REPUBLIC ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITE DR. YAHIA FARES DE MEDEA
FACULTE DES SCIENCES ET DE LA TECHNOLOGIE
Département du Génie Électrique et de l'Informatique



Projet de fin d'étude

Présenté pour l'obtention du Diplôme De master 2 en Génie Électrique

Spécialité : Commande et électronique industrielle

# THEME

# PILOTER UN MODELE SOLIDWORKS AVEC LABVIEW SOFT MOTION

Préparé et présenté par
Mr. GAHLOUZ Aissa
Mr. MESRATI Khaled

Proposé et dirigé par
Mr. BETTACHE Farouk
Mr. AMMOUR Lamine

Année universitaire 2015/2016

# Remerciements

Nous adressons nos vifs remerciements à :

— Tous les enseignants et tous ceux à qui nous devons notre formation.

— Tous ceux qui, d'une quelconque façon,Ont contribué à l'élaboration de ce mémoire.

Cordialement.

# Dédicaces

Nous dédions ce modeste travail :

— A nos chers parents qui n'ont soutenu durant toute la durée de nos études.

— A nos chères familles.

— A tous nos amis et nos proches.

— A tous nos camarades dans la promotion 2016

— A tous ceux qui nous sont chers.

Khaled et Aissa

# Résumé

**Résumé** Ce projet inclura le pilotage et la création d'une machine à commande numérique (CNC) informatisé.

La première partie du projet contient l'histoire généralisé sur la commande numérique informatisée sur les types du moteur pas a pas, des machines a commande numériques et le tiges filetée, On se base dans La deuxième partie du projet sur la création d'un modèle machine a commande numérique avec logiciel solidworks et la simulation du modèle (mouvement) La troisième partie du projet inclura une généralité sur le logiciel labview et le modèle soft motion et G-code Comme dernier étape on va apercevoir l'interconnexion des deux logiciels avec l'une des différentes simulations choisit.

**Mots clés** : CNC, Moteur pas à pas, G-code.

# Table des matières

**1 Généralité Sur Les Machines CNC** **2**

1.1 Historique . . . . . 2

1.2 Introduction . . . . . 3

1.3 Définition . . . . . 4

1.4 Généralités . . . . . 5

1.5 Principes de base CNC . . . . . 5

1.6 La programmation . . . . . 6

1.7 Produits réels . . . . . 6

1.8 Les types des machines CNC . . . . . 7

1.8.1 Commande numérique point à point . . . . . 8

1.8.2 Commande numérique par axiale . . . . . 8

1.8.3 Commande numérique de contourne . . . . . 9

1.9 Moteur pas à pas . . . . . 10

1.9.1 définition . . . . . 10

1.9.2 Types de moteur pas-à-pas . . . . . 10

1.9.3 Comparison des différents types de moteurs pas à pas . . . . . 14

1.9.4 Comment fonctionne un moteur pas à pas ? . . . . . 14

1.9.5 Domaines d'application d'un moteur pas à pas . . . . . 15

1.9.6 Les modes de commande du moteur pas à pas biphasés . . . . . 16

1.10 les tiges filetées . . . . . 21

1.10.1 la tige filetée . . . . . 21

1.10.2 le filetage . . . . . 21

1.11 Conclusion . . . . . 21

**2 Conception modèle CNC avec SolidWorks 23**

2.1 historique . . . . . 23
2.2 Introduction . . . . . 23
2.3 Fonctionnement . . . . . 24
2.3.1 Les pièces . . . . . 24
2.3.2 les assemblages . . . . . 27
2.3.3 La miss en plan . . . . . 28
2.3.4 Dessin de définition établi avec SolidWorks . . . . . 28
2.4 Conception modèle CNC avec solidworks . . . . . 30
2.4.1 Les pièces . . . . . 30
2.4.2 Les dimensions des pièces . . . . . 35
2.4.3 l'assemblage . . . . . 35
2.5 Simulation du modèle (mouvement) . . . . . 36
2.6 Conclusion . . . . . 38

**3 La programmation avec logiciel labview 39**

3.1 Introduction . . . . . 39
3.2 Historique . . . . . 40
3.3 Modèle LABVIEW soft motion . . . . . 41
3.4 Présentation des instruments virtuels . . . . . 41
3.4.1 Face-avant . . . . . 42
3.4.2 Diagramme . . . . . 43
3.5 Boucles et structures . . . . . 46
3.5.1 Boucles . . . . . 47
3.5.2 Structures Condition, Séquence et Événement . . . . . 50
3.6 G-code . . . . . 52
3.6.1 Définition . . . . . 52
3.6.2 Fonction . . . . . 52
3.6.3 Interpolation . . . . . 53

3.7 Conclusion . . . . . 55

**4 Interconnexions entre labview et solidworks 56**

4.1 Introduction . . . . . 56

4.2 la commande manuelle . . . . . 57

4.2.1 Création du projet et import des moteurs de solidworks motion . . . 57

4.2.2 Création d'un axe dans labview . . . . . 58

4.2.3 Création du VI de commande dans Labview . . . . . 60

4.2.4 Exécution du VI et simulation virtuelle du mouvement sous SW motion 65

4.3 Commande automatique . . . . . 69

4.3.1 Création le projet . . . . . 69

4.3.2 création les axes . . . . . 69

4.3.3 Commande (perceuse) . . . . . 69

4.3.4 Commande (automatique sans G code) . . . . . 71

4.3.5 Commande (automatique avec G code) . . . . . 73

4.4 Conclusion . . . . . 75

# Table des figures


1.1 Ancien machine cnc . . . . . 3
1.2 Machine cnc . . . . . 4
1.3 les axes de machine CNC. . . . . 6
1.4 produit réel. . . . . 7
1.5 Commande Numérique Point A Point . . . . . 8
1.6 Commande Praxiale. . . . . 9
1.7 Commande contourne . . . . . 9
1.8 Moteur pas à pas . . . . . 10
1.9 Vue d'un moteur à reluctance variable . . . . . 12
1.10 Structure d'un moteur pas à pas (montage bipolaire) . . . . . 15
1.11 la séquence des alimentations de phases pour un moteur à aimants permanents (mode 1) . . . . . 16
1.12 Commande dans le mode 1 . . . . . 17
1.13 la séquence des alimentations de phases pour un moteur à aimants permanents (mode 2) . . . . . 17
1.14 Commande dans le mode 2 . . . . . 18
1.15 la séquence des alimentations de phases pour un moteur à aimant permanents (Mode 3) . . . . . 18
1.16 la séquence des alimentations de phases pour un moteur à aimant permanents (Mode 4) . . . . . 19
1.17 la séquence des alimentations de phases pour un moteur à aimant permanents (Mode 5) . . . . . 20

2.1 une pièce avec solidworks . . . . . 24
2.2 Différentes fonctions logicielles . . . . . 25
2.3 Assemblage avec solidworks . . . . . 27
2.4 Dessin de définition établi avec SolidWorks . . . . . 28
2.5 Les trois concepts de base (SOLIDWORKS) . . . . . 30
2.6 création de la table . . . . . 31
2.7 La table initiale . . . . . 31
2.8 création d'une tige simple . . . . . 32
2.9 La tige simple . . . . . 32
2.10 création tige filetée . . . . . 33
2.11 tige filetée . . . . . 33
2.12 création moteur . . . . . 34
2.13 Moteur pas à pas imaginaires . . . . . 34
2.14 Fenêtre pour sélectionner assemblage . . . . . 35
2.15 assemblage deux pièces . . . . . 36
2.16 Fenêtre pour cocher solidworks motion . . . . . 36
2.17 La simulation de mouvement . . . . . 37

3.1 instruments virtuels soft motion . . . . . 41
3.2 exemple de face avant . . . . . 42
3.3 exemple de diagramme . . . . . 43
3.4 bouton rotatif . . . . . 44
3.5 Un terminal DBL . . . . . 44
3.6 icône dans le coin supérieur droit des fenêtres de la face-avant et du diagramme. 45
3.7 Une icône pour représentation graphique . . . . . 46
3.8 Boucles For . . . . . 47
3.9 Boucles While . . . . . 48
3.10 Configuration de la boucle cadencée . . . . . 49
3.11 Boucle cadencées . . . . . 49
3.12 Structures Condition . . . . . 50

3.13 Structures Séquence . . . . . 51
3.14 Structures Événement . . . . . 52
3.15 interpolation linéaire en avance rapide. . . . . 54
3.16 interpolation circulaire en sens horaire . . . . . 55

4.1 machine outil à commande numérique . . . . . 57
4.2 fenêtre de projet . . . . . 57
4.3 Fenêtre d'ouvrir l'assemblage . . . . . 58
4.4 Fenêtre d'ouvrir l'assemblage . . . . . 58
4.5 crée les axes softmotion . . . . . 59
4.6 fenêtre de sélectionner les trois axes . . . . . 59
4.7 les trois axes . . . . . 60
4.8 cocher Enable drive on transition to active mode . . . . . 60
4.9 nouvelle VI . . . . . 61
4.10 VI principale avec les sous VI . . . . . 61
4.11 sélectionner Straight-Line Move . . . . . 62
4.12 comment changer la méthode . . . . . 62
4.13 présentation une Straight-Line Move avec les composants . . . . . 63
4.14 Diagramme de la commande général . . . . . 64
4.15 Face avant de la commande générale mal positionnés . . . . . 65
4.16 Face avant de la commande générale . . . . . 65
4.17 déployer les axes avec poste de travaille . . . . . 66
4.18 apply pour complet l'interconnexion . . . . . 66
4.19 Lancez la simulation . . . . . 67
4.20 simulation entre labview et solidworks . . . . . 67
4.21 stop simulation . . . . . 68
4.22 rapport de transmission dans une liaison . . . . . 68
4.23 Réalisation du diagramme (perceuse) . . . . . 70
4.24 Realisation de la face avant (perceuse) . . . . . 70
4.25 Exécution du VI et simulation (perceuse) . . . . . 71

4.26 Réalisation du diagramme (automatique sans G code) . . . . . 72
4.27 Réalisation de la face avant (automatique sans G code) . . . . . 72
4.28 Exécution du VI et simulation . . . . . 73
4.29 Réalisation du diagramme (Programmation) avec G code . . . . . 74
4.30 Réalisation de la face avant (automatique avec G code) . . . . . 74

# Liste des tableaux

1.1 Comparison des différents types de moteurs pas à pas . . . . . 14

2.1 Les dimensions des pièces . . . . . 35

# Liste des Symboles

| | |
|---|---|
| MOCN | Machine outil à commande numérique |
| CNC | Computer Numerical Control |
| CN | Commande numérique |
| M.O | Machine Outil |
| FCN | Fraisage à commande numérique |
| TCN | Tournage à commande numérique |
| UCN | Usinage à commande numérique |
| FAO | Fabrication Assistée par Ordinateur |
| 3D | Trois démontions |
| LabVIEW | Laboratory Virtual Instrument Engineering Workbench |
| VI | Instrument virtuel |
| ISO | International Standare de Organisation |

# Introduction Générale

Dans le domaine industriel on a besoin d'utiliser des machines pour réaliser des pièces dont la majorité sont des machines CNC (commande numérique.). Le premier rôle d'une machine CNC (Computer Numerical Control), est de générer des mouvements. Elle recevra des valeurs de positionnement, de vitesse et d'accélération et générera, des consignes numériques en sortie suite à un traitement. Elle dispose d'une grande puissance de calcul et d'une plus grande souplesse d'utilisation qu'un automate programmable. Dans La dernière décennie on a aperçu que les industrialiste ne fabrique pas leur pièces qu'après la simulation informatise ce qui nous a poussé a asseye d'accomplir une simulation exact a partir de notre projet.

Notre objectif est d'avoir un pilotage et une modélisation de la machine a commande numérique avec le logiciels solidworks et commander par logiciel labview . Le présent travail s'articule autour des quatre chapitres suivants : Le premier chapitre est consacré pour la présentation des machines outil à commande numérique MOCN, un aperçu général sur la technologie des MOCN, citant la classification, l'architecture et la programmation des MOCN pour différentes commandes.

Dans le second chapitre on a concepté la machine outil à commande numérique par logiciel solidworks avec la simulation. Le troisième chapitre inclura une généralité sur le logiciel labview et le modèle soft motion et G-code Le quatrième chapitre l'interconnexion entre labview et solidworks et la simulation .

# Chapitre 1

# Généralité Sur Les Machines CNC

## 1.1 Historique

— 1955 John Parsons et l'US Air Force définissent un besoin de développer une machine-outil capable d'usiner pièces d'aéronefs tolérance complexes et étroits avec le même temps de qualité après le temps (reproductibilité). MIT est le sous-traitant et construit la machine pour le projet.

— 1959 MIT annonce automatiques Outils programmée (APT) de langage de programmation.

— 1960 Commande numérique directe (DNC). Ceci élimine des programmes de perforation de bandes de papier et permet programmeurs pour envoyer des fichiers directement aux machines outils.

— 1968 machine-outil constructeurs marché le premier centre d'usinage Kearney & Trecker.

— 1970 machines-outils CNC et commande numérique distribuée.

— 1980 systèmes FAO basés graphiques introduit. UNIX et PC basés les systèmes disponibles.

— 1990 baisse des prix dans la technologie CNC.

— 1997 PC-Windows / NT repose "Open Control architecture modulaire (OMAC)" systèmes mis en place pour remplacer les contrôleurs "firmware".




FIGURE 1.1 – Ancien machine cnc

## 1.2 Introduction

Une machine-outil à commande numérique (MOCN, ou simplement CN) est une machine outil dotée d'une commande numérique. Lorsque la commande numérique est assurée par un ordinateur, on parle parfois de machine CNC pour computer numerical command, francisé en ≪ Commande numérique par calculateur ≫ . Le premier rôle d'une machine CNC (Computor Numerical Control), est de générer des mouvements. Elle recevra des valeurs de positionnement, de vitesse et d'accélération générera, suite à un traitement, des consignes numériques en sortie. En résumé, la commande CNC va générer des mouvements selon des consignes numériques et réguler ces derniers par des systèmes en boucle fermée, alors que l'automate va acquérir des signaux binaires et mettre à jour les sorties concernées. À noter que les données reçues en entrées par l'automate sont également exploitable par la commande numérique et inversement.




FIGURE 1.2 – Machine cnc

## 1.3 Définition

Dans le domaine de la fabrication mécanique, le terme ≪ commande ≫ désigne l'ensemble des matériels et logiciels ayant pour fonction de donner les instructions de mouvements à tous les éléments d'une machine-outil :

— l'outil (ou les outils) d'usinage équipant la machine,

— les tables ou palettes où sont fixées les pièces,

— les systèmes de magasinage et de changement d'outil,

— les dispositifs de changement de pièce,

— les mécanismes connexes, pour le contrôle ou la sécurité, l'évacuation des copeaux, ...

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

— Fraisage à commande numérique (FCN) ;

— Tournage à commande numérique (TCN) ;

— Usinage à commande numérique (CUCN) ;

— Rectification à commande numérique ;

— Electro-érosion à commande numérique.

Dans chaque famille, les méthodes de montage et de travail sont totalement différentes, mais elles se rejoignent sur le principe de programmation, la grande majorité des machines utilisant un langage ISO. À cela peuvent se rajouter des interfaces dites conversationnelles ou par apprentissage qui simplifient l'utilisation de la machine.

La notion de commande numérique s'étend aussi au domaine de la chaudronnerie : découpage plasma, presse plieuse.

## 1.4 Généralités

Par extension, on appelle ≪ commande numérique ≫ l'armoire de commande recevant le programme d'usinage sous forme d'un ruban perforé, d'une bande magnétique, ou de données issues d'un ordinateur. On désigne parfois ainsi la machine complète équipée d'un tel dispositif. On parle d'un tour à commande numérique, ou d'une fraiseuse à commande numérique, par opposition à un tour conventionnel ou une fraiseuse conventionnelle, dont les mouvements sont commandés manuellement ou par un dispositif automatisé d'une façon figée.

Les machines-outils spécialisées (aléseuses-perceuses, fraiseuses) à commande numérique ont évolué en centres 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 18 (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.

## 1.5 Principes de base CNC

Toutes les machines commandées par ordinateur sont en mesure de contrôler avec précision et de façon répétée le mouvement dans différentes directions. Chacune de ces directions de mouvement est appelé un axe. Selon le type de machine il existe communément deux à cinq axes. Additionnelle, un axe CNC peut être soit un axe linéaire, dans laquelle le mouvement se trouve dans une ligne droite, ou un axe de rotation avec le mouvement suivant une trajectoire

circulaire[1].

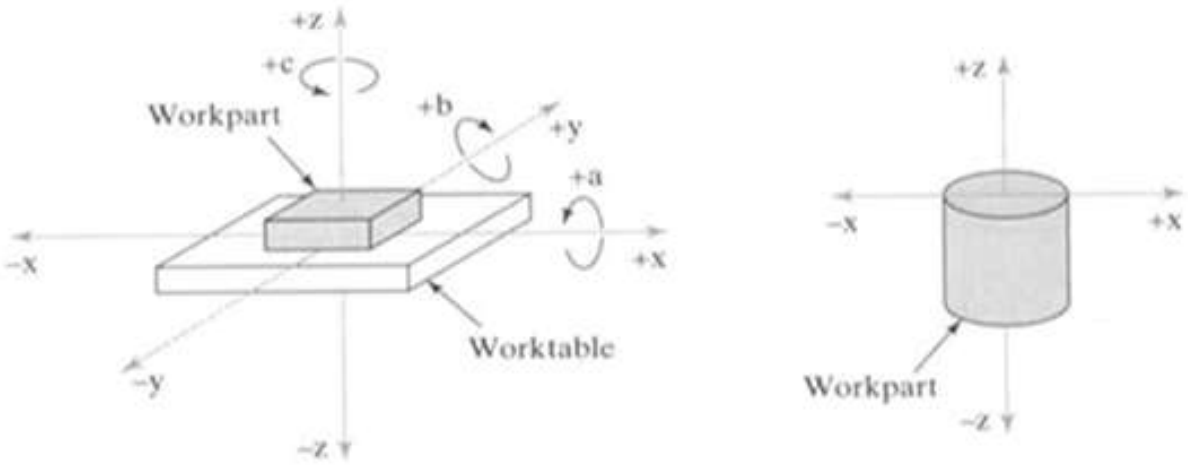



FIGURE 1.3 – les axes de machine CNC.

## 1.6 La programmation

La programmation consiste à décrire les opérations prévues dans un langage codé assimilable par le calculateur de la machine. Ce travail peut être manuel où partiellement automatisé grâce à des dispositifs d'aide à la programmation (logiciels intégrés ou externes).

