquoi SolidWorks**

On a opté pour ce logiciel d'une part pour sa simplicité, ses nombreux fonctions améliorés tous au long de ces dernières années mais aussi et surtout parce qu'il va nous permettre après de faire le lien avec le logiciel Matlab dans lequel on va créer un espace de simulation pour notre MOCN.

### III.4.3 Réalisation de la maquette

Dans tout simulateur la partie visualisation est sans doute la plus importante car c'est dans cette partie qu'on va voir le déroulement du processus de fraisage, tournage de notre MOCN et pour cela on a décomposé cette partie (réalisation de la maquette) du travail en trois étapes fondamentales qui sont les suivantes :

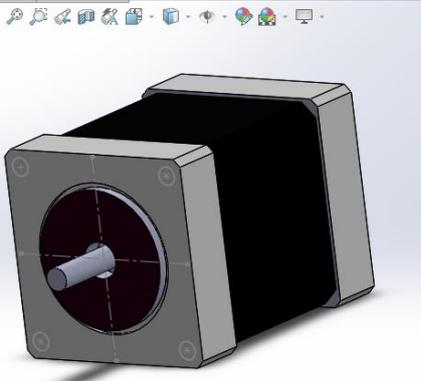
- L'étape de la réalisation des différentes pièces qui composent la machine ;
- L'étape de la réalisation des sous-assemblages qui représentent les 3 axes de déplacement ainsi que la partie immobile de la machine ;
- L'étape de la réalisation de l'assemblage final qui respecte les règles de l'export-import sur le logiciel de simulation Matlab.

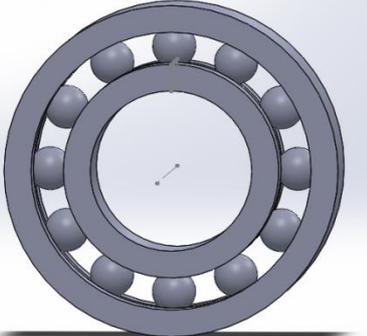
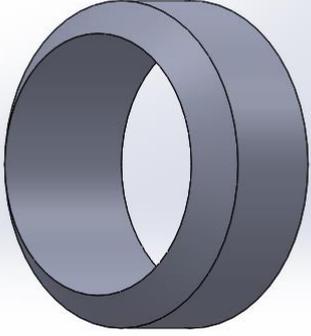
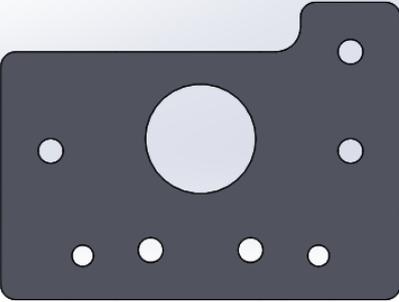
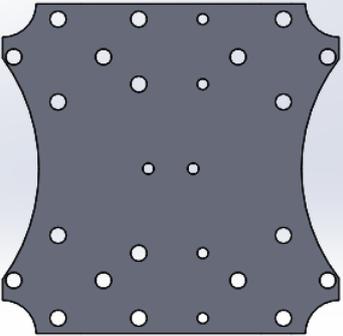
#### III.4.3.1 Etape de la réalisation des différentes pièces

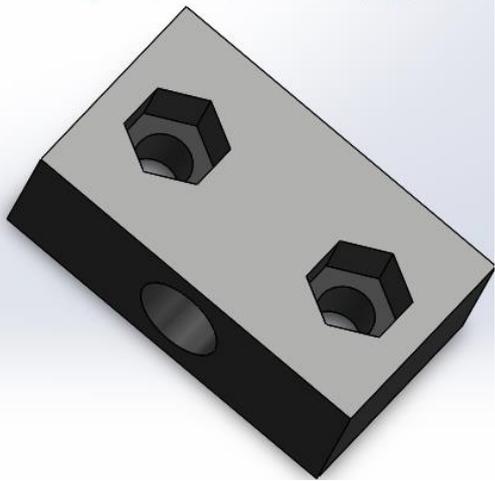
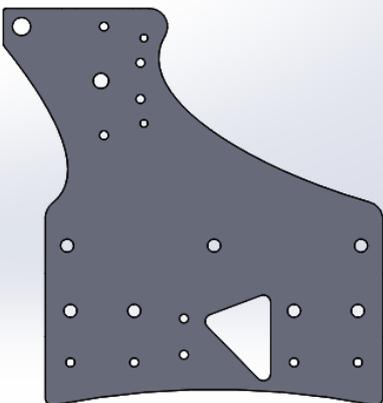
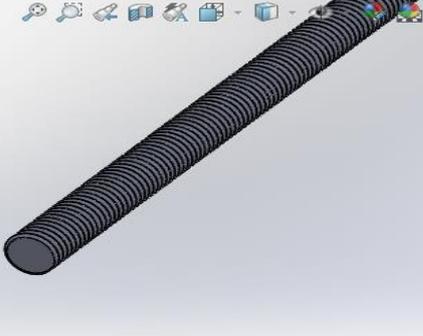
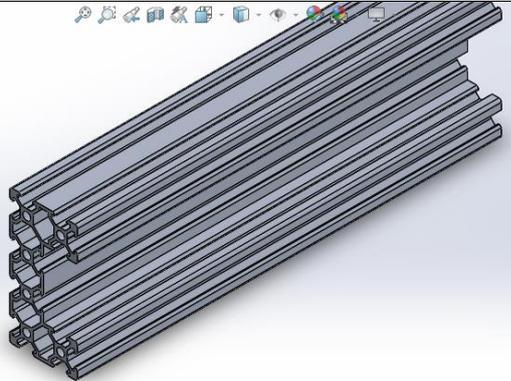
Notre MOCN se compose de plusieurs pièces avec pour chaque pièce un rôle bien définie et des dimensions à respecter afin d'éviter tout éventuelle problèmes lors de l'assemblage. Comme nous l'avant déjà définie auparavant, on a conçu notre maquette dans le logiciel SolidWorks et pour cela on a décomposé notre machine en plusieurs partie bien distinctes afin de facilité leur conception puis après on a réalisé l'assemblage de l'ensemble des pièces en utilisant des composant déjà existant dans la bibliothèque de SolidWorks.