Les logiciels d'aide à la programmation (logiciels de Fabrication Assistée par Ordinateur, F.A.O.) permettent d'éviter de fastidieux calculs de points. A partir du dessin de la pièce à usiner, ils calculent puis traduisent en ordres compréhensibles par la machine les déplacements à réaliser. Le programmeur en commande numérique reste responsable de l'analyse et de la structuration du programme[3].

## 1.7 Produits réels

Cette image représente un exemple des produits réels de poinçonne



FIGURE 1.4 – produit réel.

## 1.8 Les types des machines CNC

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

— Commande numérique point à point

— Commande numérique par axiale

— Commande numérique de contour nage

### 1.8.1 Commande numérique point à point

C'est la mise position de l'outil ou de la pièce par les déplacements non synchronises.

— Exemples d'opération d'usinage : perçage, alésage, lamage taraudage, petit fraisage.

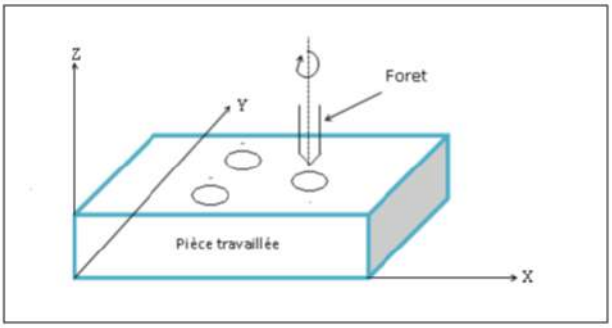



FIGURE 1.5 – Commande Numérique Point A Point

### 1.8.2 Commande numérique par axiale

Ce sont des déplacements parallèles aux axes avec les vitesses d'avance programmée. Le mouvement de coupe et de positionnement sont synchronises de façon à avoir un usinage selon des trajectoires parallèles aux axes de déplacement.[2]

Exemples d'opération d'usinage :

— Tournage

— Fraisage

— Alésage.

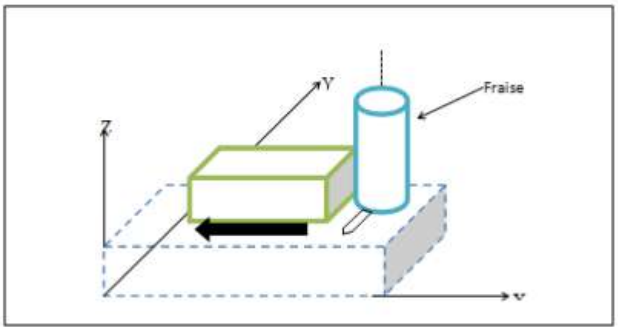



FIGURE 1.6 – Commande Praxiale.

### 1.8.3 Commande numérique de contourne

Ce sont des déplacements synchronise des divers axes avec la vitesse d'avance programmée. Les trajectoires sont décomposées en éléments de droites ou de cercles dons un ou plusieurs plans.

Exemples d'opération d'usinage : toute opération possible sur un centre de tournage ou centre d'usinage.

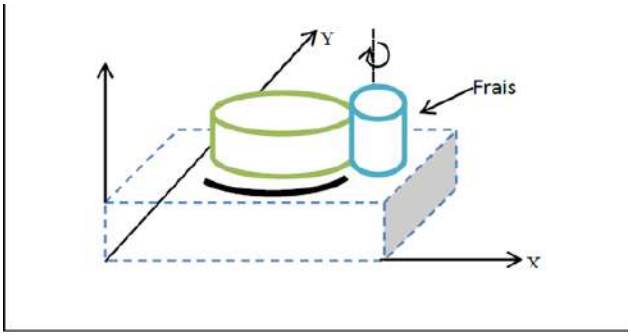



FIGURE 1.7 – Commande contourne

## 1.9 Moteur pas à pas



FIGURE 1.8 – Moteur pas à pas

### 1.9.1 définition

Un moteur pas-à-pas (stepping motor en anglais) est un moteur à Courant alternative Les moteurs pas-à pas sont des micro-moteurs à réluctance (dits également à réluctance variable), qui fonctionne grâce à l'attraction d'une masse polaire (le rotor) par un champ magnétique[1]

### 1.9.2 Types de moteur pas-à-pas

Il existe trois types des moteurs pas à pas

#### Moteur pas à pas À aimant permanent

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

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

**Avantages du moteur à aimant permanent**

— Bon marché

— Dimensions réduites

— Bon rendement

— Bon amortissement des oscillations

— Grand angle de pas (nombre de pas faible)

**Inconvénients du moteur à aimant permanent**

— Puissance faible

— Paliers en bronze ou plastique (pas de roulement)

— Couple résiduel sans courant

— Vitesse faible

### Moteur pas à pas réluctance variable

Le principe de fonctionnement de ce type de moteur pas à pas est proche de celui du moteur hybride, avec une structure dentée au rotor et au stator. Il n'y a pas d'aimant au rotor pour renforcer l'action du flux et donc pas de couple résiduel sans courant. Ce type de moteur pas à pas n'est presque plus utilisé ni fabriqué.

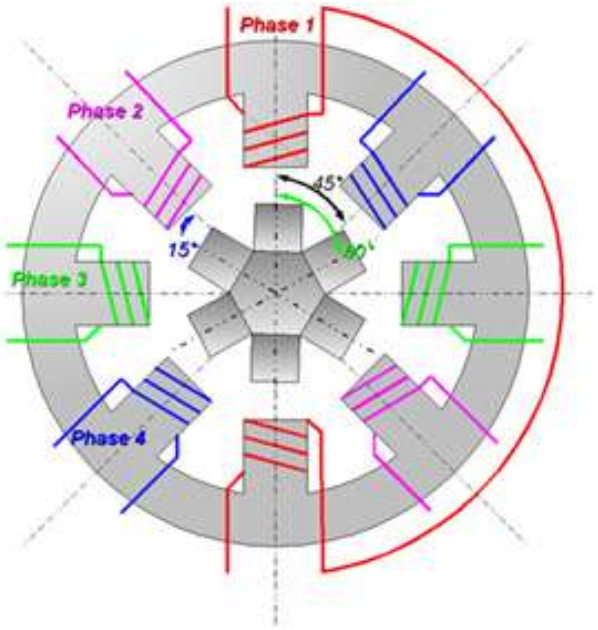



FIGURE 1.9 – Vue d'un moteur à reluctance variable

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

**Avantage de moteur à réluctance variable** Peu coûteux, d'une bonne précision. Dans l'exemple, avec seulement 4 enroulements, on obtient 24 pas (on peut facilement obtenir 360 pas). Le sens du courant dans la bobine n'a aucune importance.

### Moteur pas à pas hybride

Le moteur pas à pas "hybride" allie le principe du moteur a reluctance variable à celui du moteur a aiment permanent.Le rotor du moteur hybride comprend 2 structures régulières de dents.Ces 2 blocs sont décalés d'une $\frac{1}{2}$ dent l'un par rapport à l'autre et sont fixés de part et d'autre d'un aimant permanent magnétisé axialement. Le circuit magnétique du stator possède plusieurs pôles constitués de paquets de tôles entourés chacun d'une bobine, les

paquets de tôles se terminant par des dents.Une phase est constituée de plusieurs dents, 4 dans la plupart des cas. Tous les pôles de la phase sont décalés de façon à assurer le déphasage de 90°(quadrature).

**Avantages du moteur pas à pas hybride**

— Couple important

— Plus de puissance

— Rendement assez bon

— Courbe Start/stop assez élevée

— Bon amortissement

— Adapté au fonctionnement micro pas

— Roulement à billes pour une meilleure charge radiale et plus longue durée de vie

— Petit angle de pas

**Inconvénients du moteur pas à pas hybride**

— Inertie élevée

— Couple résiduel sans courant

— Plus couteux

— Plus volumineux

### 1.9.3 Comparison des différents types de moteurs pas à pas

TABLE 1.1 – Comparison des différents types de moteurs pas à pas

| | Aimant permanent | Hybride | Reluctance variable |
|---|---|---|---|
| Coût | Bas | Haut | Haut |
| Vitese | Bas | Très haut | Haut |
| Couple résiduel | Haut | moyen | Minimum |
| Précision du pas | Bas | Haut | Moyen |
| Amortissement | Bon | Bon | Mauvais |
| Inertie rotor | Haut | Bas | Bas |
| Rendement | Moyen | Très haut | Moyen |
| Angle de pas | 7.5°/15°/18° | 0.9°/1.8° | 1.8° |
| Nbrs de pas/tour | 48°/24°/40° | 400°/200° | 200° |

### 1.9.4 Comment fonctionne un moteur pas à pas ?

La figure 1.10 montre la structure de base d'un moteur pas à pas à aimant permanent. Le rotor est un barreau aimanté radialement. Dans ce cas simple, l'aimant est bipolaire (un pôle nord et un pôle sud). Le stator présente 2 phases (bobine1 et bobine 2).

Si on alimente la bobine 1, dans le sens +I, l'aimant va se placer en position. Si l'on supprime le courant dans la bobine 1 et qu'on l'établit dans la bobine 2 dans le sens –I, l'aimant va tourner de 90° et va se placer en position Si l'on supprime le courant dans la bobine 2 et qu'on l'établit dans la bobine 2 dans le sens –I, l'aimant va tourner de 90° et va se placer en position On rétablit enfin le courant dans la bobine 2, dans le sens +I l'aimant va se placer en position

L'aimant a donc 4 positions possibles par tour. On dit que l'angle de pas, ou le pas est de 90°. On a donc un moteur pas à pas de 4 pas par tour. Cette séquence d'alimentation des phases alimente une seule phase à la fois (1 phase ON). Aussi, en alimentant de manière appropriée 2 phases à la fois (2 phases ON) on obtiendrait également 4 positions stables décalées de 90° entre elles mais à 45° par rapport au cycle 1 phase ON.

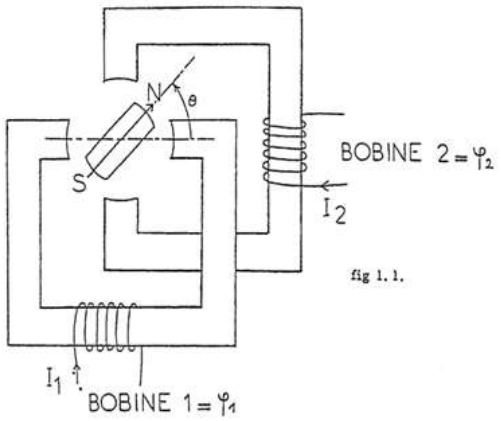



FIGURE 1.10 – Structure d'un moteur pas à pas (montage bipolaire)

### 1.9.5 Domaines d'application d'un moteur pas à pas

En peut utiliser les moteurs pas à pas dans les applications suivantes :

— horlogerie (moteurs de type Lavet ou variantes, monophasés à aimants)

— péri-informatique :

— avance des têtes de lecture de disques mémoire (dans les disques durs actuels, on utilise des actionneurs à bobine mobile plus rapides)

— imprimantes, scanners, machines à écrire...

— distributeurs automatiques :

— billets de banques

— boissons, alimentation

— lecteurs : de cartes magnétiques (tickets métro, publiphones...)

— tables de positionnement X, Y, scanners médicaux

— accessoires automobiles

— orientation d'antennes

— certains appareils photos et caméscopes

### 1.9.6 Les modes de commande du moteur pas à pas biphasés

La rotation du moteur s'effectue par une séquence d'alimentation des divers enroulements de phases en unipolaire ou en bipolaire. La génération d'un couple est intimement liée aux courants circulant dans les phases. Chaque commutation entraîne d'une phase transitoire dont la durée peut varier selon le type de contrôle (tension ou courant). Dans cette section seule le régime quasi statique est étudié. Par quasi statique on entend le comportement du moteur autour de ses points d'équilibre. Pour des raisons pédagogiques et vu leur forte utilisation, nous nous limiterons ici aux moteurs biphasés et par conséquent aux moteurs pas à pas à aimants permanents et hybrides.

#### Mode 1 : commande en pas entier, une phase ON

Une seule phase est alimentée à la fois par injection d'un courant nominal In. C'est dans ce mode que le fabricant définit le pas angulaire du moteur. La Figure 1.11 illustre la séquence des alimentations de phases pour un moteur à aimants permanents.

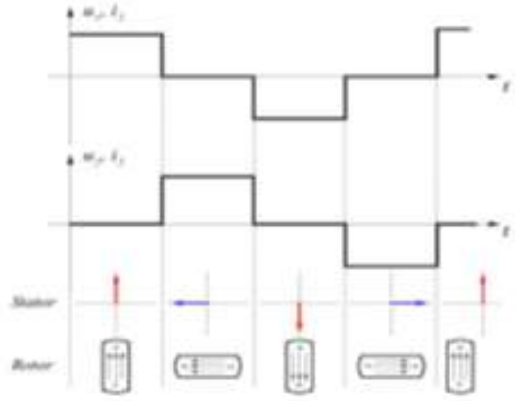


FIGURE 1.11 – la séquence des alimentations de phases pour un moteur à aimants permanents (mode 1)

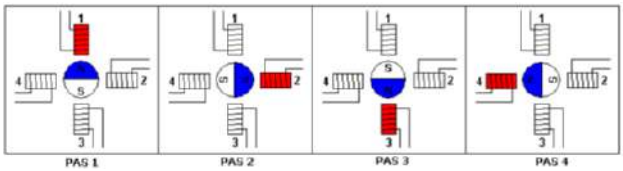



FIGURE 1.12 – Commande dans le mode 1

**Mode 2 : commande symétrique en pas entier, deux phases ON**

Deux phases du moteur sont alimentées simultanément par un courant nominal In. Le couple moteur est multiplié par $\sqrt{2}$ par rapport au contrôle selon le mode 1. Le pas angulaire est identique au mode 1 mais les positions d'équilibre sont intermédiaires aux précédentes. La Figure 1.13 illustre la séquence des alimentations de phases pour un moteur à aimants permanents.

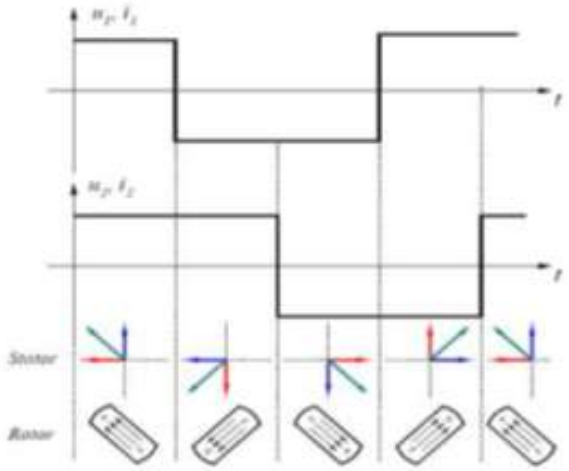


FIGURE 1.13 – la séquence des alimentations de phases pour un moteur à aimants permanents (mode 2)

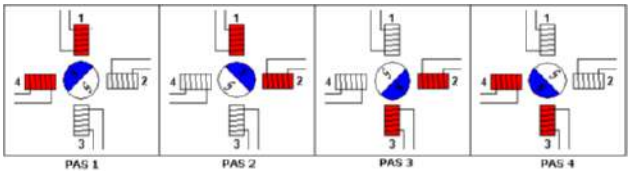



FIGURE 1.14 – Commande dans le mode 2

### Mode 3 : commande asymétrique en demi pas, une ou deux phases ON

La combinaison en alternance du mode 1 et du mode 2 permet le fonctionnement en demi-pas. Pou effectuer un déplacement donné, il faut doubler le nombre d'impulsions de commande. Les couples quasi statiques ne sont pas identiques selon si une ou les deux phases sont alimentées. La Figure 1.15 illustre la séquence des alimentations de phases pour un moteur à aimants permanents.

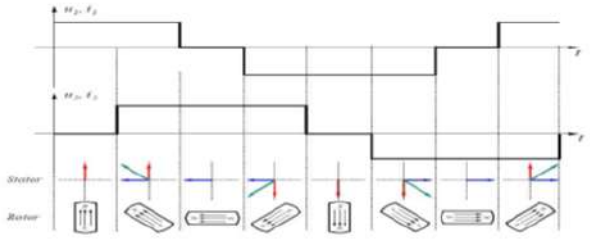


FIGURE 1.15 – la séquence des alimentations de phases pour un moteur à aimant permanents (Mode 3)

### Mode 4 : commande symétrique en demi pas, une ou deux phases ON

Pour un fonctionnement régulier en demi-pas, il faut augmenter le courant d'un facteur $\sqrt{2}$ lorsqu'une seule phase est alimentée. On a ainsi les performances de couple du mode 2 mais avec deux fois plus de pas. La Figure 1.16 illustre la séquence des alimentations de phases pour un moteur à aimants permanents.

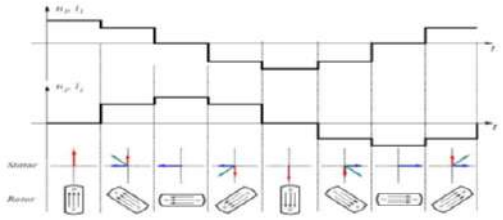


FIGURE 1.16 – la séquence des alimentations de phases pour un moteur à aimant permanents (Mode 4)

**Mode 5 : commande micro pas**

Ce mode consiste à multiplier les positions intermédiaires en asservissant le courant de chaque phase. Il constitue une généralisation du mode 4. Si l'on contrôle les courants de phases dans le moteur selon les lois :

$$i_1 = \sqrt{2} In \cos(\theta e \theta) \tag{1.1}$$

$$i_2 = \sqrt{2} In \sin(\theta e \theta) \tag{1.2}$$

Et en admettant des distributions des constantes de couple suivantes :

$$K_1(\theta e) = -KT \sin(\theta e) \tag{1.3}$$

$$K_2(\theta e) = -KT \cos(\theta e) \tag{1.4}$$

Le couple devient :

$$T = -\sqrt{2} InKT \sin((\theta e - \theta eO)) \tag{1.5}$$

$\theta eO$ représente la position angulaire électrique d'équilibre pour des courants définis par les relations (1.1 et 1.2). En donnant à $\theta eO$ n valeur équidistante de $2n/\pi$, on multiplie par n le nombre déposition d'équilibre en mode 1. On obtient apparemment une précision aussi grande que l'on veut, mais celle-ci n'a de sens que si le couple de charge est très bien connu et sans fluctuation. De plus, une telle précision ne peut être atteinte qu'avec une loi de couple moteur parfaitement sinusoïdale, donc avec des moteurs spécialement construits et

avec une alimentation en courant très précise, parfois réalisée avec des amplificateurs de puissance linéaires. Pour des moteurs classiques, le mode 5 permet une rotation continue entre deux positions d'équilibre, en amortissant les oscillations à basse fréquence de vitesse, causes de certaines instabilités dans une rotation incrémentale. En conclusion le mode 5, très performant, n'est pas économique.