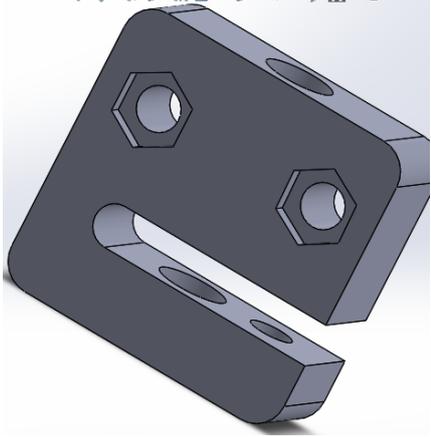
Dans ce tableau on a résumé les principales pièces réalisées dans SolidWorks :

**Tableau4** : les pièces qui composent le bâti de notre MOCN

Pièce	Nom	Fonction
	<p>Moteur pas à pas</p> <p>Remarque: en réalité ce moteur est lui-même un assemblage</p>	<p>C'est l'actionneur qui fait translater les 3 axes (X, Y, Z).</p>

	<p>Roulement à bille</p>	<p>Permet de réaliser de mouvement de rotation.</p>
	<p>Roue</p>	<p>Assemblé avec le roulement, elle permet de faire les mouvements.</p>
	<p>Pièce (1)</p>	<p>Élément qui fait partie du bâti</p>
	<p>Pièce (2)</p>	<p>Élément qui fait partie du bâti</p>

	<p>Pièce (3)</p>	<p>Élément qui fait partie du bâti</p>
	<p>Pièce (4)</p>	<p>Élément qui fait partie du bâti</p>
	<p>Tige fileté</p>	<p>Élément qui fait partie du bâti</p>
	<p>Barre</p>	<p>Élément qui fait partie du bâti</p>

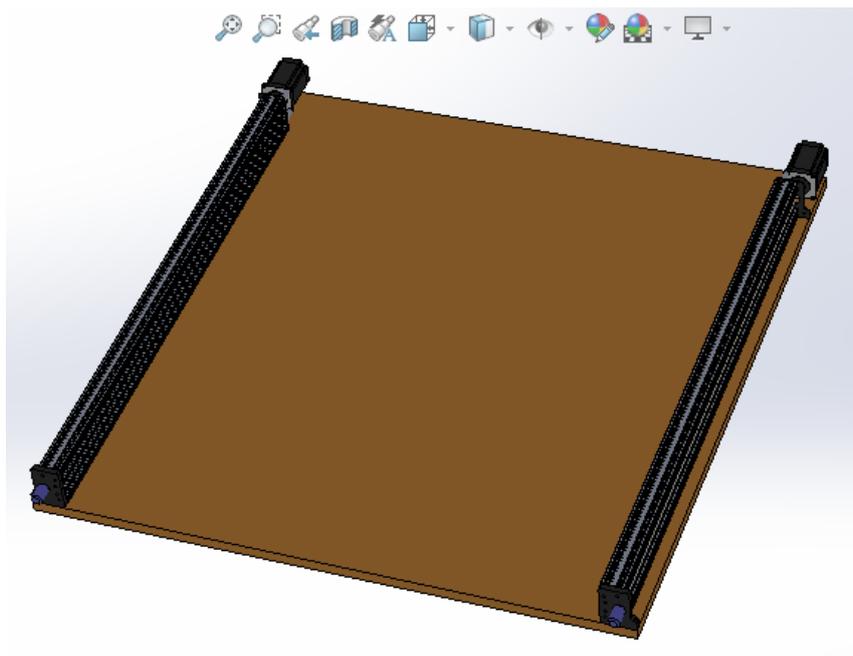
	Pièce (5)	Elément qui fait partie du bâti
---	-----------	---------------------------------

### III.4.3.2 Etape de la réalisation des sous-assemblages

Afin de préparer l'importation de la machine dans le logiciel de simulation il est judicieux d'assembler notre MOCN en sous-assemblages avant de réaliser l'assemblage final et pour cela on l'a décomposé en 4 parties qui sont les suivantes:

#### III.4.3.2.1 la table

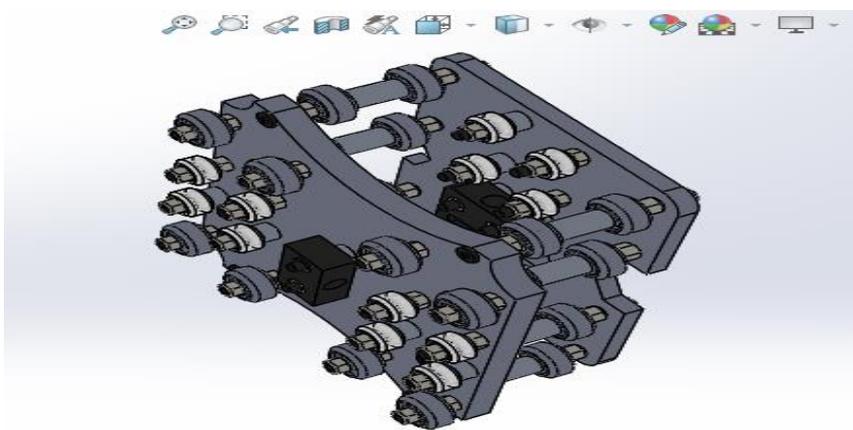
C'est la partie immobile de la machine, elle comporte une planche en bois de 1 mètre carrée et deux barres de un mètre chacune qui seront les supports pour le déplacement de l'axe Y, ce sous-assemblage comportera aussi une tige filetée qui permettra par la suite de l'assembler avec une autre pièce de l'axe Y.



**Figure III.6:** La table

#### III.4.3.2.2 l'axe X

Pour le déplacement selon l'axe X on a utilisé deux pièces préalablement assemblés avec des roues afin de réaliser une translation fluide et de deux pièces (3) et (5) qui vont permettre par la suite d'introduire les deux tiges filetées, une pour l'axe X et l'autre pour l'axe Z.



**Figure III.7:** Axe X

### III.4.3.2.3 l'axe Y

Il se compose d'une tige raccordée à deux pièces à chaque extrémités et qui sont-elles mêmes assemblées avec des roues pour permettre le mouvement de translation de l'axe Y.

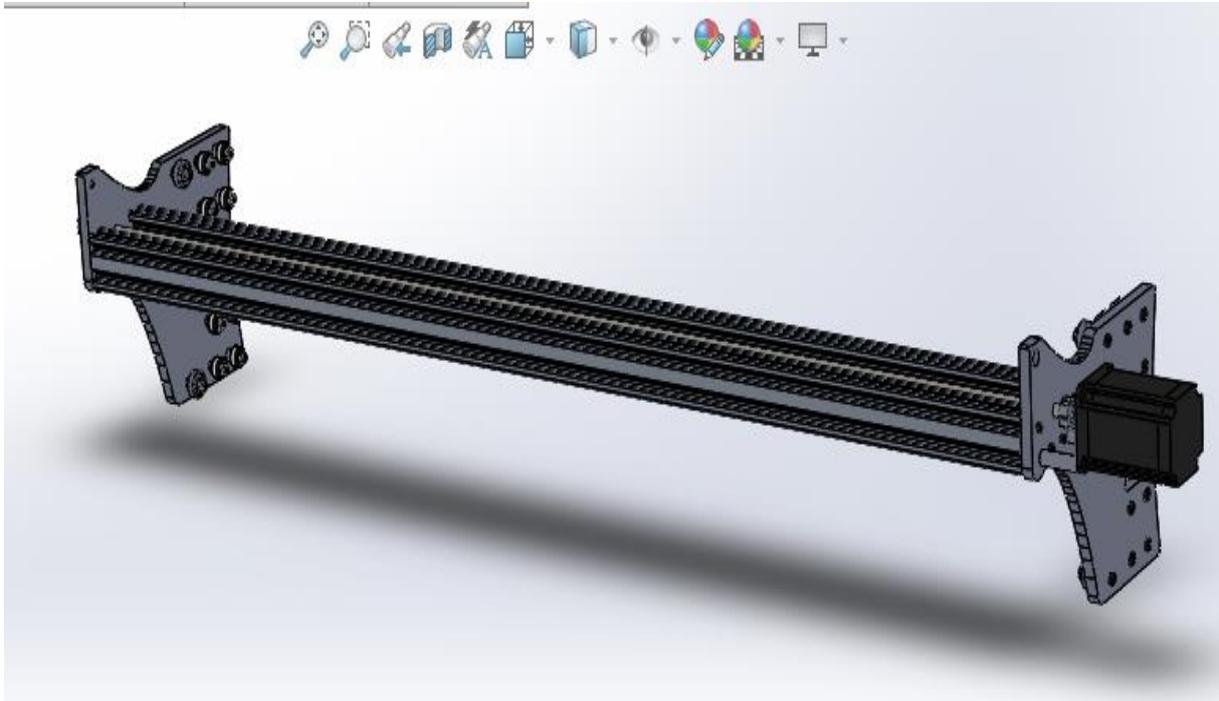


Figure III.8: Axes-Y

### III.4.3.2.4 l'axe Z

Ce dernier sous-assemblage est composé d'une tige de 0.5 mètre, d'une tige filetée et d'une pièce assemblé avec des roues afin d'assurer le mouvement selon l'axe Z (mouvement de haut en bas).

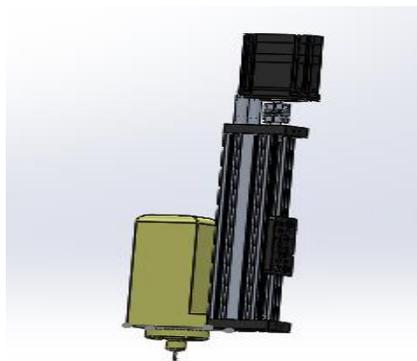
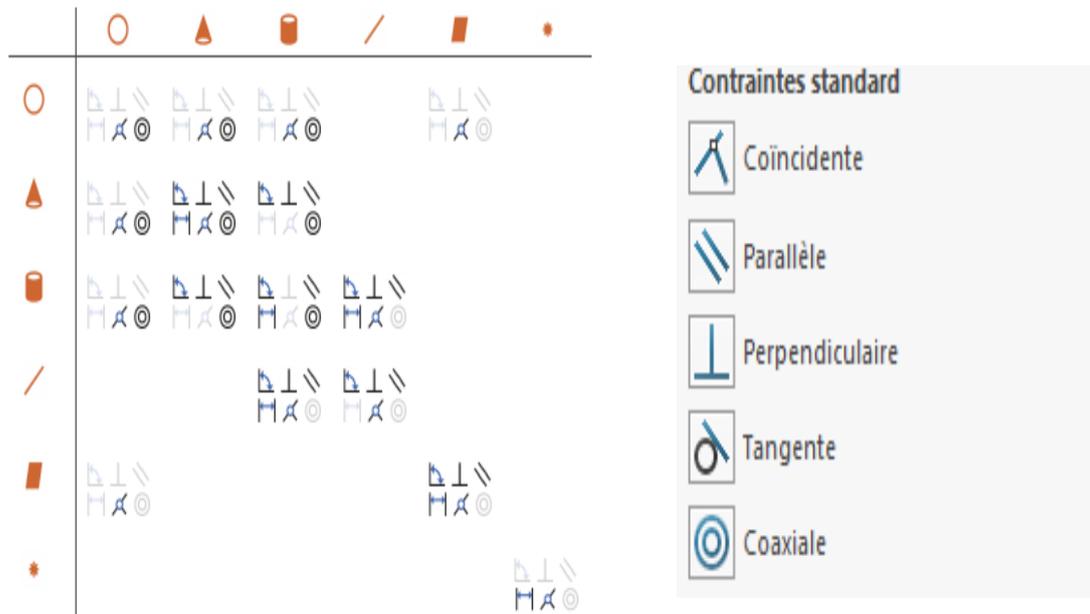


Figure III.9: Axe Z

### III.4.3.3 L'étape de la réalisation de l'assemblage final

Cette étape est sans doute la plus importante car chaque contrainte qu'on va utiliser dans cette assemblage aura un impact directe sur le résultat de l'import dans Matlab, le tableau suivant résume les différent contraintes possible entre deux pièces selon leur formes géométriques

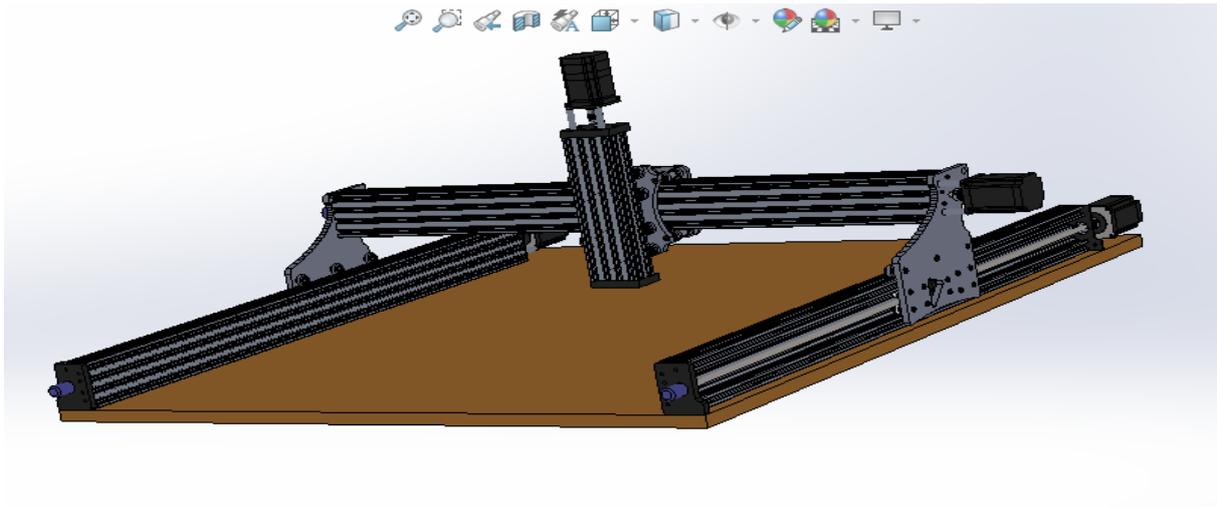


**Figure III.10:** Les différentes contraintes d'assemblage

Par ailleurs comme dans notre cas les mouvements qu'on souhaite définir étant des mouvements de translation, les contraintes qu'on a utilisé afin de réaliser ce mouvement sont les suivantes:

- coaxiale
- parallèle

Finalement on est arrivé au résultat suivant :



**Figure III.11 :** Assemblage final de notre MOCN

### III.5 Conclusion

Dans ce chapitre on a présenté notre machine-outil en précisant ces dimensions et son principe de fonctionnement ainsi que les différentes parties qui composent cette dernière, par la suite on s'est intéressé à la maquette numérique qui sera la partie visualisation de notre simulateur et pour cela on a commencé par définir le principe de la CAO en précisant le logiciel qu'on a employé et enfin on a expliqué les différents étapes de la réalisation de la maquette sur SolidWorks.