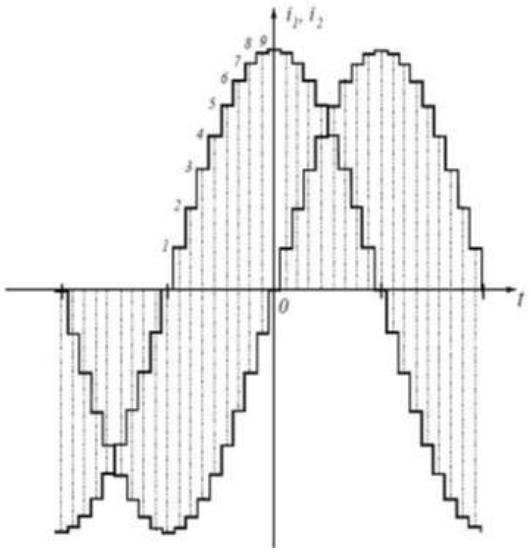



FIGURE 1.17 – la séquence des alimentations de phases pour un moteur à aimant permanents (Mode 5)

### Caractéristique dynamique de moteur pas à pas

Les moteurs pas à pas ne sont pas des moteurs rapides, les plus rapides dépassent rarement la vitesse maximale de 3000 tr/min.

Cette "lenteur" aidant, et ces moteurs étant naturellement sans balais (la majorité des moteurs pas à pas de haute qualité est de plus équipée de roulements à billes), ces moteurs ont une durée de vie extrêmement longue, sans nécessité d'entretien.

Cette première partie a été consacrée à l'étude de différents types de moteurs pas à pas ainsi que de différents modes de commande. Cette étude a montré que pour générer un mouvement incrémental de translation. Le moteur pas à pas linéaire à attaque directe représente. En effet, cette solution se distingue de celle utilisant un moteur pas à pas rotatif

par l'absence d'organe de transmission, réduisant le nombre de composant augmentant ainsi la flexibilité du système. Les modes de commande présentés dans cette partie pour des moteurs pas à pas rotatifs peuvent être adoptés et adaptés au cas des structures linéaires

## 1.10 les tiges filetées

### 1.10.1 la tige filetée

Une tige filetée est en mécanique le composant mâle d'un système vis/écrou destiné à l'assemblage de pièces ou à la transformation de mouvement.

### 1.10.2 le filetage

Un filet est une structure hélicoïdale utilisée pour la transformation rotation/translation. Cette structure peut se trouver enroulée autour d'un cylindre plein (tige filetée, par exemple une vis) ou creux (trou taraudé, par exemple un écrou). Le pas de vis mesure la distance entre une forme de la structure et sa plus proche répétition trouvée par translation selon l'axe du cylindre. Dans un système vis-écrou, selon la valeur du pas de vis, le diamètre du cylindre et le coefficient de frottement des matériaux utilisés,

— soit la rotation seule peut entraîner la translation (vis d'assemblage)

— soit la translation seule peut entraîner la rotation (manche de tournevis ≪ automatique ≫)

— soit le système est réversible

Le choix du pas de vis (ainsi que des autres paramètres) dépend donc de l'utilisation envisagée. Le plus grand nombre des applications se trouve dans la première catégorie (la rotation produit la translation, de manière irréversible) Le filetage peut se rencontrer également sur des cônes ou tronc de cône. Ils ne sont généralement employés que dans une zone d'engagement d'une pièce filetée sur cylindre (vis à bois, à tôle.)

## 1.11 Conclusion

Dans ce chapitre on a présenté les machines CNC, les commandes les plus utilisées, les Classifications des machines CNC et Composants d'une CNC. L'utilisation de la machine

CNC présente un grand intérêt pour la fabrication en petite et moyenne série ainsi que pour les formes complexes des pièces à usiner.

Et définirez les moteurs pas à pas avec les types et le fonctionnement avec les tiges filetées. Les machines CNC sont commandées par des codes spécifiques qui seront traité dans le chapitre trois.

# Chapitre 2

# Conception modèle CNC avec SolidWorks

## 2.1 historique

SolidWorks Simulation a d'abord été commercialisé sous le nom COSMOS/Works puis COSMOS Works avant de devenir SolidWorks Simulation en 2008. Le logiciel a été développé par Structural Research and Analysais Corporation (SRAC), et la première version a été disponible dans SolidWorks 1995. Le 30 mars 2001, SRAC a été racheté par Dassault Systèmes, qui avait auparavant racheté SolidWorks Corp. en 1997. De ce fait, COSMOS Works est devenu un produit SolidWorks à part entière et non plus un produit partenaire.[7]

## 2.2 Introduction

En dessin technique, un dessin d'ensemble est la représentation d'un mécanisme complet (ou partiel) permettant de situer chacune des pièces qui le composent. Les pièces sont dessinées, à une échelle dépendant des dimensions réelle du mécanisme et de la feuille accueillant le dessin, à leur position exacte (assemblées), ce qui permet de se faire une idée concrète du fonctionnement du mécanisme.

SolidWorks Simulation est un logiciel de calcul par éléments finis intégré à SolidWorks. Il utilise les fichiers pièces et assemblages auxquels il intègre des études dans chacune desquelles sont définis les matériaux, les chargements, les conditions aux limites ainsi que les paramètres

de maillage et d'analyse. Les résultats sont contenus dans un fichier portant l'extension.

## 2.3 Fonctionnement

SolidWorks est un modeleur 3D utilisant la conception paramétrique. Il génère 3 types de fichiers relatifs à trois concepts de base : la pièce, l'assemblage et la mise en plan. Ces fichiers sont en relation. Toute modification à quelque niveau que ce soit est répercutée vers tous les fichiers concernés. Un dossier complet contenant l'ensemble des relatifs à un même système constitue une maquette numérique.[5] De nombreux logiciels viennent compléter l'éditeur SolidWorks. Des utilitaires orientés métiers (tôlerie, bois, BTP...), mais aussi des applications de simulation mécanique ou d'image de synthèse travaillent à partir des éléments de la maquette virtuelle.

### 2.3.1 Les pièces

La pièce est l'objet 3D monobloc. La modélisation d'une telle entité dépendra de la culture de l'utilisateur. Comme de nombreux logiciels conviviaux, SolidWorks permet d'aboutir à un même résultat apparent par des voies souvent différentes. C'est lors de la retouche de ces fichiers ou de leur exploitation qu'on appréciera la bonne méthode. Une pièce est la réunion d'un ensemble de fonctions volumiques avec des relations d'antériorité, des géométriques, des relations booléennes (ajout retrait)... Cette organisation est rappelée sur l'arbre de construction. Chaque ligne est associée à une fonction qu'on peut renommer à sa guise.

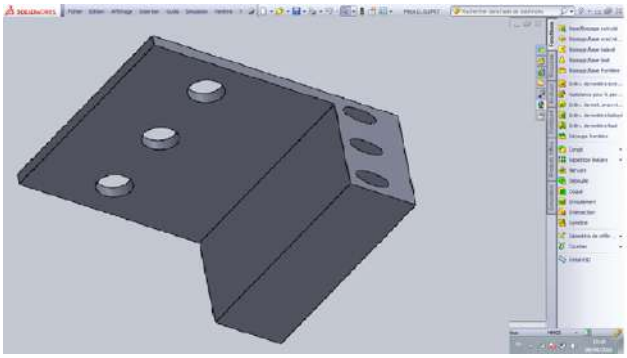


FIGURE 2.1 – une pièce avec solidworks

Parmi les fonctions génératrices on trouve :

— l'extrusion : déplacement d'une section droite dans une direction perpendiculaire à la section. La section est définie dans une esquisse (qui apparaît alors dans l'arbre de création comme élément générateur de la fonction). Cette esquisse contient l'ensemble des spécifications géométriques (cotation) nécessaires à la complète définition de la section. Cet ensemble de cotes auquel il faut ajouter la (ou les) longueur d'extrusion constitue l'ensemble des paramètres de la fonction ; il est possible de les modifier une fois la fonction validée.

— la révolution : déplacement d'une section droite autour d'un axe, ou extrusion suivant un cercle ou un arc de cercle.

— le balayage : déplacement d'une section droite le long d'une ligne quelconque. Lorsque la génératrice de balayage est gauche, l'esquisse est en 3 dimensions.

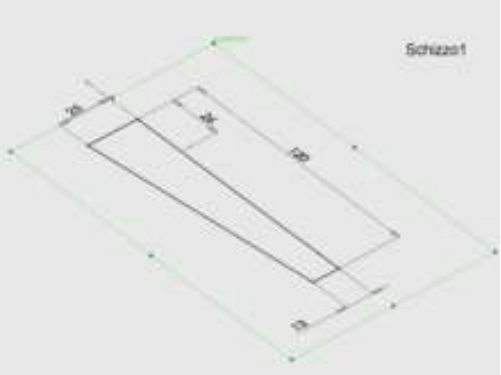



FIGURE 2.2 – Différentes fonctions logicielles

D'autres fonctions, plutôt orientées métier intègrent des notions qu'il serait fastidieux de modéliser :

— congés et chanfreins,

— nervure,

— dépouille,

— coque (permettant d'évider un objet en lui conférant une épaisseur constante),

— trous normalisés (perçages, mortaises...),

— plis de tôle...

Des fonctions d'ordre logiciel comme la répétition linéaire, circulaire, curviligne ou par symétrie... Les dernières versions autorisent la réalisation de pièces momentanément disjointes, ce qui permet de concevoir un objet fonctionnellement, c'est-à-dire en définissant d'abord les éléments fonctionnels, puis en joignant les différentes parties par de la matière (nervures, carter...).

L'édition de familles de pièces est possible en associant à SolidWorks, le tableur Microsoft Excel : Un tableau devient ainsi éditeur des références (lignes) donnant la valeur des paramètres variables des fonctions (colonnes) :

— valeur de certaines cotes ;

— valeur de certaines propriétés (nombre d'occurrence...) ;

— état de suppression d'une fonction.

Ainsi, il est possible d'obtenir à partir d'un seul fichier de type pièce, l'ensemble des modèles de vis d'assemblage (toutes forme de tête ou dimensions), ou encore toutes les combinaisons de briques Lego (1x1, 1x2 etc.). Avec tous ces outils, la méthode de conception d'une pièce très proche du métier du concepteur qui ne se soucie plus de savoir où placer les traits du dessin mais les formes de la pièce. Les possibilités d'éditions sont complétées par un ensemble d'outils de mesures géométriques. Ainsi il est possible de connaitre le volume de la pièce, son poids, la position décentre, sa matrice d'inertie, la surface...

### 2.3.2 les assemblages

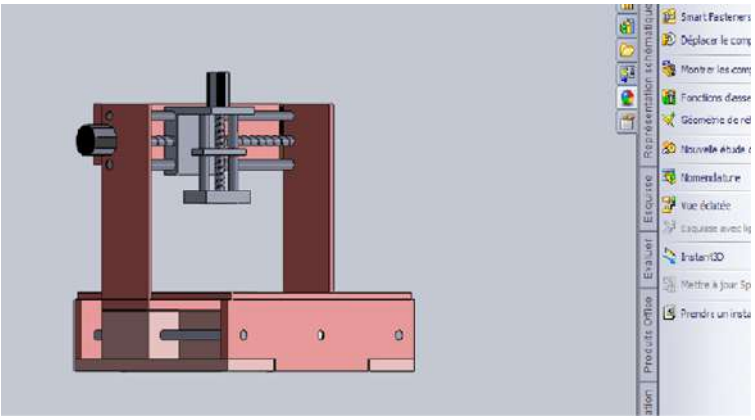


FIGURE 2.3 – Assemblage avec solidworks

Les assemblages sont obtenus par la juxtaposition de pièces. La mise en position de pièces est définie par un ensemble de contraintes d'assemblage associant, deux entités respectives par une relation géométrique (coïncidence, tangence, coaxial ite...). Dans une certaine mesure, ces associations de contraintes s'apparentent à la liaison mécanique entre les pièces. Le mécanisme monté, s'il possède encore des mobilités, peut être manipulé virtuellement. On peut alors aisément procéder à des réglages à l'aide des différents outils disponibles (déplacement composants, détection de collision ou d'interférence, mesure des jeux, etc.)

Comme pour les pièces, la gestion de l'ensemble est portée par un arbre de création qui donne accès à l'arbre de création de chaque pièce. Il est d'ailleurs possible d'assembler des assemblages,[5] donc de former des sous-groupes de pièces. Cette opération étant préalable ou en cours d'édition.

L'intérêt de cet outil, c'est qu'il rend possible la création d'une pièce dans l'assemblage. Il propose la même méthode au concepteur que celle qu'il appliquait sur la table à dessin : tout concevoir en même temps. En effet, à part sur les petits ensembles simples (ou déjà définis), il n'est pas raisonnable de concevoir chaque pièce dans son coin pour corriger ensuite, lors de l'assemblage, les problèmes éventuels d'interférence, ou de coïncidence. La conception intégrée lie automatiquement les géométries des pièces entre elles, si bien qu'une modification sur une, est automatiquement répercutée sur les autres. Alors l'édition de pièce est la

conséquence de l'édition de l'ensemble. De plus, Solidworks gère les références cycliques. En effet, non seulement il est possible de créer des pièces enfants à partir de la pièce parent, mais il est aussi possible de modifier la pièce parent en fonction des pièces enfants sans générer de boucle de mise à jour.

### 2.3.3 La miss en plan

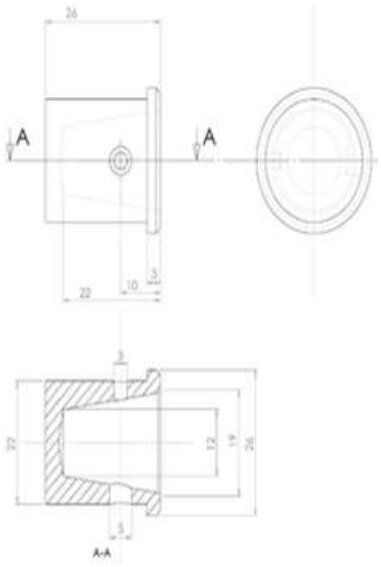



FIGURE 2.4 – Dessin de définition établi avec SolidWorks

### 2.3.4 Dessin de définition établi avec SolidWorks

Les mises en plan concernent à la fois les pièces (dessin de définition) ou les assemblages (dessin d'ensemble). Pour aboutir à un plan fini d'une pièce on peut estimer mettre 2 fois moins de temps qu'avec un outil DAO (temps de conception et exécution du dessin). En effet, en DAO, chaque trait est indépendant, et c'est au dessinateur de savoir quelles entités graphiques sont concernées par une modification. Le logiciel 3D ne fait qu'une projection de l'objet. Les modifications éventuelles sont opérées sur l'objet représenté, et ne concernent pas directement le plan.

### Vues

La projection sur plan du modèle ne pose aucun problème. Aujourd'hui il est très facile d'obtenir un plan, forcément juste (avec un logiciel de DAO il est possible d'éditer un plan faux !). Les vues en coupes, les vues partielles, perspectives, sont exécutées d'un simple clic. Les seuls problèmes encore rencontrés concernent la représentation des filetages et taraudages dans les assemblages. De plus, chaque vue peut être exécutée avec un habillage différent, filaire, conventionnel ou ombré rendant encore plus accessible la lecture de plans aux non initiés.

### Cotation

La cotation regroupe l'ensemble des spécifications géométriques définissant la pièce. Bien sûr, les paramètres déclarés des esquisses en font partie. Ils peuvent être automatiquement intégrés, de manière équilibrée, à la mise en plan. À ce niveau, il est encore possible de modifier la pièce en changeant la valeur des cotes. L'indépendance de ces paramètres, se rapproche du principe dit d'indépendance (cotation GPS). Cependant, la mise en forme définitive de la cotation demande encore un peu de travail (avec un éditeur DAO ou sur le calque aussi !).

### Fond de plan

SolidWorks par défaut propose ses propres fonds de plan. Mais il est possible de les faire soi-même ,[4] Il existe deux types de fond de plan :

— Le fond de plan statique, où il faut remplir à la main chacun des champs.

— Le fond de plan dynamique, où il se remplit automatiquement suivant les paramètres mis dans l'assemblage ou dans la pièce.

— Des modèles sont proposés (équivalent du .dot de Word).

### Nomenclature

Le fichier assemblage contient chacune des pièces qui composent l'assemblage, on peut donc sortir de façon automatique la nomenclature appartenant à la maquette 3D.

## 2.4 Conception modèle CNC avec solidworks

Dans notre projet on a choisi de crée une machine a commande numérique a trois-axes en utilisant le logiciel solidworks Pour former cette machine il faut passe par deux étapes nécessaires étapes de Conception de pièces et étape d'assemblage.

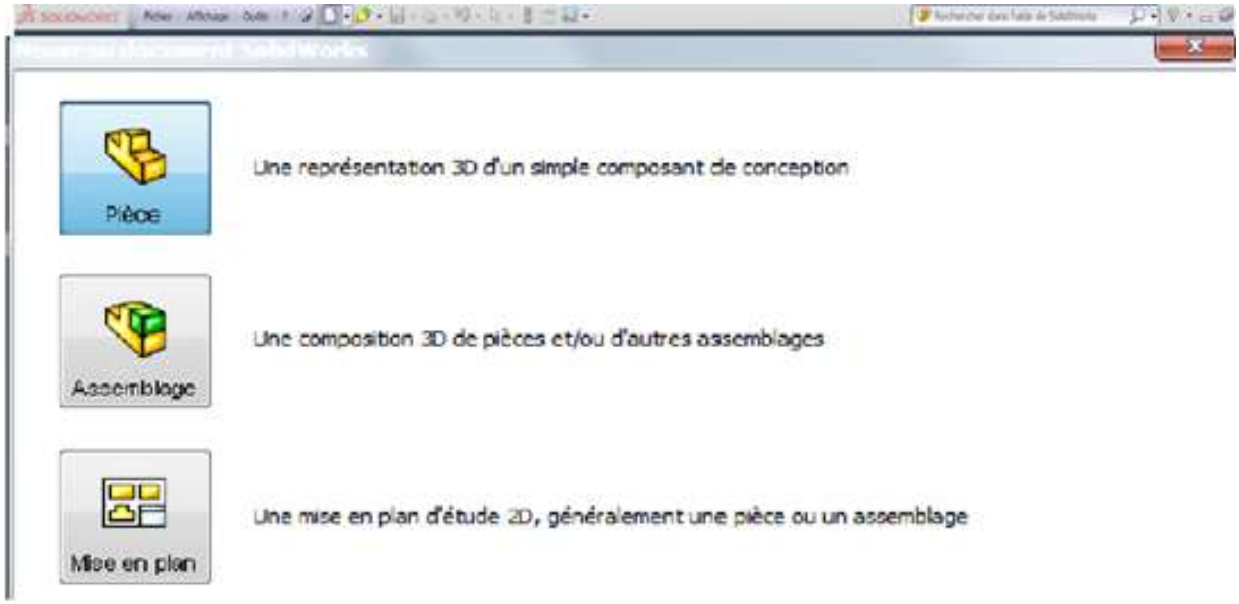



FIGURE 2.5 – Les trois concepts de base (SOLIDWORKS)

### 2.4.1 Les pièces

On a plusieurs différente pièces dont les plus utilise sont :

#### la table initiale

C'est la plus grande pièce qui maintient l'ensemble de la structure et elle sert come une pose de la pièce sculpté En a pondre plan de face Pour crée la table initiale en a choisi la (nouveau file) et sélectionner la fonction pièces sélectionner esquisse et pondre esquisse rectangle . Et sélectionner la fonction bossage extradai.

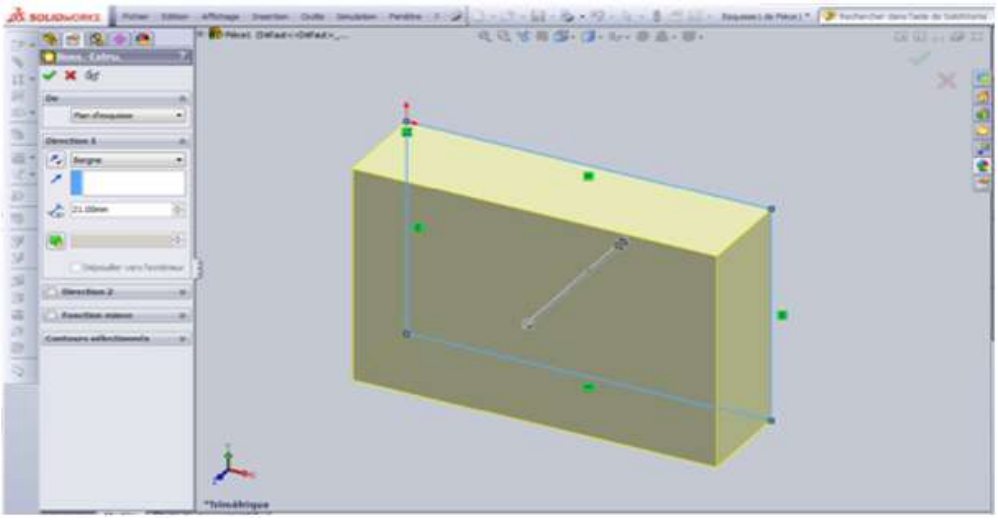


FIGURE 2.6 – création de la table

Et pondre les distance et clique sur OK pour terminiez l'opération

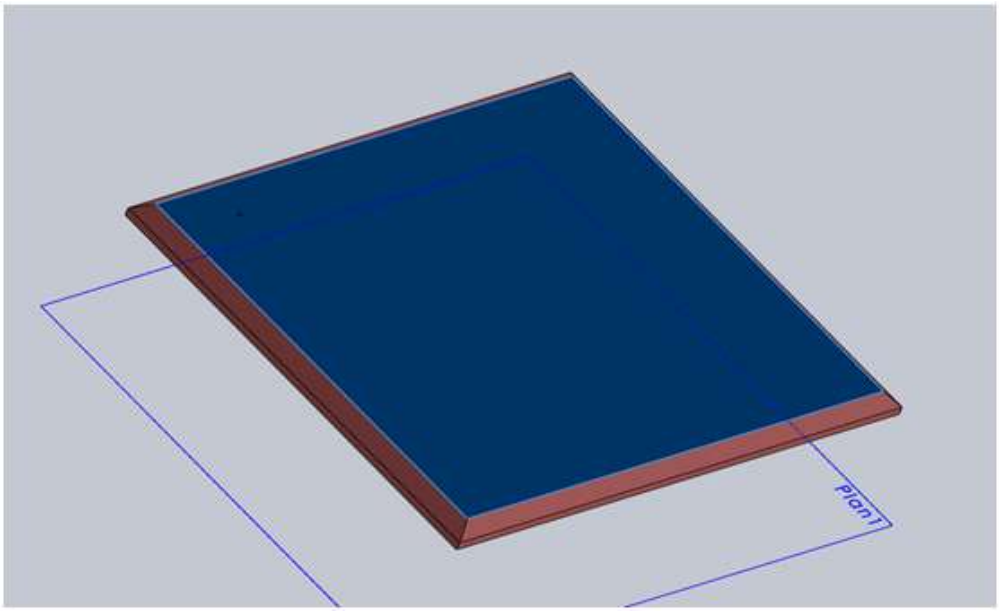



FIGURE 2.7 – La table initiale

### Les tiges d'axes

On a trois axes dont chaque un comporte trois tiges et l'axe médian elle doit être filetée et fixé au moteur d'axe pour démarré suite au a la mise en marche du moteur et permettre une activation de la structure de L'axe Pour crée les tiges sélectionner l'esquisse de cercle et la fonction de bossage.

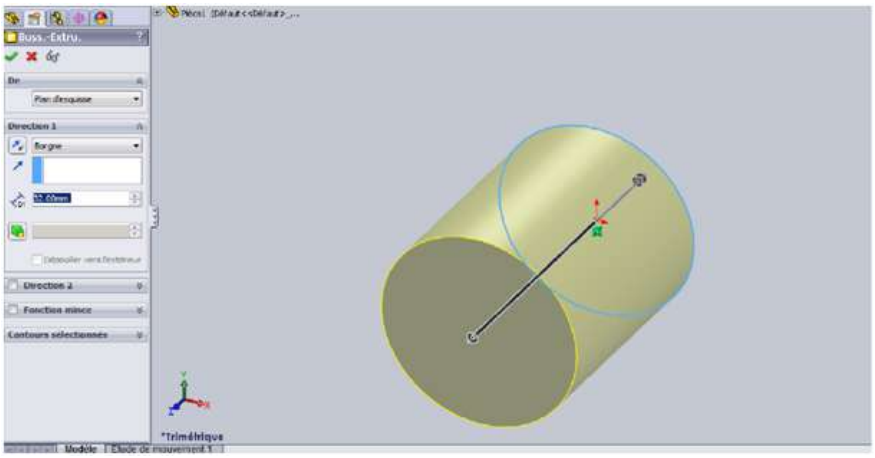


FIGURE 2.8 – création d'une tige simple

Et pondre les distance et clique sur OK pour terminiez l'opération.

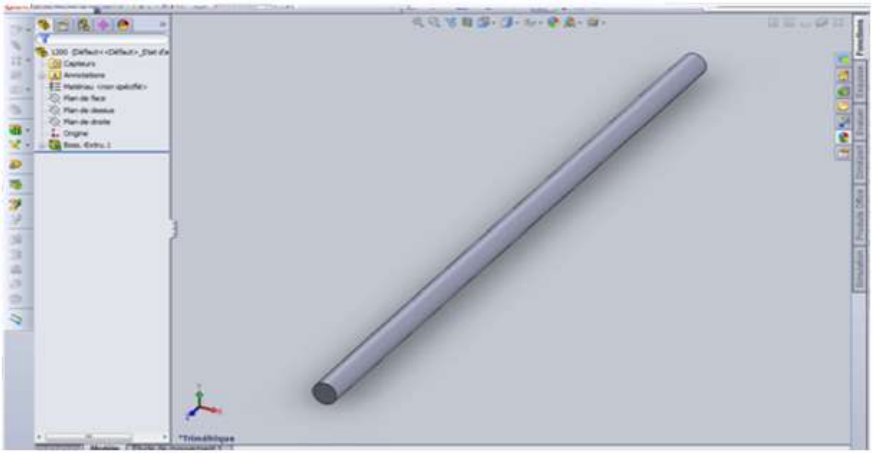


FIGURE 2.9 – La tige simple

**filetée la tige** Pour filetée une tige il faux :

— Sélectionner plan de face

— Créer une esquisse et dessiner

— Le cercle, le point et profil

— Click sur Hélice et spirale

— Click sur Enlèvement de matière .balayé

On obtient le filetage

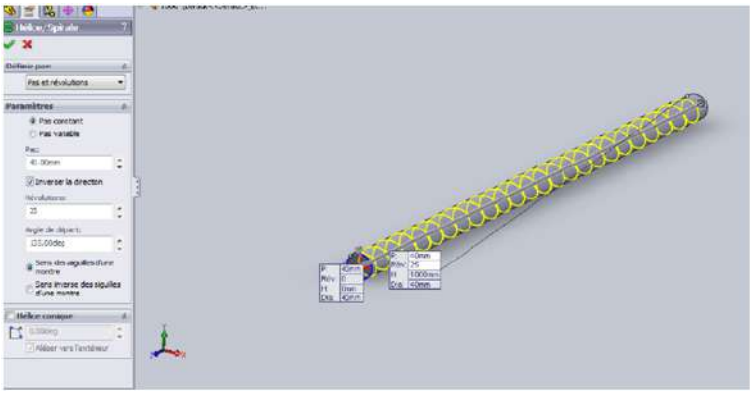


FIGURE 2.10 – création tige filetée

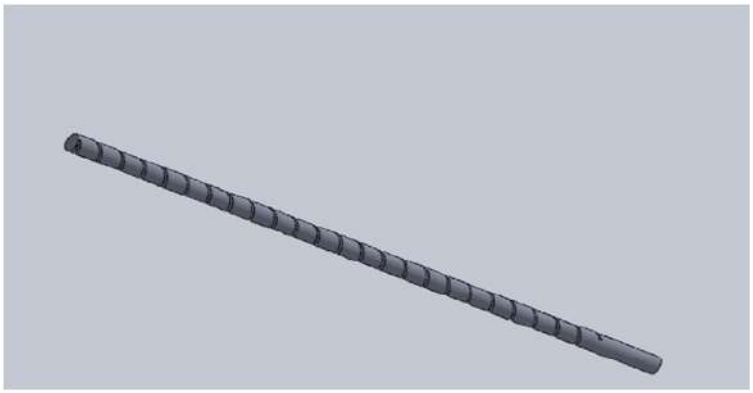


FIGURE 2.11 – tige filetée

**les moteurs**

Cette machine comporte trois moteurs pas à pas dont chaque un fait tourné est propre axe selon notre commande Pour crée les moteurs sélectionner l'esquisse de polygone et la fonction de bossage.

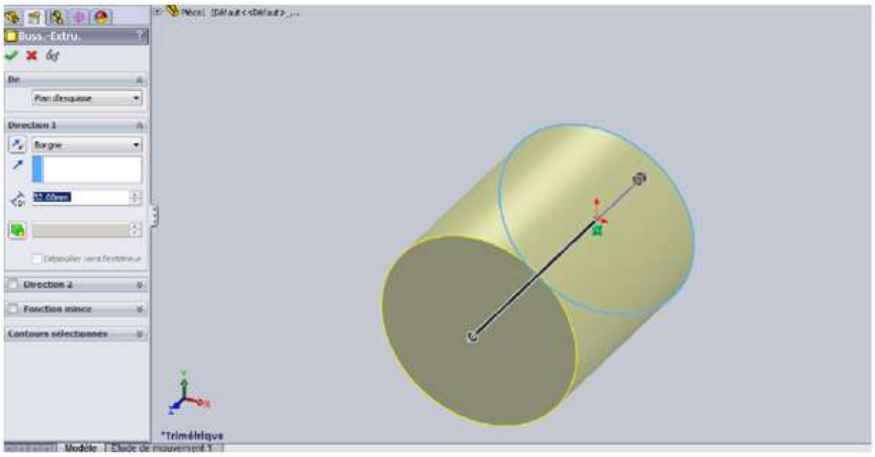


FIGURE 2.12 – création moteur

Et pondre les distance et clique sur OK pour terminiez l'opération.

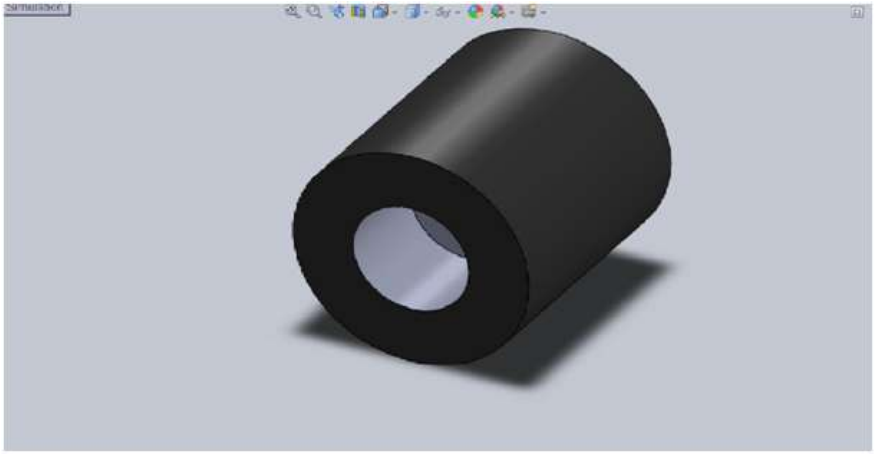


FIGURE 2.13 – Moteur pas à pas imaginaires

### 2.4.2 Les dimensions des pièces

TABLE 2.1 – Les dimensions des pièces

| Pièces | Distance X | Distance Y | Distance Z |
|---|---|---|---|
| La table | 1150mm | 1000mm | 50mm |
| Les tiges | 1200mm | 20mm | 20mm |
| Les moteurs | 120mm | 70mm | 70mm |

### 2.4.3 l'assemblage

L'assemblage c'est la deuxième étape dans la création, pour assemblé les pièces pondre nouveau file d'assemblage.

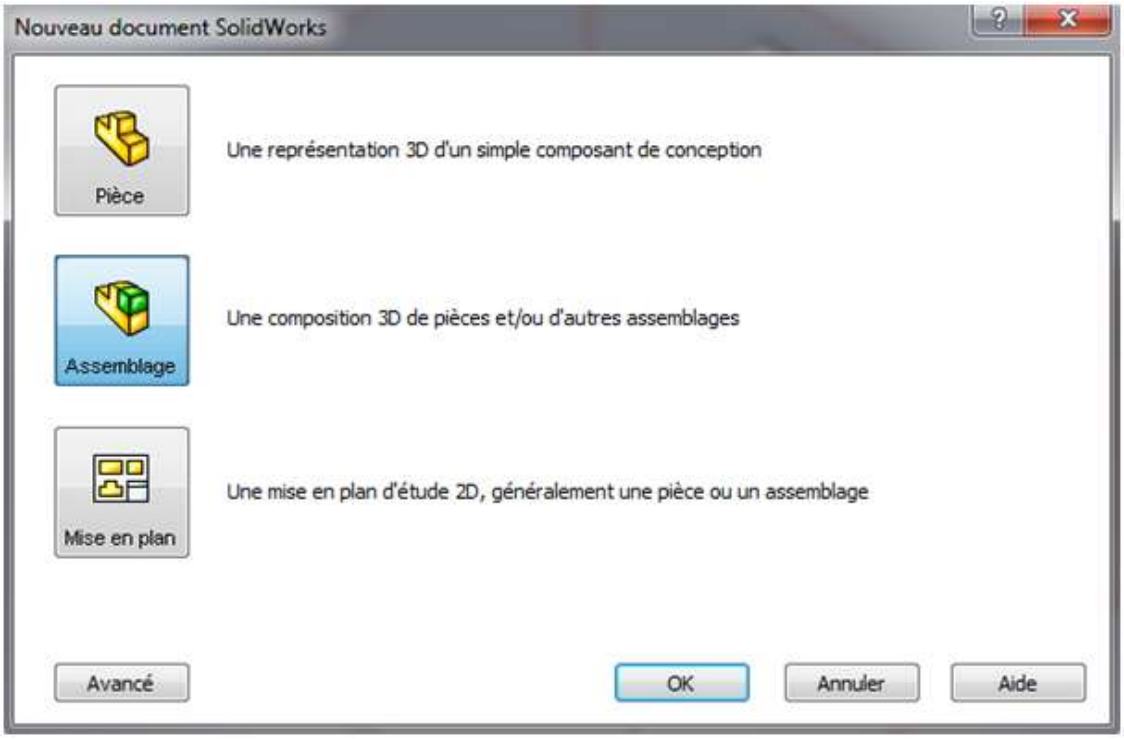



FIGURE 2.14 – Fenêtre pour sélectionner assemblage

Clique sur l'option parcourir et choisi les pièces d'assemblages. Et clique sur la fonction contacter et assembler les pièces avec les trois dimensions.

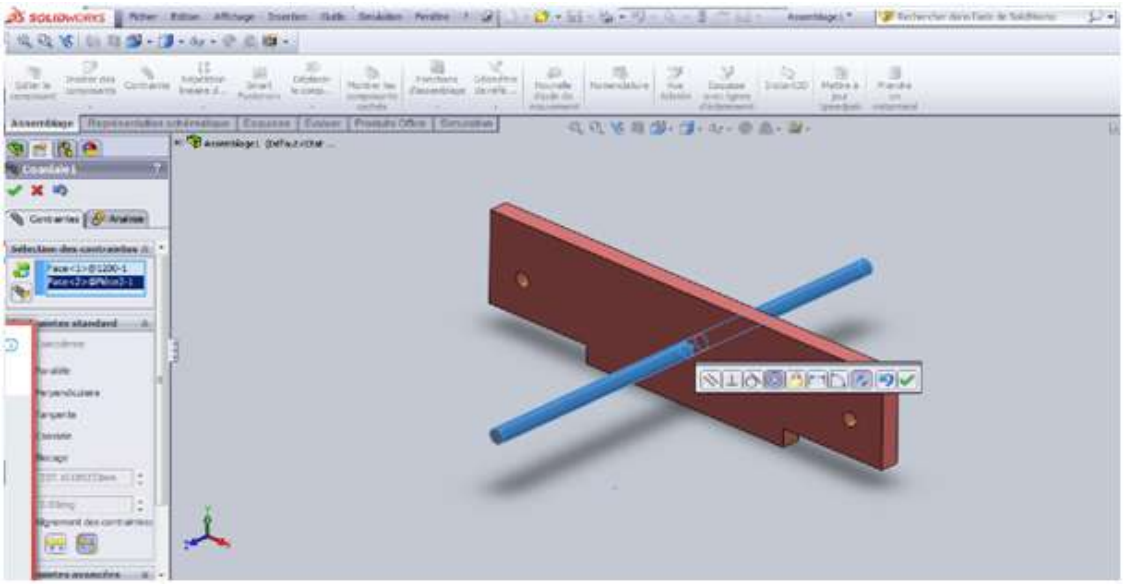


FIGURE 2.15 – assemblage deux pièces

Et pondre les distance d'assemblage et clique sur OK pour terminiez l'opération.