# **Chapitre IV :**

## **Réalisation du simulateur sur Matlab**

### **IV.1 Introduction**

Tous au long des chapitres précédant on a expliqué de long en large le principe des MOCN, ses différents caractéristiques, sa programmation avec le Gcode après cela, on a entamé la réalisation de notre simulateur en créant la maquette numérique qui sera le support visuelle de notre simulateur et donc dans ce dernier chapitre on va expliquer comment on a réalisé la partie commande sur le logiciel mathématique de simulation Matlab et pour cela on a commencé par faire l'export/l'import de la maquette, ensuite on va créer un algorithme qui va nous permettra de simuler n'importe quel programme que notre machine-outil pourra réaliser en pratique.

### **IV.2 Définition d'un simulateur**

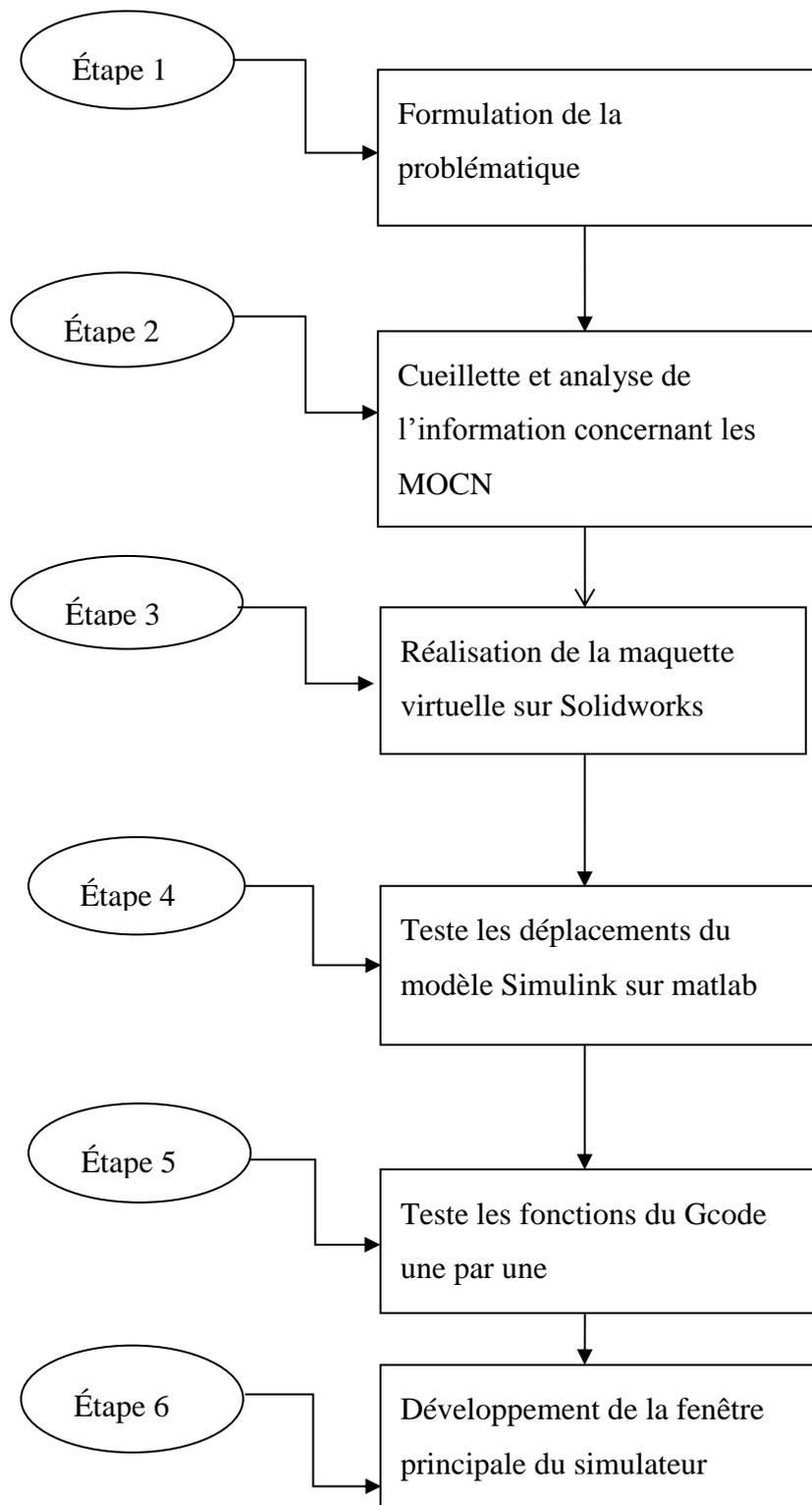
Dans son sens restreint, un simulateur est un logiciel ou un espace de travail sur ordinateur qui permet de suivre un processus ou un système représenter mathématiquement et le plus fidèlement possible. Ces essais ou simulations permettent d'étudier le système sans le représenter physiquement.

### **IV.3 But de notre simulateur**

- détecter les limites du programme à base du langage Gcode
- avoir un aspect visuel de la machine pour pouvoir la présenter aux futurs fabricants ou utilisateur;
- avoir une vision plus large des fonctions préparatoires de type G et les fonctions auxiliaires (de type M);
- avoir la possibilité de simuler tout ou une partie d'un programme d'usinage avec un simulateur propre à la machine et sans être obligé d'avoir recourt aux logiciels disponibles sur le marché;
- permettre au futur acquéreur de la machine de s'entraîner et de se familiariser avec le produit avant même de l'avoir en mains.

### **IV.4 Les étapes de la réalisation du simulateur**

La réalisation d'un simulateur nécessite de suivre des étapes prédéfinies afin de mieux appréhender la problématique et ensuite définir les solutions, elles sont comme suit :



**Figure IV.1:** Etape de l'élaboration du simulateur

Les trois premières étapes ont été abordées dans les chapitres précédents et donc maintenant on va aborder le reste des étapes.

#### **IV.4.1 Teste les déplacements du model simulink dans matlab**

Afin de tester les déplacements de la machine, on a commencé par faire l'export/import de la machine à partir de Solidworks vers matlab ou plus précisément Simscape.

##### **IV.4.1.1 Simscape**

C'est une partie de Matlab qui permet de créer des modèles de systèmes physiques dans l'environnement Simulink. Grâce à Simscape, on peut créer des modèles de composants physiques en s'appuyant sur des connexions physiques qui s'intègrent directement dans des diagrammes de blocs et autres paradigmes de modélisation. [16]

Simscape offre la possibilité d'utiliser des maquettes numériques importés directement de logiciel de conception tel que SolidWorks, Catia ou encore AutoCad dans le but de développer leur système de contrôle et tester leur performances.

##### **IV.4.1.2 L'export de SolidWorks**

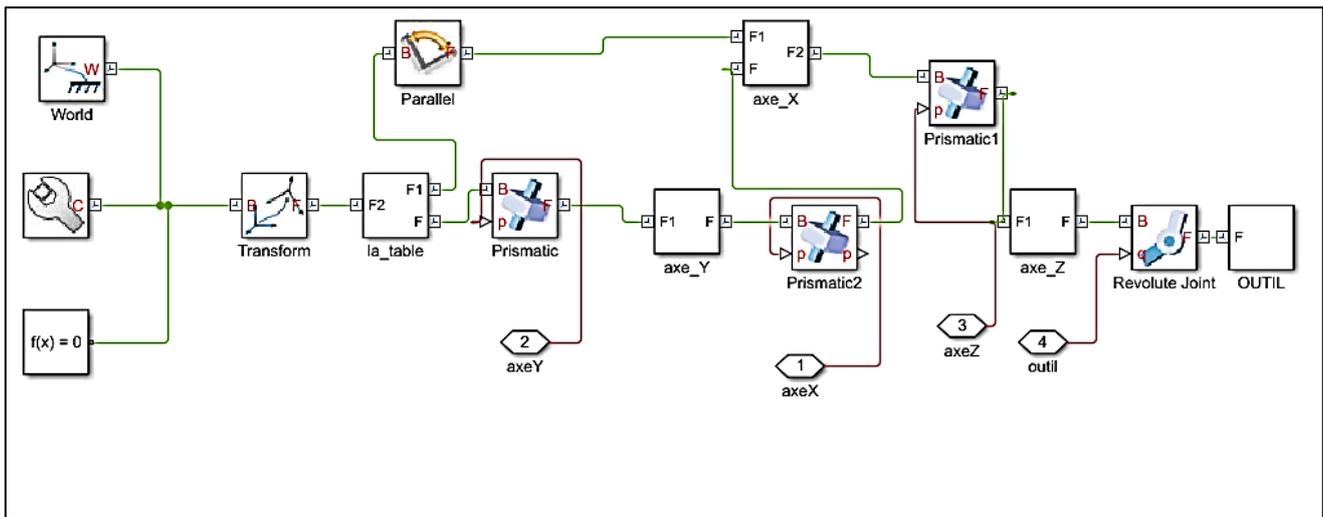
Pour faire l'import de SolidWorks vers Matlab, il est impérativement nécessaire de bien préparer le modèle et cela en respectant les règles suivantes :

- créer le système dans un fichier assemblage qui respecte les liaisons (contraintes) entre les différentes pièces qui compose notre machine
- s'assurer que le système assemblé fonctionne manuellement sur SolidWorks si l'on veut garder les caractéristiques de mouvement propre à notre machine,
- bien définir le système de coordonnées,
- contrôler la résolution et régler les paramètres de qualité d'image sur SolidWorks afin d'éviter d'avoir une simulation trop lente;

Une fois tous ces paramètres régler, on commence l'opération de l'import; sur le menu on clique sur Outil puis sur complément et on sélectionne Simscape Multibody link, et l'import se fait en créant des fichiers STL (STEP) et un fichier XML, si l'opération a été exécuté avec succès, un message nous indiquera que l'import est bien terminé.

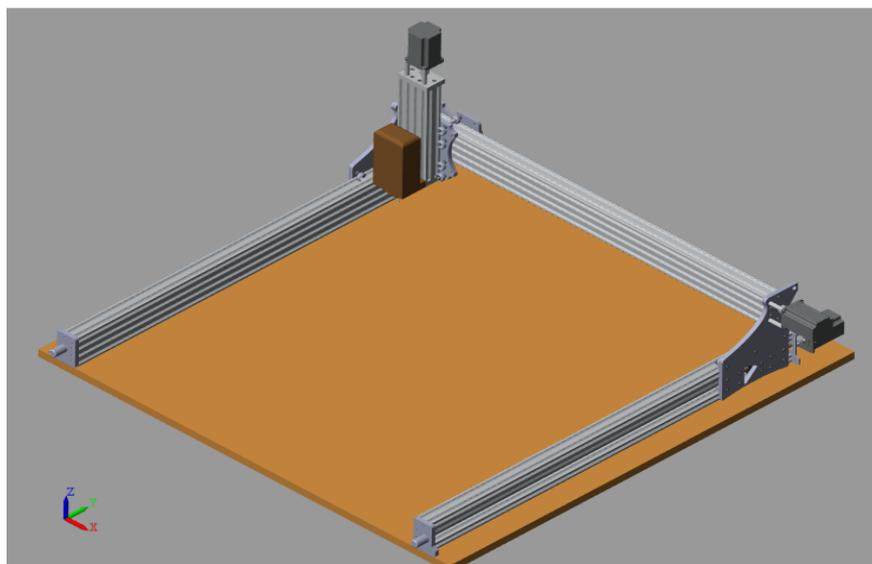
##### **IV.4.1.3 l'import dans Matlab**

Une fois le model exporté, il faut ramener le dossier comportant les fichiers STL et le fichier XML dans l'espace de travail Matlab et l'importation se fait avec la commande `smimport ('nom du fichier.XML')`, un model Simulink Simscape sera automatiquement créer.



**Figure IV.2:** schéma bloc de notre MOCN

Une fois le model créé la visualisation de la machine sur matlab se fait en cliquant sur simulation puis sur Update diagram.



**Figure IV.3 :** notre MOCN sur Matlab

#### IV.4.2 vérification des déplacements des trois axes

Après avoir importé la maquette on a commencé par tester les trois déplacements de la MOCN selon les 3 axes et cela en reliant les blocs prismatiques de chaque axes à des intégrateurs et des constantes afin de créer une rampe et donc un déplacement.

##### ➤ Observations

Nous avons observé que les axes X et Y ne répondent pas de la manière souhaitée (sens de déplacement) et donc on a ajouté des gains statiques avec une valeur -1 afin que les déplacements soient dans le bon sens.