## 2.5 Simulation du modèle (mouvement)

Le bute de notre projet est comment simulée la machine a commande numérique a partir de plusieurs logiciel. Dans logiciel solidworks une option : étude de mouvement. La première chose de l'étude de mouvement : cochée la fonction (solidworks motion) :

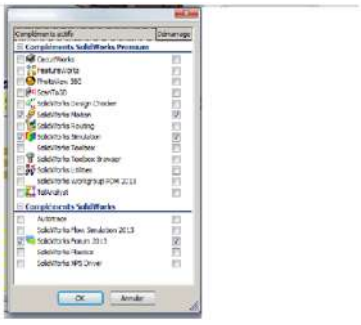


FIGURE 2.16 – Fenêtre pour cocher solidworks motion

Pour l'étude de mouvement de ma machine a commande numérique :

— Clique étude de mouvement.

— Choisi la fonction moteur : déplacer un composant comme s'il état par un moteur active.

— Commander le moteur choisi avec les types moteurs séculaires ou bien moteurs linéaires.

— Choisi la pièce et la direction de moteur.

— Commander la vitesse.

— Termenie l'opération.

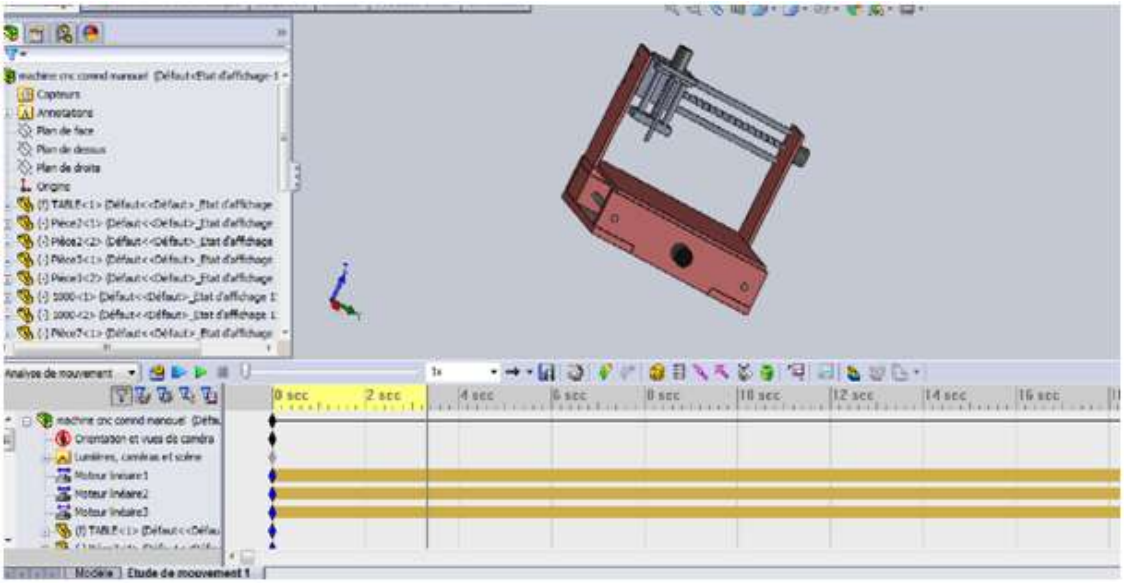



FIGURE 2.17 – La simulation de mouvement

Il ya plusieurs d'option dans l'étude de mouvement :

— Calculer l'étude de mouvement.

— Choisi plusieurs moteurs en même temps.

— Vitesse de lecteur : commande la vitesse de lecteur.

— Le diagramme pour l'agrandir.

— Orientation et vues de camera.

## 2.6 Conclusion

Dans ce chapitre nous avons définir logiciel solidworks et le fonctionnement e en même temps montré les étapes de concéption pour chaque pièce et après assemblés les pièces et dernièrement la simulation avec le mouvement.

# Chapitre 3

# La programmation avec logiciel labview

## 3.1 Introduction

LabVIEW (Laboratory Virtual Instrument Engineering Workbench) est un langage de programmation graphique qui utilise des icônes au lieu de lignes de texte pour créer des applications. Contrairement aux langages de programmation textuels où ce sont les instructions qui déterminent l'ordre d'exécution du programme, LabVIEW utilise la programmation par flux de données ; c'est le flux des données transitant par les noeuds sur le diagramme qui détermine l'ordre d'exécution des VIs et des fonctions.

Les VIs, ou instruments virtuels, sont des programmes LabVIEW qui imite les instruments physiques.

Dans LabVIEW, vous construisez une interface utilisateur à l'aide d'un ensemble d'outils et d'objets. L'interface utilisateur d'un VI est appelée la face-avant. Ensuite, vous créez le code en utilisant des représentations graphiques de fonctions pour commander les objets de la face-avant. Ce code source graphique est aussi appelé code G ou code du diagramme.

Le diagramme contient ce code. D'une certaine manière, le diagramme ressemble à un organigramme.

## 3.2 Historique

C'est en 1986 que la première version de LabVIEW voit le jour sur Macintosh. Il s'ensuit un travail incessant pour ajouter des fonctionnalités :

— 1986 : LabVIEW 1.0, sous Mac OS

— 1990 : LabVIEW 2.0, Langage compilé, optimisation des performances

— 1992 : LabVIEW 2.5, sous Windows 3.1 et Solaris

— 1993 : LabVIEW 3.0

— 1994 : LabVIEW 3.0.1, sous Windows NT

— 1994 : LabVIEW 3.1

— 1995 : LabVIEW 3.1.1, intégration de l'Application Builder

— 1996 : LabVIEW 4.0

— 1997 : LabVIEW 4.1

— 1998 : LabVIEW 5.0, multitâche, conteneurs ActiveX, assistant pour les cartes d'acquisition (DAQ) et assistant pour le contrôle d'instrument

— 1999 : LabVIEW 5.1, portage sous Linux ; première version de LabVIEW temps réel

— 2000 : LabVIEW 6.0i, optimisation pour le partage d'interfaces sur internet, contrôles au look 3D, référence d'un contrôle

— 2001 : LabVIEW 6.1, optimisation et corrections

— 2003 : LabVIEW 7.0, VI Express, première version des toolkits pour FPGA, pour Palm OS et pour Windows Mobile

— 2004 : LabVIEW 7.1, traduit en Français, Allemand et Japonais

— 2005 : LabVIEW 8.0, Explorateur de projet, XControls, variables partagées

— 2006 : LabVIEW 8.20, programmation orientée objets

— 2007 : LabVIEW 8.5, première version du toolkit Statechart, gestion optimisée des machines multicœur

— 2008 : LabVIEW 8.6, nettoyage automatique des diagrammes, quick drop (déposé rapide), services Web

— 2009 : LabVIEW 2009 (9.0), extraits de VIs, récursivité native, nouvel éditeur d'icônes, boucles For parallélisables

— 2010 : LabVIEW 2010, Bibliothèques de projet empaquetées, Fonctions de flux réseau (Transfert de données en continu)

— 2011 : LabVIEW 2011

— 2012 : LabVIEW 2012

— 2013 : LabVIEW 2013

— 2014 : LabVIEW 2014

— 2015 : LabVIEW 2015

## 3.3 Modèle LABVIEW soft motion

Le logiciel labview contiens plusieurs model parmi eux le modèle soft motion qui es pont qui relie le logiciel labview avec d'autre logiciels pour avoir un mouvement au niveau des axes ; comme la synchronisation des deux logiciel labview et solidworks en utilisant le modèle précédent pour la mise en marche de la machine utile a commande numérique

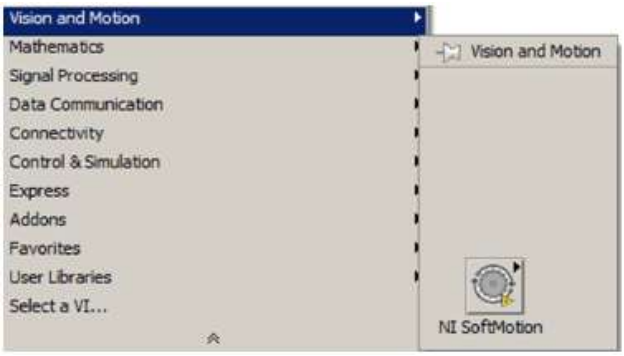



FIGURE 3.1 – instruments virtuels soft motion

## 3.4 Présentation des instruments virtuels

Les programmes LabVIEW portent la dénomination d'instruments virtuels ou VIs, car leur apparence et leur fonctionnement imitent ceux d'instruments réels tels que les oscil-

loscopes et les multimètres. Chaque VI utilise des fonctions qui manipulent les entrées de l'interface utilisateur ou d'autres sources et qui affichent ces informations ou les déplacent vers d'autres fichiers ou ordinateurs[8].

Un VI contient les trois composantes suivantes :

— Face-avant — Sert d'interface utilisateur.

— Diagramme — Contient le code source graphique qui définit les fonctionnalités du VI.

— Icône et connecteur — Identifie l'interface au VI pour que vous puissiez utiliser celui-ci dans un autre VI. Un VI à l'intérieur d'un autre VI est appelé sous-VI. Un sous-VI correspond à un sous-programme dans des langages de programmation textuels.

### 3.4.1 Face-avant

La face-avant est l'interface utilisateur du VI. La figure suivante montre un exemple de face-avant

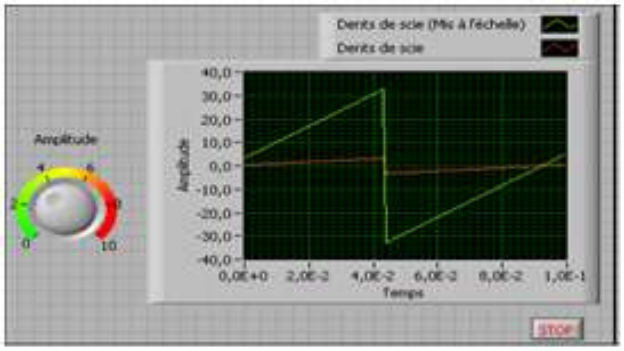



FIGURE 3.2 – exemple de face avant

Vous construisez la face-avant avec des commandes et des indicateurs qui sont respectivement les terminaux d'entrée et de sortie interactifs du VI.
Les commandes sont des boutons rotatifs, des boutons-poussoirs, des cadrans et autres mécanismes d'entrée. Les indicateurs sont des graphes, des LED et d'autres afficheurs de sortie. Les commandes simulent les mécanismes d'entrée des instruments et fournissent des données au diagramme du VI. Les indicateurs simulent les mécanismes de sortie d'instruments et affichent les données que le diagramme acquiert ou génère.

### 3.4.2 Diagramme

Après avoir construit la face-avant, vous devez ajouter le code en utilisant les représentations graphiques des fonctions pour commander les objets de la face-avant. Le diagramme contient ce code source graphique, aussi appelé code G ou code du diagramme. Les objets de la face-avant apparaissent en tant que terminaux sur le diagramme.
Le VI suivant montre plusieurs objets élémentaires du diagramme, tels que des terminaux, des fonctions et des fils de liaison.

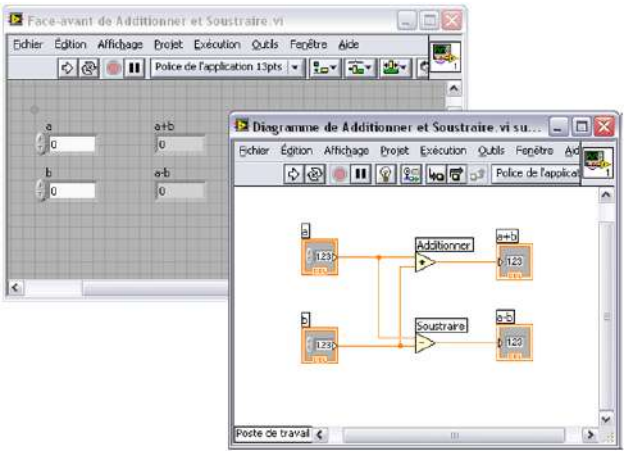



FIGURE 3.3 – exemple de diagramme

#### Terminaux

Les terminaux représentent le type de données de la commande ou de l'indicateur. Vous pouvez configurer des commandes ou des indicateurs de la face-avant pour qu'ils s'affichent sous forme d'icônes ou de terminaux de type de données sur le diagramme. Par défaut, les objets de la face-avant sont représentés par des terminaux icônes. Par exemple, un terminal icône de bouton rotatif, représenté de la façon suivante, représente un bouton rotatif sur la face-avant.




FIGURE 3.4 – bouton rotatif

La mention DBL au bas du terminal représente un type de données numérique à virgule flottante double précision. Un terminal DBL, illustré ci-dessous, représente une commande numérique à virgule flottante double précision.




FIGURE 3.5 – Un terminal DBL

### Noeuds

Les nœuds sont des objets du diagramme qui comportent des entrées et/ou des sorties et qui réalisent des opérations pendant l'exécution du VI. Ils sont analogues aux déclarations, opérateurs, fonctions et sous-programmes écrits dans des langages de programmation textuels. Les fonctions Additionner et Soustraire de la figure précédente correspondent à des nœuds.

### Fils de liaison

Le transfert des données entre les objets d'un diagramme s'effectue par les fils de liaison. Dans la figure précédente, les fils de liaison connectent les terminaux des commandes et des indicateurs aux fonctions Additionner et Soustraire. Chaque fil de liaison a une source de données unique, mais vous pouvez le câbler à de nombreux VIs et fonctions qui lisent les

données. Les fils de liaison sont de couleurs, de styles et d'épaisseurs différents en fonction de leur type de données. Un fil de liaison brisé apparaît sous la forme d'une ligne noire en pointillés avec un symbole X rouge au milieu. Les fils de liaison brisés apparaissent pour de multiples raisons, notamment lorsque vous essayez de câbler deux objets dont les types de données sont incompatibles.

### Structures

Les structures sont des représentations graphiques des boucles et des conditions des langages de programmation textuels. Utilisez des structures dans votre diagramme pour répéter des blocs de code et pour exécuter le code de manière conditionnelle ou dans un ordre spécifique.

### Icône et connecteur

Une fois que vous avez construit la face-avant et le diagramme d'un VI, créez son icône et son connecteur pour pouvoir l'utiliser en tant que Sous -VI. L'icône et le connecteur correspondent au prototype de fonction des langages de programmation textuels. Chaque VI affiche une icône, comme celle qui est représentée ci-dessous, dans le coin supérieur droit des fenêtres de la face-avant et du diagramme.

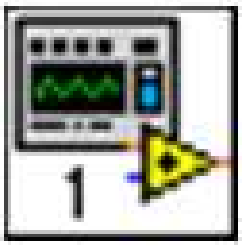



FIGURE 3.6 – icône dans le coin supérieur droit des fenêtres de la face-avant et du diagramme.

Une icône est la représentation graphique d'un VI. Elle peut contenir du texte, des images ou les deux. Si vous utilisez un VI comme sous-VI, l'icône identifie le sous-VI sur le diagramme du VI. Vous pouvez double-cliquer sur l'icône pour la personnaliser ou la modifier. Vous devez également construire un connecteur, représenté ci-dessous, pour utiliser le VI comme sous-VI.

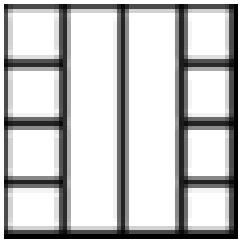


FIGURE 3.7 – Une icône pour représentation graphique

Le connecteur est un groupe de terminaux qui correspond aux commandes et aux indicateurs de ce VI ; ce groupe est semblable à la liste de paramètres d'un appel de fonction dans les langages de programmation textuels. Le connecteur définit les entrées et sorties que vous pouvez connecter au VI que vous voulez utiliser comme sous-VI. Un connecteur reçoit des données sur ses terminaux d'entrée et transmet ces données au diagramme par les commandes de sa face-avant. Il reçoit les résultats sur ses terminaux de sortie par les indicateurs de sa face-avant.

## 3.5 Boucles et structures

Les structures sont des représentations graphiques de boucles et de conditions dans les langages de programmation textuels. Utilisez des structures dans votre diagramme pour répéter des blocs de code et pour exécuter le code de manière conditionnelle ou dans un ordre spécifique[8].

Tout comme les autres nœuds, les structures possèdent des terminaux qui les connectent à d'autres nœuds du diagramme, elles s'exécutent automatiquement lorsque des données en entrée sont disponibles et fournissent des données aux fils de liaison de sortie lorsque l'exécution est terminée.

Chaque structure a une bordure spécifique et redimensionnable pour contenir la section du diagramme qui s'exécute selon les règles de la structure. La section du diagramme à l'intérieur du cadre de la structure est appelée sous-diagramme. Les terminaux qui fournissent et récupèrent les données dans les structures sont appelés des tunnels. Un tunnel est un point de connexion sur le cadre d'une structure.[8]

Utilisez les structures suivantes de la palette Structures pour contrôler l'exécution du diagramme :

— **Boucle For :** Exécute un sous-diagramme un certain nombre de fois.

— **Boucle While :** Exécute un sous-diagramme jusqu'à ce qu'une condition ait lieu.

— **Boucle cadencées :** Exécute un ou plusieurs sous-diagrammes avec des limites de temps et des retards.

— **Structure Condition :** Contient plusieurs sous-diagrammes, un seul d'entre eux s'exécutant selon la valeur d'entrée transmise à la structure.

— **Structure Séquence :** Contient un ou plusieurs sous-diagrammes, qui s'exécutent séquentiellement .

— **Structure Événement :** Contient un ou plusieurs sous-diagrammes s'exécutant selon l'interaction de l'utilisateur avec le VI.

Cliquez avec le bouton droit sur le cadre d'une structure pour afficher son menu local. Utilisez les boucles For et While pour contrôler des opérations répétitives.

### 3.5.1 Boucles

#### Boucles For

Une boucle For, illustrée ci-après, exécute un sous-diagramme un nombre de fois défini.