#### IV.4.3 Teste des fonctions du Gcode

Avant de développer l'algorithme principal il a tout d'abord fallu créer et tester les algorithmes des fonctions du Gcode, on a commencé par les fonctions G de déplacement telle que G00, G01, G02, G03 et les fonctions (auxiliaire) M qui concerne l'outil (vitesse, sens, marche et arrêt).

#### IV.4.3 Développement et implémentation de l'algorithme de simulation

On a créé un algorithme qui fait appel à un fichier texte (.txt) et qui arrive à le lire ligne par ligne et à générer des matrices de déplacement pour chaque ligne de code, toutes ces matrices sont regroupées dans une seule et même matrice de déplacement global dans le Workspace de matlab, la machine reçoit cette matrice à travers un bloc nommé (From Workspace), ce bloc a pour rôle de faire le lien entre la partie Simscape (simulink) où se trouve notre machine et la partie programmation où se trouve le programme du G\_code.

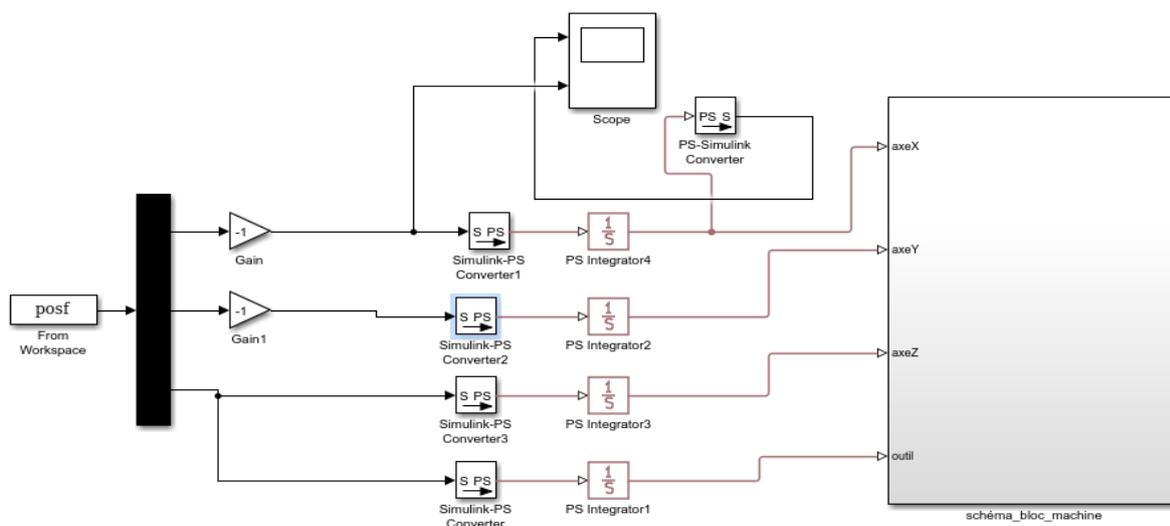


Figure IV.4 : Schéma du simulateur

```

while ischar(tline)
    %disp(tline)
    %length(tline)
    %tline(1:2)
    entree=sscanf(tline,'%*c %*c %*c %f %f %f');
    if strcmp('G00',tline(1:3))
        disp('good 0');
        entree=sscanf(tline,'%*c %*c %*c %f %f %f');
        %a=dvaleur(tline);
        %xf=str2double(a(1))
        %yf=str2double(a(2))
        %x00=x0:(xf-x0)/100:xf;
        %y00=y0:(yf-y0)/100:yf;
        if ((entree(1,1)-pos0(1,1))~=0)
            x00=zeros(1,101)
        else
            x00=zeros(1,101)
        end
        if ((entree(2,1)-pos0(1,2))~=0)
            y00=zeros(1,101)
        else
            y00=zeros(1,101)
        end
        if ((entree(3,1)-pos0(1,3))~=0)
            z00=zeros(1,101)
        else
            z00=zeros(1,101)
        end
        %x001=x001-z001;
    end
end

```

FigureIV .5 : Appel de la fonction G00

```

elseif strcmp('G02',tline(1:3))
    disp('good 2');
    entree=sscanf(tline,'%c %c %c %f %f %f %f %f %f');
    R=sqrt((entree(1,1)-entree(4,1))^2+(entree(2,1)-entree(5,1))^2)
    if ((entree(1,1)-pos0(1,1))~=0)
        x00=pos0(1,1):(entree(1,1)-pos0(1,1))/100:entree(1,1)
    else
        x00=zeros(1,101)
    end
    if ((entree(2,1)-pos0(1,2))~=0)
        %y00=pos0(1,2):(entree(2,1)-pos0(1,2))/100:entree(2,1)
        y00=sqrt(R^2-(x00-entree(4,1)).^2)+entree(5,1)
    else
        y00=zeros(1,101)
    end
    %if ((entree(3,1)-pos0(1,3))~=0)
        %z00=pos0(1,3):(entree(3,1)-pos0(1,3))/100:entree(3,1)
    %else
        z00=zeros(1,101)+pos0(1,3)
    %end
    %[x00',y00',z00']
    posf=[posf;[x00',y00',z00']]
    pos0=posf(length(posf),:);

elseif strcmp('G03',tline(1:3))
    disp('good 3');

else
    disp('good 100');

```

FigureIV.6 : Appel de la fonction G02

#### IV.5 Conclusion

Dans ce dernier chapitre on a finalisé notre simulateur en introduisant au modèle Simulink qui est la partie visualisation de notre simulateur un algorithme qui permet de faire appel au programme du Gcode et de l'exécuter.

## **Conclusion générale**

## **Conclusion générale**

Ce modeste travail nous a offert l'opportunité d'approfondir nos connaissances pour ce qui concerne les machines-outils à commande numérique car comme nous l'avant déjà expliquer auparavant, la conception d'un simulateur nécessite une recherche pousser et une analyse détaillée des problèmes que le simulateur doit impérativement solutionner.

Dans la chaine de développement et de conception d'une machine-outil à commande numérique jusqu'à l'étape de la réalisation d'un programme d'usinage, de nombreux problèmes peuvent être détecter tel que les problèmes de dimensionnement des axes (limites du programme d'usinage) et c'est pour cela que la réalisation d'un simulateur reste la meilleure solution pour éviter de nombreuse complications mais aussi pour tester les programmes d'usinage une fois que la machine a été conçu pour éviter un perte de temps et une perte de matière première.

Dans ce travail on a pu réaliser que certaines des fonctions du Gcode mais avec encore plus de temps et de recherche, il est possible de réaliser la majorité des fonctions.

La réalisation de ce simulateur sur l'espace de travail Matlab présente quelque limite puisque on peut le simuler que sur le logiciel Matlab.

## BIBLIOGRAPHIE

[1] Université de KAIROUAN (Tunisie), Cours Génie mécanique, chapitre 1, les machines à commande numérique

[2] Gilles prud'homme, 1995, « commande numérique des machines-outils » Technique de l'ingénieur, génie mécanique, usinage, B7130.

[3] MERGHACHE Sidi Mohamed magister, Contrôle et mesure intégrée sur machine-outil CNC. Contribution à l'identification et étude des erreurs géométriques.

[4] Salvan Jorisnet Joly Killian,,<http://genie-electrique.insa-strasbourg.fr/projet-genie-electrique-par-alternance-conception-de-la-commande-dune-machine-3-axes-et-realisation-dun-jumeau-numerique/>,2017/2018

[5] Mr Ouatah El hanafi, CFAO, cours, 2018, électromécanique, Université Abdarraahmane Mira, Béjaia.

[6] Zahia HESSAINIA Université Des Frères Mentouri Constantine 1 Faculté Des Sciences De La Technologie Département De Génie Mécanique

[7] TSTI GMP, productique étude des machines outil à C.N. PG & GT 28/11/2008

[8] Sources et contributeurs de l'article

Programmation de commande numérique Source:

<http://fr.wikipedia.org/w/index.php?oldid=73873596> Contributeurs: Anthere, Aristote2, Arnaud.Serander, Badmood, Bapti, Betbert 29, Bobodu63, Butterfly austral, Calcineur, Carrystory, Chaps the idol, Cédric Boissière, DamienR69120, Domsau2, DonCamillo, Ercii, Esprit Fugace, F&TI "Monin Stéphane", Flavian, François-Karim, Ggal, Gmp.sti, Haha, Hemmer, Hortense B., ICN2, JD, Jef-Infojef, Jerome66, Ji-Elle, Kilith, Klem88, Litlok, Lmaltier, Ltrlg, Michel c12, Michel95150, Mro, PRZ, Raphio, Sanao, Sebism, Sebjarod, Sherbrooke, Silex6, Stéphane33, Tanjaoui, Taveneaux, Trusty, Wik CAM, Witoki, Yuki55, Zedh, Zetud, Zubro, 166 modifications anonymes, [http://robotix.ah-oui.org/user\\_docs/dos10/Gcode-fr.pdf](http://robotix.ah-oui.org/user_docs/dos10/Gcode-fr.pdf), 09/04/2018

[9] Lebearcnc, <https://lebearcnc.com/>, 25 06 2018

[10] sitemeca.free.fr, 08/04/2018

[11] <http://tu-quincy.fr/codes-de-programmation-iso-machine-num/>, 04/04/2018, 11:38

[12] Commandes Numérique NUM 1020/1040/1050/1060/,NUM Catalogue 2000.Manuel de programmation et Manuel de L'opérateur

[13] **PATCICE OGUIC.** (2004). moteurs pas à pas et PC. édition technique et scientifique française (Dunod), pp. "3-88

[14] **GERARD YVRAUT.** (1999). les moteurs pas à pas. Séminaire Bellegarde novembre 1999.

[15] Brohen LOUHICHI, thèse « Intégration CAO/Calcul par reconstruction du modèle CAO à partir des résultats éléments finis », 2008

[16] Patrick BLAIN, Technique de l'ingénieur « CAO et méthodologie de conception » document B 2 810

[17] [www.SolidWorks.com](http://www.SolidWorks.com), 26 06 2018

[18] Christian Tavernier « Arduino Maitrise sa programmation et ses cartes d'interface (shield) », paris 2011, DUNOD

[19] D.Duret, « simulation de gamme d'usinage », revue de l'ingénieur et de technicien de l'enseignement technique, n°229, 1981

[20] A.Cheikh, N.Cheikh, 2011, cours de CFAO1, Université de Tlemcen, Algérie

[21] [www.dzduino.com](http://www.dzduino.com) , 18/04/2018

[22] [www.mathworks.com](http://www.mathworks.com), 25/04/2018

## Annexes

---

### Les G codes :

Code G	Fonction
G00	Positionnement rapide
G01	Interpolation linéaire(avance de coupe)
G02	Interpolation circulaire ou interpolation hélicoïdale sens horaire
G03	Interpolation circulaire ou interpolation hélicoïdale sens anti horaire
G02.2-G03.2	Interpolation développante(sens horaire/anti horaire)
G02.3-G03.3	Interpolation exponentielle(sens horaire/anti horaire)
G02.4-G02.4	Conversion de coordonnées tridimensionnelle(sens horaire /anti horaire)
G04	Temporisation
G05	Commande de contournage A1
G05.1	Commande de contournage A1/Lissage Nano/interpolation lisse
G05.4	Activation/désactivation HRV3,4
G06.2	Interpolations NURBS
G07	Interpolation avec axe hypothétique
G07.1(G107)	Interpolation cylindrique
G08	Commande de contournage A1
G09	Arrêt précis
G10	Entrée de donnée programmable
G10.6	Recul de l'outil et reprise
G10.9	Commutation programmable de la programmation du diamètre/programmation du rayon
G11	Annulation du mode d'entrée des données programmables
G12.1	Mode interpolation en coordonnées polaires
G13.1	Annulation du mode interpolation en coordonnées polaires
G15	Annulation de la commande de coordonnées polaires
G16	Commande de coordonnées polaires
G17	Sélection du plan Xp Yp
G18	Sélection du plan Xp Zp
G19	Sélection du plan Yp Zp
G20(G70)	Entrée en pouces
G21(G71)	Entrée en mm
G22	Fonction de vérification de course enregistrée activée
G23	Fonction de vérification de course enregistrée désactivée
G25	Détection des fluctuations de la vitesse de broche désactivée
G26	Détection des fluctuations de la vitesse de broche activée
G27	Contrôle du retour à la position de références
G28	Retour automatique à la position de référence
G29	Déplacement depuis la position de référence
G30	Retour à la 2 <sup>ème</sup> , 3 <sup>ème</sup> et 4 <sup>ème</sup> position de référence
G30.1	Retour à la position de référence flottante
G31	Fonction de saut
G31.8	Saut d'axe de boîte d'avance électrique
G33	Filetage
G34	Filetage à pas variable
G35	Filetage circulaire sens horaire
G36	Filetage circulaire sens anti horaire