FIGURE 3.8 – Boucles For

### Boucles While

Comme les boucles Do et les boucles Repeat-Until des langages de programmation textuels, les boucles While, représentées ci-après, exécutent un sous-diagramme jusqu'à ce qu'une condition ait lieu.



FIGURE 3.9 – Boucles While

### Boucles cadencées

Une boucles cadencée synchronisé avec le moteur de balayage NI permet à vos applications de mouvement sensibles au temps d'exécuter à la vitesse de balayage. Tout code placé à l'intérieur d'une boucles cadencée est garantie pour exécuter une fois par période de balayage ou à un intervalle spécifié. Vous devez réduire les allocations de mémoire dans des boucles cadencées pour éviter d'introduire la gigue dans le système. NI SoftMotion blocs fonctionnels sont généralement utilisés sur le matériel exécutant un système d'exploitation en temps réel pour créer des applications de commande de mouvement déterministe en utilisant le paradigme de programmation du bloc fonction. Lorsque vous utilisez NI SoftMotion blocs fonctionnels sur Windows avec une simulation SolidWorks, la synchronisation du moteur de balayage NI est pas garantie.

Effectuez les étapes suivantes pour configurer la boucle cadencée :

— Cliquez droit sur Poste de travail et sélectionnez Nouveau ≫VI dans le menu contextuel pour ouvrir un VI vide.

— Placez une boucle cadencée sur le diagramme du VI. La boucle cadencée est situé sur la Timed Structures palette.

— Double-cliquez sur le nœud d'entrée de la boucle cadencée pour ouvrir la Configurer la boucle cadencée boîte de dialogue.

— Sous boucle Timing Source, pour Type de source, sélectionnez Synchroniser pour Scan Engine.

Vous pouvez cliquer sur leAide bouton pour obtenir des informations sur la synchronisation du moteur de balayage NI. La Configurer la boucle cadencée boîte de dialogue doit ressembler à la figure suivante.

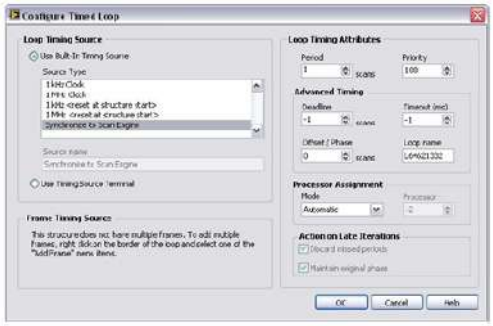



FIGURE 3.10 – Configuration de la boucle cadencée

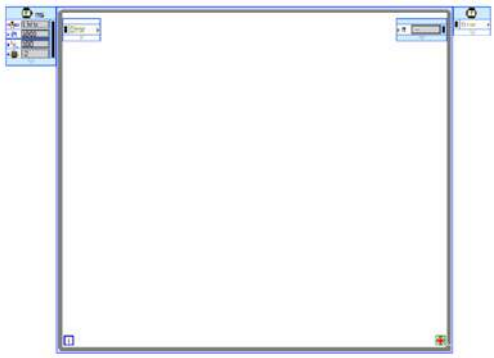


FIGURE 3.11 – Boucle cadencées

### Contrôle de la vitesse

Vous voudrez peut-être contrôler la vitesse d'exécution d'un processus, par exemple la vitesse à laquelle les données sont placées sur un graphe déroulant. Vous pouvez utiliser une fonction Attendre dans la boucle pour patienter pendant une période de temps exprimée en millisecondes avant que la boucle ne s'exécute à nouveau.

## 3.5.2 Structures Condition, Séquence et Événement

Les structures Condition, Séquence empilée, Séquence déroulée et les structures Événement contiennent de multiples sous-diagrammes. Une structure Condition exécute un sous-diagramme selon la valeur d'entrée transmise à la structure. Une structure Séquence empilée et une structure Séquence déroulée exécutent tous leurs sous-diagrammes dans l'ordre séquentiel. Une structure Événement exécute ses sous-diagrammes selon les interactions de l'utilisateur avec le VI.

### Structures Condition

Une structure Condition, représentée ci-après, comporte plusieurs sous-diagrammes, ou conditions.

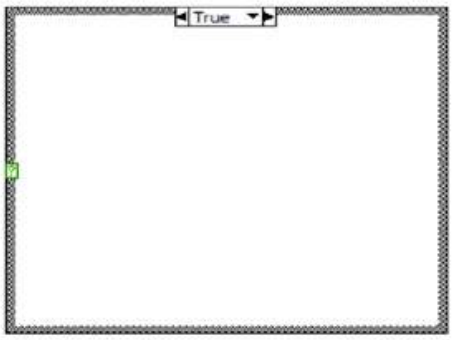



FIGURE 3.12 – Structures Condition

Un seul sous-diagramme est visible à la fois et la structure exécute seulement une condition à la fois. Une valeur d'entrée détermine quel sous-diagramme s'exécute. La structure

Condition est similaire aux déclarations switch ou aux déclarations if...then...else des langages de programmation textuels.

### Structures Séquence

Les structures Séquence contiennent un ou plusieurs sous-diagrammes, ou étapes, qui s'exécutent dans l'ordre séquentiel. Dans chaque étape d'une structure Séquence, comme dans le reste du diagramme, la dépendance des données détermine l'ordre d'exécution des noeuds. Les structures Séquence ne sont pas couramment utilisées dans LabVIEW. Il existe deux types de structures Séquence : la structure Séquence déroulée et la structure Séquence empilée.

La structure Séquence déroulée, illustrée ci-après, affiche toutes les étapes à la fois et les exécute de gauche à droite quand toutes les données câblées aux étapes sont disponibles, jusqu'à ce que la dernière étape ait été exécutée. Les valeurs de données quittent chaque étape lorsque celle-ci finit de s'exécuter.



FIGURE 3.13 – Structures Séquence

### Structures Événement

Une structure Événement, représentée ci-après, possède un ou plusieurs sous-diagramme ou conditions d'événement, dont un seul s'exécute lorsque la structure s'exécute.

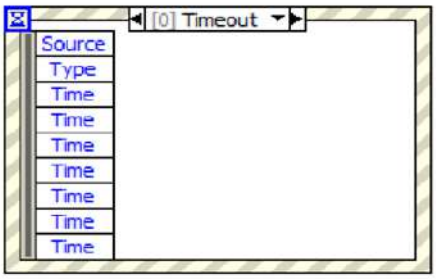



FIGURE 3.14 – Structures Événement

La structure Événement attend qu'un événement se produise, puis exécute la condition appropriée pour gérer cet événement. Les événements peuvent être générés dans l'interface utilisateur, par des E/S externes ou par d'autres parties de l'application. Les événements d'interface utilisateur comprennent les clics de souris, la saisie par les touches du clavier et ainsi de suite. Les événements d'E/S externes comprennent les compteurs ou les déclenchements matériels qui avertissent qu'une acquisition de données est achevée ou qu'une erreur s'est produite. D'autres types d'événement peuvent être générés par programmation et utilisés pour communiquer avec différentes parties de l'application. LabVIEW prend en charge les événements de l'interface utilisateur et ceux qui sont générés par programmation, mais pas les événements des E/S externes.

## 3.6 G-code

### 3.6.1 Définition

Le code G est un langage de programmation, ce langage est très simple est très entendu pour les machines outils à commande numérique et guider la machine pour sa fonction.[9]

### 3.6.2 Fonction

Tous les mots d'un langage machine autres que ceux définissant les cotes, et nécessaires pour assurer le fonctionnement d'une machine-outil.

— G – fonctions préparatoires.

— F – fonctions vitesse d'avance.

— S – fonctions vitesse de broche.

— T – fonctions outil.

— M – fonctions auxiliaire.

**Fonction préparatoires**

Fonction définies par l'adresse G et préparant la logique à un type De calcul ou à une action déterminée. Exemple : G00, G01, G02, G03, G04,...etc.

**Composition maximum d'un bloc**

exemple : N22 G02 X10 Y60 Z50 I10 J10 K10 F35 ou :

— N - Numéro de bloc.

— G – fonction préparatoire.

— X – Mouvement suivant l'axe X.

— Y – Mouvement suivant l'axe Y.

— Z – Mouvement suivant l'axe Z.

— I – coordonnée de l'axe du rayon en X.

— J – coordonnée de l'axe du rayon en Y.

— K – coordonnée de l'axe du rayon en Z.

— F – valeur de l'avance en millimètres par milliseconde.

### 3.6.3 Interpolation

**Interpolation linéaire**

$G00X_Y_Z$ : Déplacement linéaire depuis la position courante jusqu'au point de coordonnées $X, Y, Z$. seules les axes à déplacer est à indiquer.

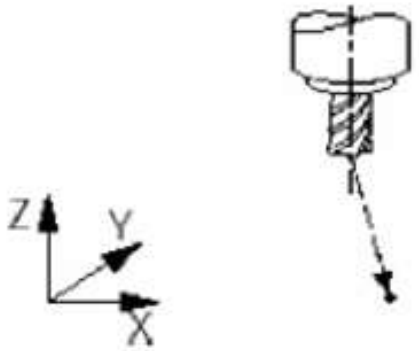



FIGURE 3.15 – interpolation linéaire en avance rapide.

Exemple : $G00X12Y3$ ; déplacement linéaire à ($X = 12 et Y = 3$).

**Interpolation linéaire en avance travail**

$G01X_Y_Z_F$ : Interpolation linéaire en avance travail (donnée par F) depuis la position courante jusqu'au Point de coordonnées X, Y, Z. seules les axes à déplacer sont à indiquer, si F n'est pas donné, La dernière avance travail utilisée sera reprise.
Exemple : G01 X12 Y3 F300 ; déplacement à (X=12, Y=3) à 300 mm/min.

**Interpolation circulaire en sens horaire**

$G02X_Y_I_J_F$ : Interpolation circulaire en X, Y dans le sens horaire en avance travail (donnée par F) depuis La position courante jusqu'au point de coordonnées X, Y en décrivant un arc centré en I, J. si F n'est pas donné, la dernière avance travail utilisée sera reprise.

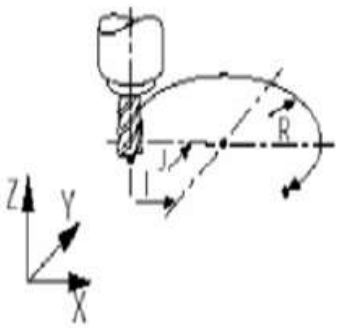



FIGURE 3.16 – interpolation circulaire en sens horaire

Exemple : G02 X10 Y10 I05 J05 F200.0 ; Arc à (10,10) avec le centre à l'origine.

**Remarque** : il existe d'autres fonctions G code.

## 3.7 Conclusion

Dans ce chapitre inclura une généralité sur le logiciel de programmation labview avec Présentation des instruments virtuel et le modèle soft motion et langage de programmation G code.

# Chapitre 4

# Interconnexions entre labview et solidworks

## 4.1 Introduction

Le logiciel de programmation Labview possède de nombreux modules. L'un de ces modules : ≪ Soft motion ≫, permet de communiquer avec solidworks, et ainsi piloter une machine a commande numerique . Il est donc possible de prototyper et simuler une application de contrôle d'axes sur une machine a commande numérique avant d'acheter les éléments physiques. La programmation graphique de haut niveau offre de nombreuses fonctionnalités intéressantes telles que :

— Contrôle et commande d'axes en vitesse, position, accélération.

— Suivi de trajectoires.

— Détection de collision.

— ...

Nous allons voir à travers deux commandes :

— Commande manuelle

— Commande automatique

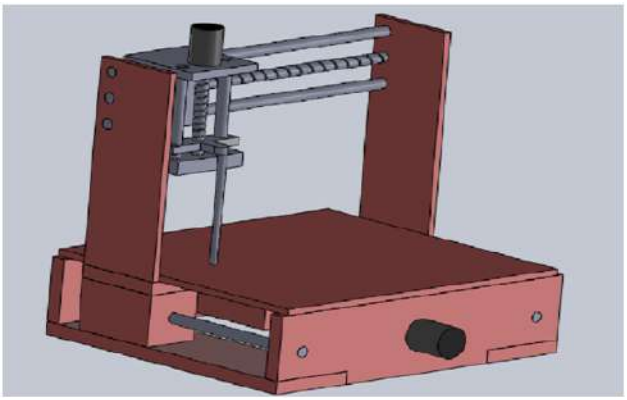


FIGURE 4.1 – machine outil à commande numérique

## 4.2 la commande manuelle

### 4.2.1 Création du projet et import des moteurs de solidworks motion

Lancez le logiciel Labview, créez un nouveau projet intitulé ≪ commande manuel≫. Lancez le logiciel solidworks, et ouvrez le fichier ≪assemblage .SLDASM ≫ Dans le projet Labview, cliquez bouton droit sur Poste de travail puis Nouveau / Solidworks Assembly. L'assemblage précédemment ouvert dans solidworks est détecté, et Labview se propose d'importer les moteurs de cet assemblage. Validez par OK.

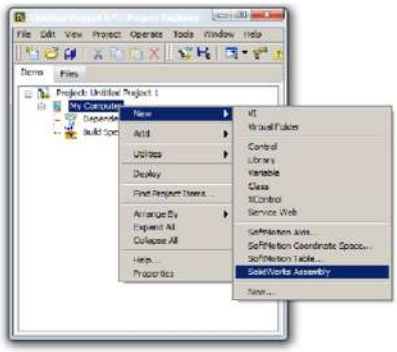



FIGURE 4.2 – fenêtre de projet

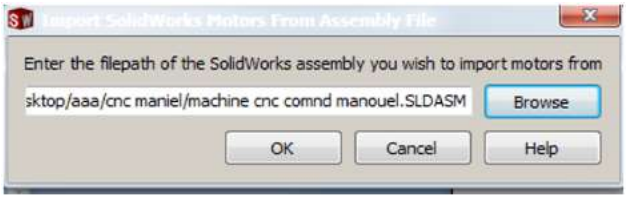



FIGURE 4.3 – Fenêtre d'ouvrir l'assemblage

Remarquez que l'assemblage solidworks apparait maintenant dans le projet avec les trois moteurs.

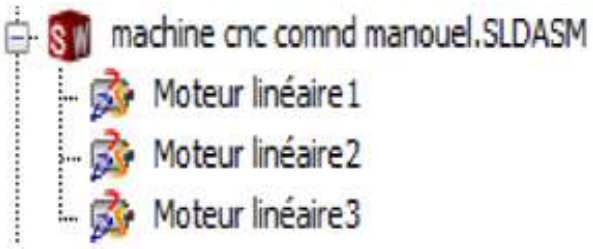



FIGURE 4.4 – Fenêtre d'ouvrir l'assemblage

### 4.2.2 Création d'un axe dans labview

Vous allez maintenant créer trois axes dans le projet Labview., et ainsi le lien sera réalisé entre Labview et Solidworks.

Pour cela, dans le projet Labview, cliquez bouton droit sur Poste de travail, puis Nouveau / NI SoftMotion Axis...

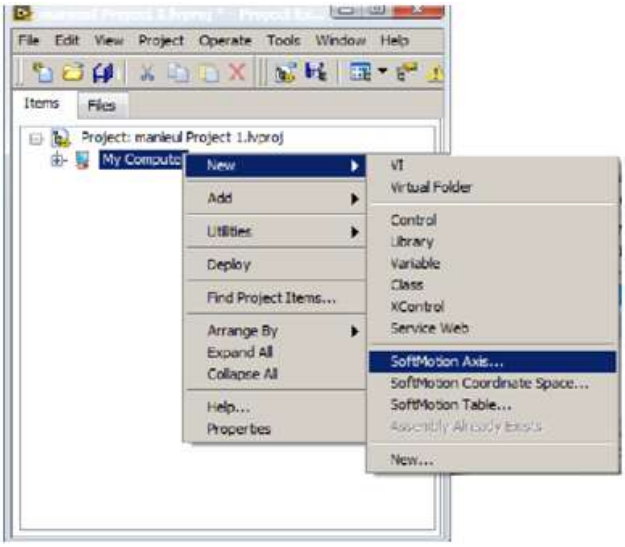



FIGURE 4.5 – crée les axes softmotion

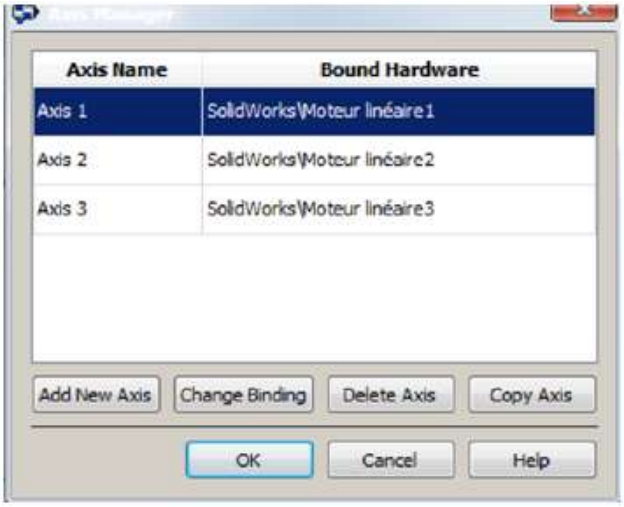



FIGURE 4.6 – fenêtre de sélectionner les trois axes

Une fenêtre ≪ Axis Manager ≫ apparait. Cliquez sur add New Axis pour créer trois axes. Trois axes intitulé ≪ Axis 1, axis 2, axis 3 ≫ Apparait dans le projet




FIGURE 4.7 – les trois axes

Afin de simplifier notre projet, vous allez dans les propriétés de chaque axe cocher la case ≪ Enable drive on transition to active mode ≫. Cela évite de rendre opérationnel (enable) l'axe par une fonction supplémentaire appelée Power dans le VI que vous allez créer par la suite. Cliquez bouton droit dans le projet Labview sur ≪ Axe 1,2,3 ≫ puis Propriétés, et cocher cette case ≪ Enable drive on transition to active mode ≫. Au passage, dans les propriétés de l'axe, vous voyez qu'il est possible de modifier énormément de paramètres.