## Annexes

G37	Mesure automatique de la longueur d'outil
G38	Compensation d'outil de coupe ou de rayon de pointe d'outil : vecteur de conservation
G39	Compensation d'outil de coupe ou de rayon de pointe : interpolation circulaire angulaire
G40	<ul style="list-style-type: none"> <li>▪ Compensation d'outil de coupe ou de rayon de pointe d'outil : Annulation</li> <li>▪ Compensation d'outil de coupe tridimensionnelle : Annulation</li> </ul>
G41	<ul style="list-style-type: none"> <li>▪ Compensation d'outil de coupe ou de rayon de pointe d'outil :Gauche</li> <li>▪ Compensation d'outil de coupe tridimensionnelle : Gauche</li> </ul>
G41.2	Compensation d'outil de coupe pour usinage 5 axes :gauche(type1)
G41.3	Compensation d'outil de coupe pour usinage 5 axes :(correction du bord d'attaque)
G41.4	Compensation d'outil de coupe pour usinage 5 axes :gauche(type1)(compatible avec FS16i)
G41.5	Compensation d'outil de coupe pour usinage 5 axes :gauche(type1) (compatible avec FS16i)
G41.6	Compensation d'outil de coupe pour usinage 5 axes :gauche(type2)
G42	<ul style="list-style-type: none"> <li>▪ Compensation d'outil de coupe ou de rayon de pointe d'outil :Droite</li> <li>▪ Compensation d'outil de coupe tridimensionnelle : Gauche</li> </ul>
G42.2	Compensation d'outil de coupe pour usinage 5 axes :droite(type1)
G42.4	Compensation d'outil de coupe pour usinage 5 axes :droite(type1) (compatible avec FS16i)
G42.5	Compensation d'outil de coupe pour usinage 5 axes :droite(type1) (compatible avec FS16i)
G42.6	Compensation d'outil de coupe pour usinage 5 axes :droite(type2)
G40.1	Annulation de la commande normale du profil
G41.1	Commande normale du profil activée :gauche
G41.2	Commande normale du profil activée :droite
G43	Compensation de longueur d'outil+
G44	Compensation de longueur d'outil-
G43.1	Compensation de longueur d'outil dans la direction de l'axe de l'outil
G43.4	Contrôle du point de centre de l'outil (type1)
G43.5	Contrôle du point de l'outil (type 2)
G45	Augmentation de la correction d'outil
G46	Diminution de la correction d'outil
G47	Double augmentation de la correction d'outil
G48	Double diminution de la correction d'outil
G49(G49.1)	Annulation de la compensation de longueur d'outil
G50	Annulation d'échelle
G51	Echelle
G50.1	Annulation de l'image miroir programmable
G51.1	Image miroir programmable

## Annexes

---

G50.2	Annulation du tournage polygonal
G51.2	Tournage polygonal
G52	Définition du système de coordonnées locales
G53	Définition du système de coordonnées machine
G53.1	Commande de direction de l'axe de l'outil
G54(G54.1)	Sélection du système de coordonnées pièce 1
G55	Sélection du système de coordonnées pièce 2
G56	Sélection du système de coordonnées pièce 3
G57	Sélection du système de coordonnées pièce 4
G58	Sélection du système de coordonnées pièce 5
G59	Sélection du système de coordonnées pièce 6
G60	Positionnement unidirectionnel
G61	Mode arrêt précis
G62	Correction d'angle automatique
G63	Mode taraudage
G64	Mode d'usinage
G65	Appel de macro
G66	Appel de macro A
G66.1	Appel de macro B
G67	Annulation appel de macro A/B
G68	Démarrage de la rotation du système de coordonnées ou activation du mode de conversion de coordonnées tridimensionnelles
G69	Annulation de la rotation du système de coordonnées ou activation du mode de conversion de coordonnées tridimensionnelles
G68.2	Sélection du système de coordonnées de fonctions
G72.1	Copie de profil (copie de rotation)
G72.2	Copie de profil(copie linéaire)
G73	Cycle de perçage avec déburrage
G74	Cycle de taraudage à gauche
G76	Cycle d'alésage fin
G80	Annulation de cycle fixe
G80.5	Paire de boîte d'avance électronique :démarrage de la synchronisation
G80.8	Boîte d'avance électronique :démarrage de la synchronisation
G81	Cycle de perçage ou cycle de centrage
G81.1	Tronçonnage
G81.5	Paire de boîte d'avance électronique :démarrage de la synchronisation
G81.8	Boîte d'avance électronique :démarrage de la synchronisation
G82	Cycle de perçage ou d'alésage inverse
G83	Cycle de perçage avec déburrage
G84	Cycle de taraudage
G84.2	Cycle de taraudage rigide(FS15)
G84.3	Cycle de taraudage rigide gauche (FS15)
G85	Cycle d'alésage
G86	Cycle d'alésage
G87	Cycle d'alésage inverse
G88	Cycle d'alésage

## Annexes

G89	Cycle d'alésage
G90	Programmation en absolue
G91	Programmation en incrémental
G91.1	Vérification de la valeur incrémental maximale spécifiée
G92	Définition du système de coordonnées pièce ou limitation de la vitesse maximale de la broche
G92.1	Prédéfiniion du système de coordonnées pièce
G93	Avance à temporisation inverse
G94	Avance par minute
G95	Avance par tour
G96	Contrôle de vitesse de surface constante
G97	Annulation du contrôle de vitesse constante
G98	Cycle fixe :retour au niveau initial
G99	cycle fixe :retour au niveau du point R
G107	Interpolation cylindrique
G112	Mode interpolation en coordonnées polaires
G113	Annulation du mode interpolation en coordonnées polaires

Les fonctions auxiliaires M :

Liste des codes M les plus utilisés. Ne sont pas inclus les codes M constructeur.

Type	Signification	Fonctions
M00	Arrêt programmé	Après
M01	Arrêt optionnel	Après
M02	Fin de programme(identique à M30)	Après
M03	Rotation broche sens horaire	Avant
M04	Rotation broche sens trigonométrique(ou anti-horaire)	Avant
M05	Arrêt rotation broche	Après
M06	Changement outil	Après
M07	Marche second arrosage	Avant
M08	Marche arrosage principal	Avant
M09	Arrêt arrosage	Après
M19	Orientation broche	Après
M30	Fin de programme(identique à M02)	Après
M98	Appel de sous-programme	-
M99	Retour de sous-programme	après

## Annexes

---

### Les caractéristiques de la machine DENZER-CNC :

Dimensions de la surface de la table <ul style="list-style-type: none"><li>• Longueur (cm)</li><li>• Largeur (cm)</li></ul>	97 97
Encombrement de la machine <ul style="list-style-type: none"><li>• Longueur (cm)</li><li>• Largeur (cm)</li><li>• Hauteur (cm)</li></ul>	110 110 128
Nombre des étaux	2
Taille des étaux (cm)	96
Nombre de vitesse de la broche	1
La vitesse de la broche <ul style="list-style-type: none"><li>• A vide (tr/mn)</li><li>• Avec fraise (tr/mn)</li></ul>	30000 28000
Distance entre le nez de la broche et la surface de la table <ul style="list-style-type: none"><li>• Minimale (mm)</li><li>• Maximale (mm)</li></ul>	10 260
Déplacement maximale longitudinale de la table (cm)	90
Déplacement maximale transversale de la table (cm)	90
Déplacement maximale verticale de la table (cm)	25
Poids (kg)	50