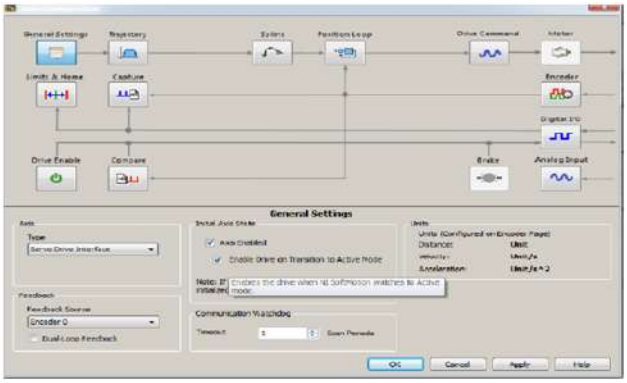



FIGURE 4.8 – cocher Enable drive on transition to active mode

### 4.2.3 Création du VI de commande dans Labview

#### Réalisation du diagramme (Programmation)

Vous allez maintenant créer l'application permettant de piloter les trois moteurs. Pour cela cliquez bouton droit sur Poste de travail puis Nouveau / VI. Une face avant et un

diagramme apparaissent.




FIGURE 4.9 – nouvelle VI

Enregistrez ce VI sous le nom ≪ VI commande 3 moteur.vi ≫.




FIGURE 4.10 – VI principale avec les sous VI

Réalisation de l'application de commande dans le VI. Vous allez dans un premier temps commander en vitesse et accélération les trois moteur du la machine. Insérez pour cela une commande ≪ Straight-Line Move ≫ dans le diagramme du VI, vous la trouverez dans la palette de fonctions à Vision et Motion / SoftMotion / Advanced / Function Blocks / Straight-Line Move.

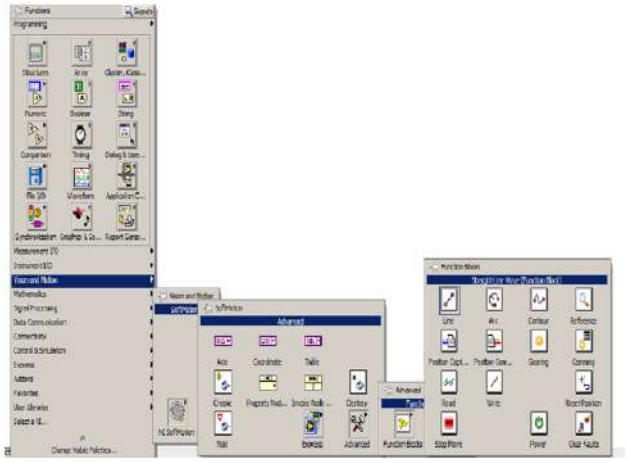


FIGURE 4.11 – sélectionner Straight-Line Move

Par défaut la fonction Straight- Line Move est configurée en Déplacement absolu≪ Absolute ≫, or nous souhaitons piloter uniquement en vitesse et accélération les trois moteurs du la machine. Cliquez bouton droit sur le bloc fonction Straight-Line Move puis Select Method / Velocity.

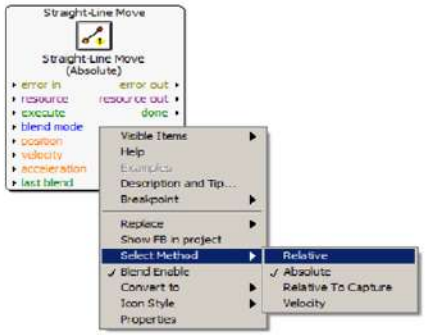



FIGURE 4.12 – comment changer la méthode

Ce bloc fonction Straight-Line Move comporte les entrées et sorties suivantes :

— error in / error out : entrée et sortie qui permettent de gérer les messages d'erreurs qui pourraient survenir durant l'exécution de l'application. Nous ne les utiliserons pas par la suite.

— ressource : Entrée sur laquelle sera connecté l'axe à commander, dans notre cas ≪ Axe moteur ≫.

— ressource out : Sortie permettant de connecter en série d'autres blocs fonction (Power, Stop Move,. . . ) agissant sur le même axe que celui connecté sur ressource.

— execute : Entrée booléenne qui lorsqu'elle est à 1 exécute le bloc Straight-Line Move, c'est-àdire accélère l'axe connecté sur ressource jusqu'à la vitesse désirée.

— velocity : Vitesse souhaitée en degrés/s de l'axe piloté connecté sur ressource.

— acceleration : Accélération en degrés/s$^2$ pour atteindre la vitesse souhaitée.

— done : Sortie booléenne qui est à 1 lorsque la vitesse souhaitée de l'axe est atteinte. Connectez une par une ces entrées et sorties, pour cela :

— Cliquez bouton droit sur l'entrée ressource (lorsque la bobine apparait), puis Créer / Commande.

— Cliquez bouton droit sur l'entrée execute (lorsque la bobine apparait), puis Créer / Commande.

De la même façon :

— Créez une commande pour la vitesse et l'accélération.

— Créez un indicateur sur la sortie done.

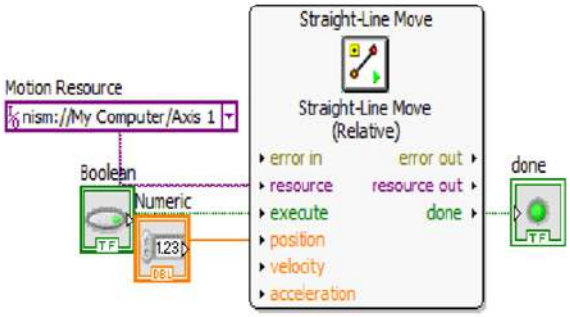



FIGURE 4.13 – présentation une Straight-Line Move avec les composants

De la même façon :

Créez une commande pour les deux moteurs 2 et 3.

Maintenant vous allez insérer l'ensemble de votre programmation (tout le contenu de diagramme) dans une boucle While. Pour cela sélectionnez dans la palette de fonction à Programmation / Structures la boucle While, et entourer le contenu du diagramme...

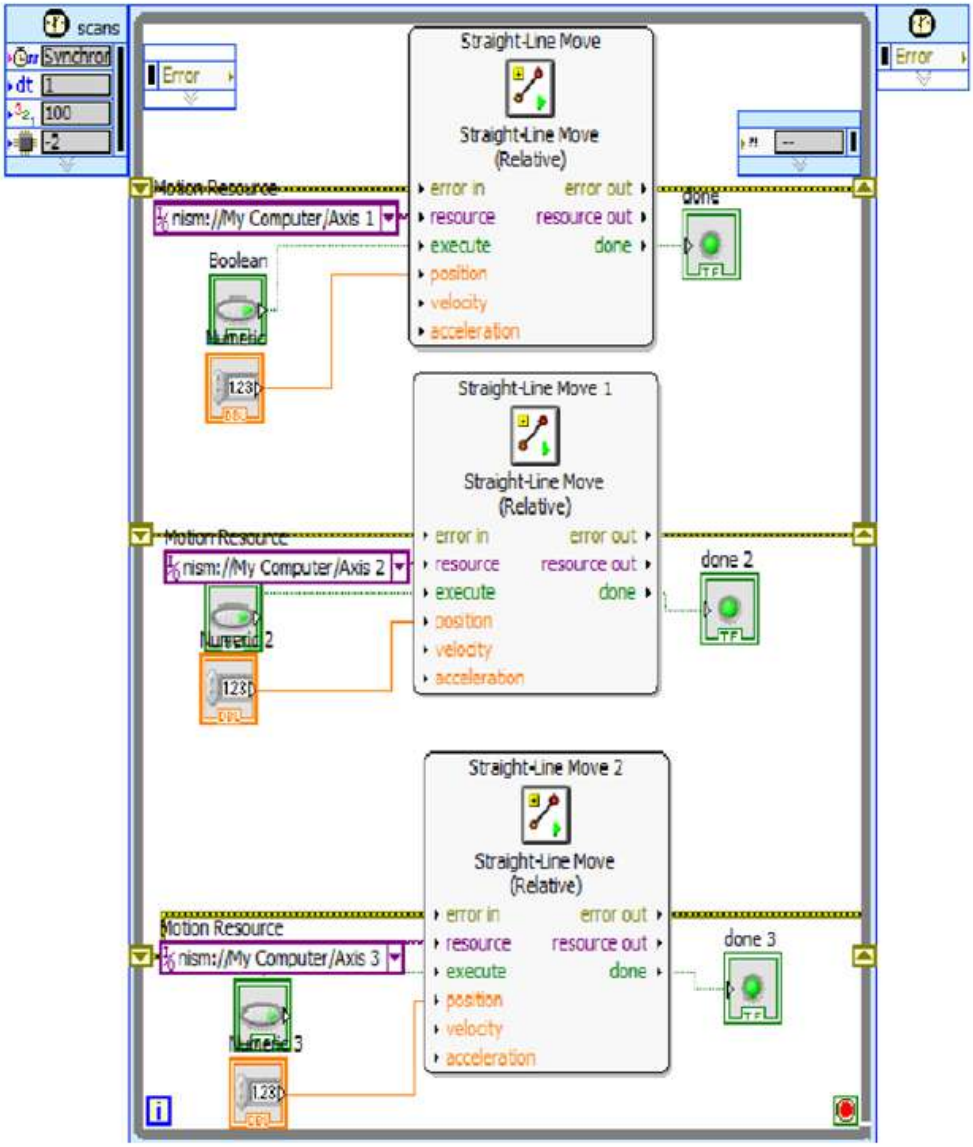



FIGURE 4.14 – Diagramme de la commande général

**Réalisation de la face avant (Interface utilisateur)**

Vous allez maintenant réorganiser la face avant afin qu'elle soit plus lisible. Afficher la fenêtre windows dans laquelle se trouve la face avant du VI, les différentes commandes et indicateurs insérés dans le diagramme apparaît mais sont mal positionnés.

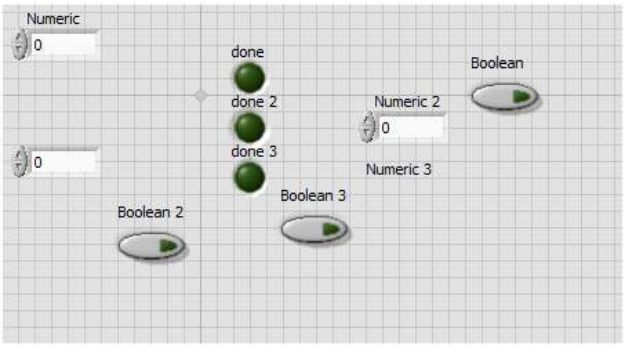



FIGURE 4.15 – Face avant de la commande générale mal positionnés

Réorganisez les afin d y voir plus clair et renommer les différentes étiquettes de la sorte :

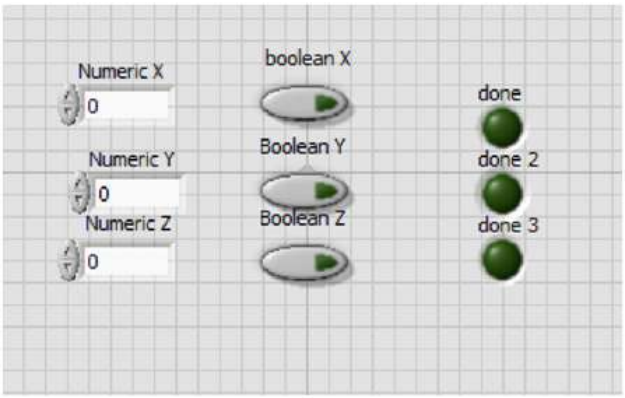



FIGURE 4.16 – Face avant de la commande générale

### 4.2.4 Exécution du VI et simulation virtuelle du mouvement sous SW motion

Vous devez déployer les axes Labview dans solidworks motion, cliquez pour cela bouton droit sur ≪ Axe moteur ≫ dans la fenêtre du projet Labview, puis Déployer.

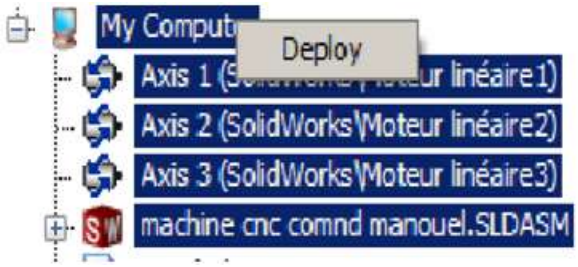



FIGURE 4.17 – déployer les axes avec poste de travaille

Une fenêtre apparait et propose de ≪ Passer en mode configuration puis en mode actif ≫ afin de déployer les Axes les moteurs. Cliquez sur appliquer.

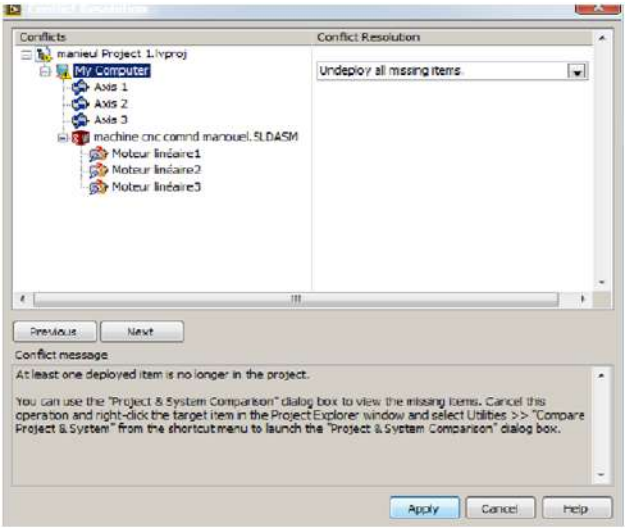



FIGURE 4.18 – apply pour complet l'interconnexion

Lancez la simulation (le moteur de calcul) dans solidworks motion en cliquant bouton droit sur ≪ assomblage machine .SLDASM ≫ dans la fenêtre de projet Labview, puis Start simulation. Si vous revenez dans la fenêtre Solidworks, vous vous apercevrez que le moteur de calcul à démarrer et que le temps s'écoule.

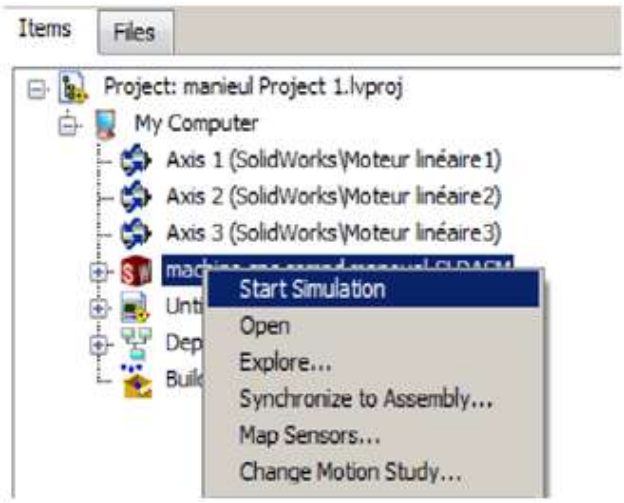



FIGURE 4.19 – Lancez la simulation

Affichez la face avant de votre VI puis exécutez-le en cliquant sur simulation. Essayez d'avoir à l'écran votre face avant et la machine solidworks en manipulant les fenêtres windows.

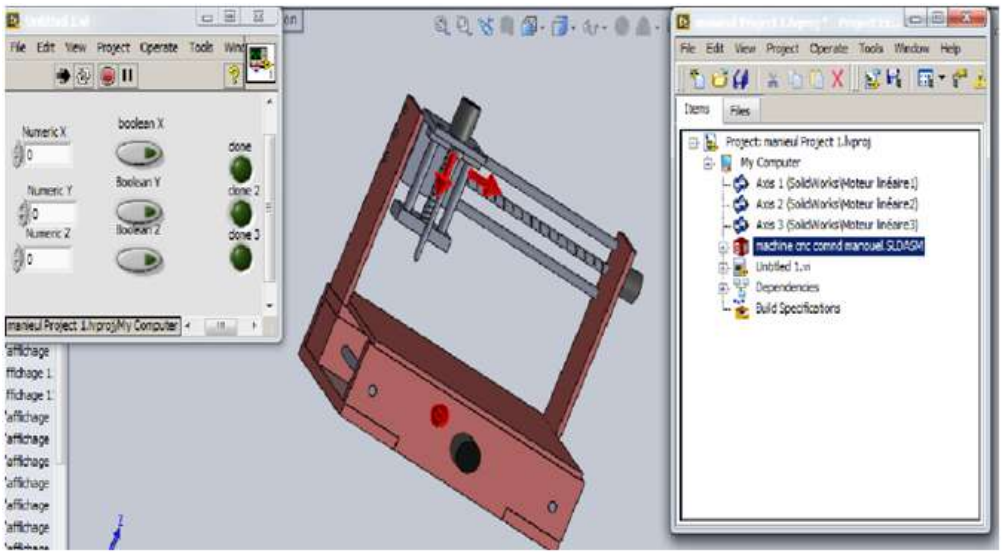


FIGURE 4.20 – simulation entre labview et solidworks

• Enfin appuyez sur le bouton ≪ Lancer le mouvement ≫ de votre face avant. Et commander les distances d'axes de la machine. Lorsque vous souhaitez interrompre le calcul de

l'animation, faites en cliquant bouton droit sur ≪ assemblage machine .SLDASM ≫ puis Stop Simulation.

Si vous arrêtez la simulation directement dans motion, il vous faudra quitter votre projet Labview et le ré-ouvrir. Vous pouvez ensuite révisionnel l'animation en cliquant sur lecture simulation dans Solidworks motion.

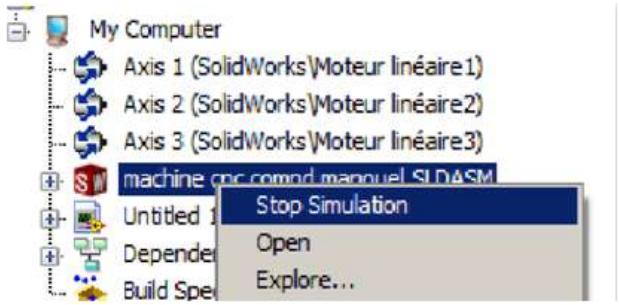



FIGURE 4.21 – stop simulation

Si entre deux simulations vous modifiez votre modèle solidworks (rapport de transmission dans une liaison, position d'un moteur, ...), il vous faut mettre à jour le projet Labview en cliquant bouton droit sur ≪ assemblage machine .SLDASM ≫ dans la fenêtre de projet puis Synchronize to Assembly

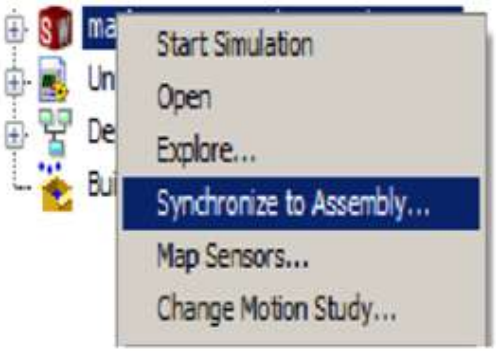



FIGURE 4.22 – rapport de transmission dans une liaison

## 4.3 Commande automatique

### 4.3.1 Création le projet

— Lancez le logiciel Labview, créez un nouveau projet intitulé ≪ commande de perceuse ≫.

— Lancez le logiciel solidworks, et ouvrez le fichier ≪ commande automatique de perceuse .SLDASM ≫

— Dans le projet Labview, cliquez bouton droit sur Poste de travail puis Nouveau / Solidworks Assembly. L'assemblage précédemment ouvert dans solidworks est détecté, et Labview se propose d'importer les moteurs de cet assemblage.

— Validez par OK

### 4.3.2 création les axes

Dans le projet Labview, cliquez bouton droit sur Poste de travail, puis Nouveau / NI softMotion Axis... Une fenêtre ≪ Axis Manager ≫ apparait. Cliquez sur add New Axis pour créer trois axes.

### 4.3.3 Commande (perceuse)

Initialement on vas travaille sur un programme d'une perceuse, alors on va procède par la réalisation du diagramme suivi par la réalisation de la face avant ie : face d'utilisateur finalement en va exécuté le programme .

**Réalisation du diagramme (Programmation)**

En procède comme sur la commande manuelle mais au lieu de faire trois Straight-Line Move en associe sur chaque position son propre Straight-Line Move ; par contre on projette chaque ancien error out comme la nouvelle error in du nouveau Straight-Line Move et en fixe la distance de chaque position

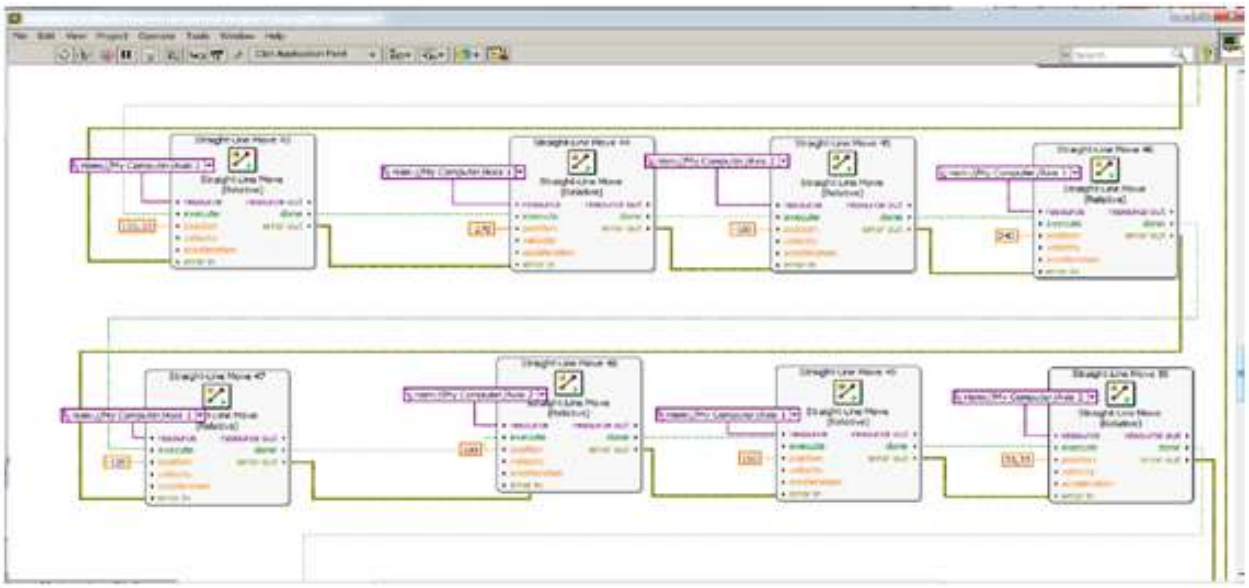


FIGURE 4.23 – Réalisation du diagramme (perceuse)

### Réalisation de la face avant (Interface utilisateur)

Après avoir terminé la réalisation du diagramme en vas passer par suite par la réalisation de la face avant, c'est la qu'on aperçoit un mal positionnement des icônes alors on va les repositionné comme sur la figure ci dessous

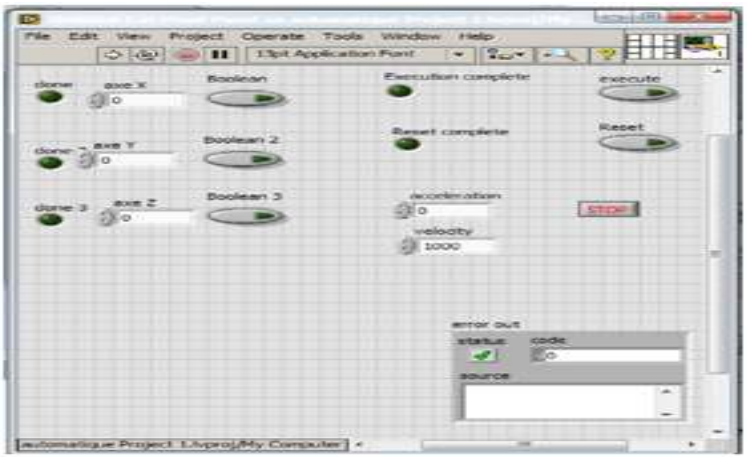



FIGURE 4.24 – Realisation de la face avant (perceuse)

**Exécution du VI et simulation virtuelle du mouvement sous SW motion**

Vous devez déployer les axes Labview dans solidworks motion, cliquez pour cela bouton droit sur ≪ Axes moteurs ≫ dans la fenêtre du projet Labview, puis Déployer.

Lancez la simulation (le moteur de calcul) dans solidworks motion en cliquant bouton droit sur ≪ assemblage perceuse machine .SLDASM ≫ dans la fenêtre de projet Labview, puis Start simulation. Si vous revenez dans la fenêtre Solidworks, vous vous apercevrez que le moteur de calcul à démarrer et que le temps s'écoule.

Affichez la face avant de votre VI puis exécutez-le en cliquant sur simulation. Essayez d'avoir à l'écran votre face avant et la machine solidworks en manipulant les fenêtres windows.

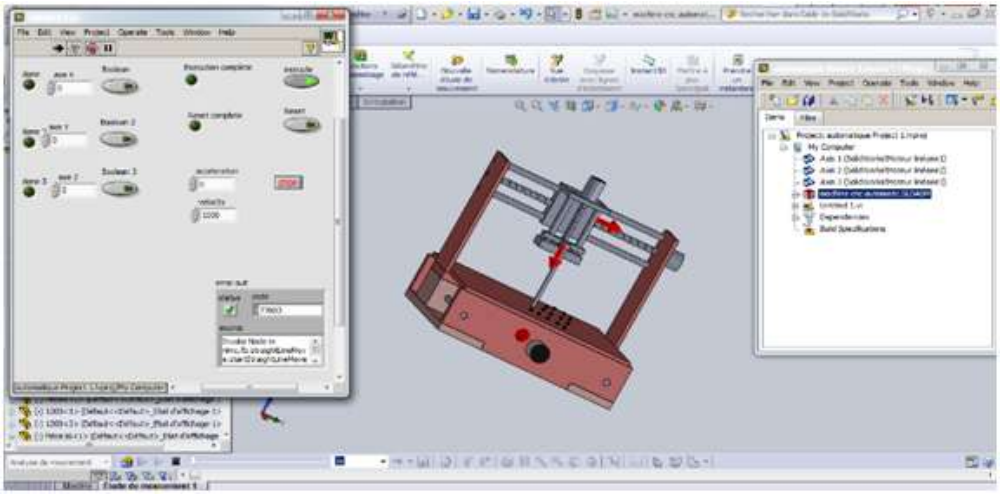


FIGURE 4.25 – Exécution du VI et simulation (perceuse)

### 4.3.4 Commande (automatique sans G code)

Apre avoir travaillé sur la commande de perceuse en va la commande pour écrire un mot(AISSA)

**Réalisation du diagramme (Programmation)**

Le diagramme a réalisé sera comme celui de la perceuse avec une modification des constantes de position et des axes selon la distance et direction recherchées

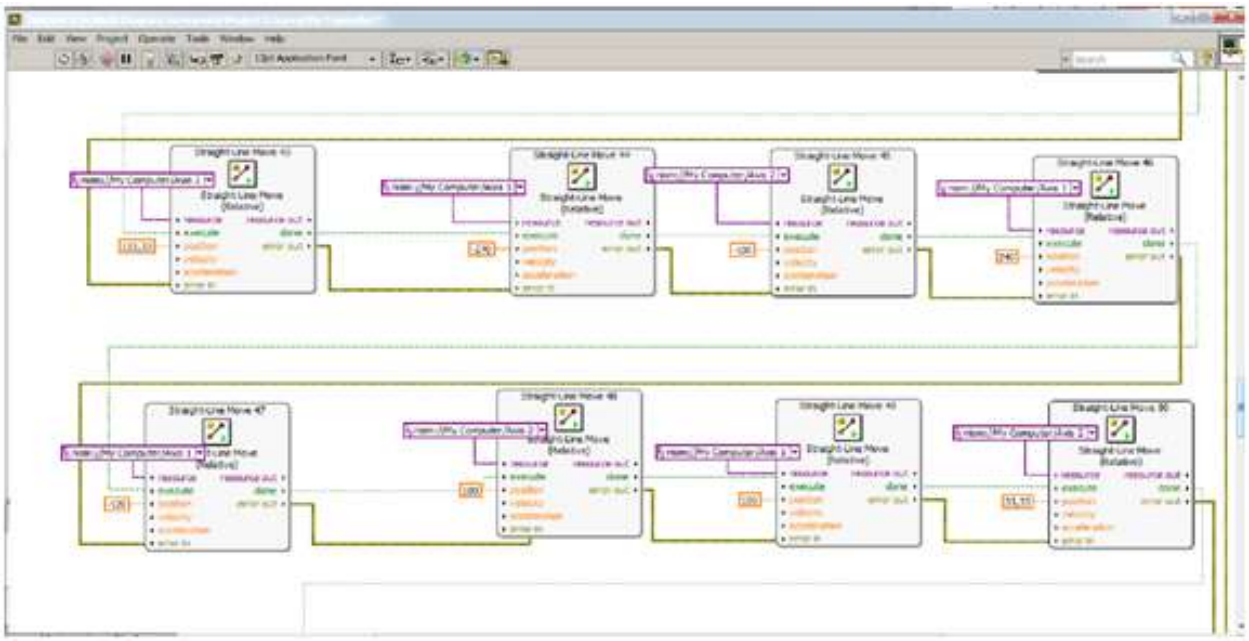


FIGURE 4.26 – Réalisation du diagramme (automatique sans G code)

### Réalisation de la face avant (Interface utilisateur)

Après avoir terminé la réalisation du diagramme en vas passer par suite par la réalisation de la face avant, c'est la qu'on aperçoit un mal positionnement des icônes alors on va les repositionné comme sur la figure ci dessous




FIGURE 4.27 – Réalisation de la face avant (automatique sans G code)

#### Exécution du VI et simulation virtuelle du mouvement sous SW motion

Finalement on va visualiser le résultat de la simulation comme présenté dans la figure si dessous

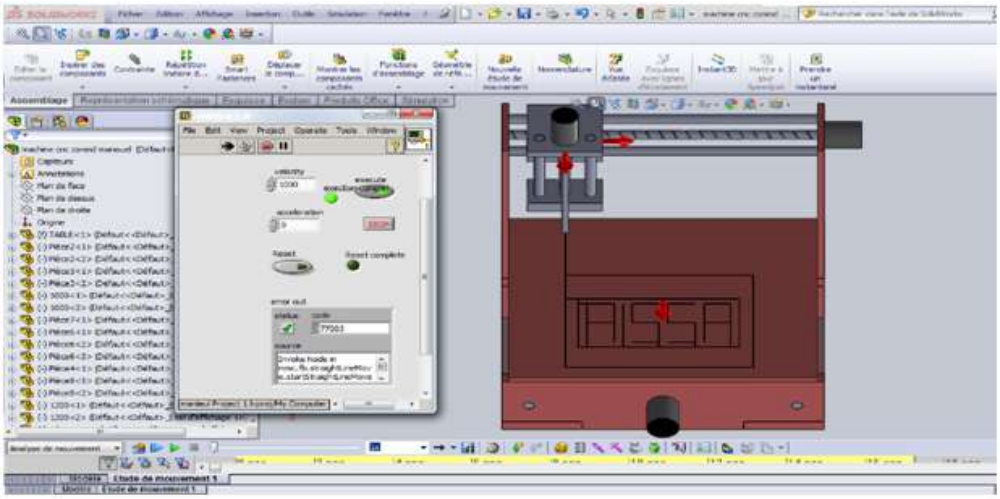


FIGURE 4.28 – Exécution du VI et simulation

### 4.3.5 Commande (automatique avec G code)

#### Réalisation du diagramme (Programmation)

Dans cette étape on va essayer de lancé la simulation de deux ou plusieurs axes ce qui nous pousse a intégré la fonction pickline dans le boucle for et la mettre dans un sous système en créant une constante appelé G-code, et le joindre avec un graphe XY pour visualisé notre schéma par suite, ce qui est représenté dans la figure si dessous

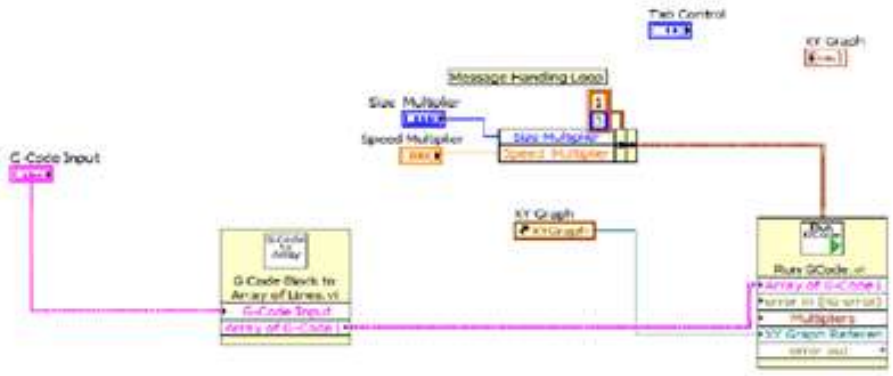



FIGURE 4.29 – Réalisation du diagramme (Programmation) avec G code

### Réalisation de la face avant (Interface utilisateur)

Après la réalisation du diagramme en passe a la face avant ou en va trouve la fenêtre de graphe XY et la fenêtre du G code ou en va le mettre ; le résultat de simulation et représenté dans la figure si dessous

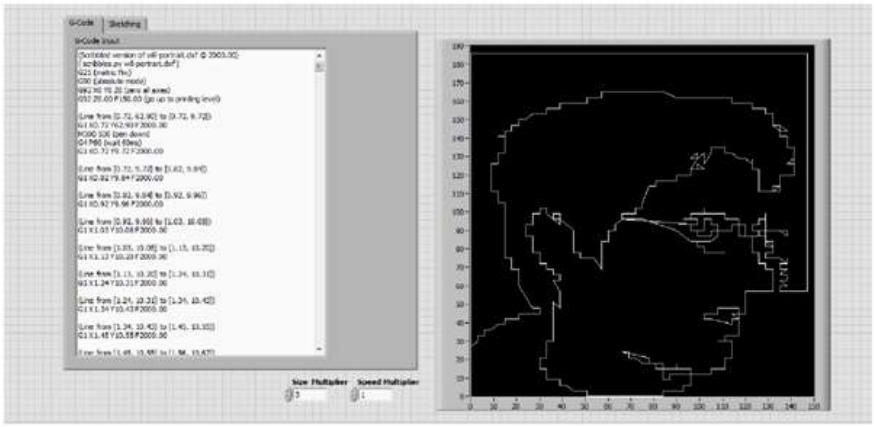


FIGURE 4.30 – Réalisation de la face avant (automatique avec G code)

## 4.4 Conclusion

Dans ce chapitre on a présenté comment commander la machine outil à commande numérique avec un logiciel de programmation graphique LABVIEW et l'interconnexion entre LABVIEW et logiciel SOLIDWORKS avec deux commande, la premier c'est une commande manuelle et la deuxième c'est une commande automatique.

# Conclusion Générale et Perspectives

Le domaine des machines CNC est très riche et demandé dans le marché dont la machine outil a commande numérique à 3 axes. Cela est dû à la grande importance des MOCN car ces dernières permettent de faire des multitudes d'opérations sur les matériaux come le sculptage, fraisage, moulage ... etc. L'objectif de ce travail est l'étude et pilotage de la machines à outil à commande numérique (MOCN). Pour cela nous avons concepté la machine outil a commande numérique avec le logiciel solidworks et la simulée en utilisant le même logiciel, ensuite on a procède sa programmation en utilisant le logiciel graphique LABVIEW sous deux commande comme suite :

— La première : commande manuelle.

— La deuxième : commande automatique.

— La dernière : l'interconnexion entre les deux logiciels

Pendant la procédure de ce travaille en a fait fasse a plusieurs difficultés permis eux :

— Le plus grand problème ces l interconnexion entre les deux logiciels labview et solidworks au moment d introductions du programme G-code car le solidworks n'as pas pu exécuter les axes

— Le deuxième obstacle c'est que le lancement des deux logiciel ensemble vas cause un blocage du pc, ce qui nous pousse a utilise un pc avec une très haut configuration

— Pour écrire notre commande en utilisant l option ≪ trace une trajectoire on a besoin des deux axes XY qui ne se trouve pas en solidworks (solidworks comprend trois axes XYZ) alors le passage d une lettre a une autre vas laisser une ligne de passage qu'on n'a pas pu la supprimé,

# Bibliographie


[1] *BENBEKHTI Ahmed, Etude de réalisation d'un support pour affutage des forets sur une machine à commande numérique, en octobre 2013( ESI OUED SMAR)*

[2] *David Maître, M. Etienne, Introduction à la commande numérique, 17 mars 2000 (UNIV PARIS SUDE)*

[3] *AFNOR - Cônes d'emmanchement, - norme AFNOR NF E 60-023 - Mai 1985 (UNIV PARIS SUDE)*

[4] *Philippe DEPEYRE, Fabrication mécanique, Année 2004-2005 (Université de la Réunion)*

[5] *Prise en main de SolidWorks PDF (Académie d'Orléans-Tours)*

[6] *solidworks.fr Profil de la société Purdue 1997 (Université Purdue Research and Education Centre for Information Systems)*

[7] *tutorial solidworks 2007*

[8] *National Instruments, NI, ni.com*

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