

**République Algérienne Démocratique et Populaire**  
**Ministère de l'Enseignement Supérieure et de la Recherche Scientifique**  
**Université Abderrahmane Mira de Béjaia**



**Faculté de Technologie**  
**Département d'Automatique, Télécommunication et d'Electronique**

## **Projet de Fin d'Etudes**

Pour l'obtention du diplôme de Master

Filière : Automatique

Spécialité : Automatique et informatique industrielle

### **Thème**

*Planification de trajectoires et commande d'un bras*

*Palettiseur (CEVITAL)*

#### **Préparé par :**

***BENGHERBI THINHINANE***

***BOUDIAB LYDIA***

#### **Dirigé par :**

***Mr. MENDIL Boubekour***

***Mr. MAOUCHE Lyazid***

#### **Membres de jury :**

***Mr.Nait mohend Nassim***

***Mr.Lehouche hocine***

**Année universitaire : 2024/2025**

## ***Remerciement***

***On remercie Dieu le tout puissant de nous avoir donné la santé et la volonté  
d'entamer et de terminer ce mémoire.***

***Tous d'abord, ce travail ne serait pas aussi riche et n'aurait pas pu avoir le jour  
sans l'aide et l'encadrement de Mr B. MENDIL BOUBEKEUR, nous lui  
exprimons notre profonde gratitude pour la qualité de son encadrement  
exceptionnel, pour sa patience, sa rigueur et sa disponibilité durant notre  
préparation de ce mémoire...***

***Nos remerciements les plus sincères vont également à Mr MAOUCHE LYAZID  
ingénieur à CEVITAL pour son soutien moral, son aide précieuse et ses  
encouragements.***

***Nous tenons à exprimer notre reconnaissance aux membres de jury qui ont  
accepté d'évaluer ce travail, pour l'intérêt qu'ils y ont porté et pour leurs  
remarques constructives.***

***Enfin nous adressons nos remerciements à toute personne qui a participé de près  
ou de loin à l'accomplissement de ce mémoire. Nous souhaitons également  
adresser nos remerciements à l'ensemble des enseignants qui ont participé à nos  
formations et ont enrichi notre parcours académique.***

## *Dédicace*

*Tout d'abord, je remercie Dieu, notre créateur de m'avoir donné la force, la volonté et le courage nécessaire à l'accomplissement de ce travail.*

*J'ai le grand plaisir de dédier ce modeste travail :*

*A mes très chers parents qui m'ont guidée, encouragée et soutenue toute au long de ma vie. Ma mère qui a sacrifié tant de choses afin de me voir devenir la personne que je suis aujourd'hui, et mon père qui m'a tout donné pour réaliser mes rêves et qui me pousse toujours à aller plus loin.*

*A mes frères et sœurs qui ont toujours été à mes côtés.*

*A la mémoire de mon cher grand-père.*

*A tous les membres de la famille.*

*Et a tous mes amis, en particulier : ma chère Elissa pour sa présence constante malgré la distance.*

*Enfin, je remercie ma binôme, Lydia, qui a contribué à la réalisation de ce travail.*

***BENGHERBI Thinhinane***

## *Dédicace*

*Je dédie ce travail, à mes chers parents pour leur amour et leurs sacrifices, à mes frères et sœurs pour leur soutien constant, à ma binôme, pour son accompagnement précieux durant toutes les étapes de ce projet. Ainsi qu'à toutes les personnes qui m'ont accompagnée, soutenue ou conseillée tout au long de mon parcours universitaire.*

*BOUDIAB Lydia*

## *Liste des abréviations*

AC : Courant Alternatif

ATAN2 : En trigonométrie, atan2 à deux arguments est une variation d'ARC tangente.

API : Automate Programmable Industriel

CPU : Central Processing Unit

DB : Bloc de données

DC : Courant Continu

DDL : Degrés de Liberté

DH : Denavit et Hartenberg

MGD : Modèle géométrique direct.

MGI : Modèle cinématique inverse.

FBD : Function Block Diagram

I/O : Input / Output

IHM : Interface Homme-Machine

IL : Instruction List

LD : Ladder Diagram

OB : Bloc d'organisation

P : Prismatique

PB : Bloc fonctionnel

PC : Partie Commande

PD : Proportionnel – Dérivé

PID : Proportionnel – Intégral – Dérivé

PLC : Programmable Logic Controller

PO : Partie Opérative

Profibus : Process Field Bus

PROFINET : Process Field Network

R : Rotoïde

SFC : Sequential Function Chart

ST : Structured Text

TTL : Transistor-Transistor Logic

# *Sommaire*

<b>Introduction générale .....</b>	<b>1</b>
------------------------------------	----------

## **Chapitre I : Présentation de l'unité de conditionnement d'huile**

I.1 Introduction.....	3
I.2 Offre de produits agroalimentaires de CEVITAL.....	3
I.3 Présentation de l'entreprise CEVITAL .....	4
I.3.1 Organigramme de l'entreprise .....	4
I.3.2 Activités et objectifs de l'entreprise.....	5
I.4 Unité de conditionnement d'huile.....	5
I.4.1 Organisation du travail à l'unité de conditionnement d'huile.....	5
I.4.2 Lignes de conditionnement d'huile.....	5
I.4.3 Etude des lignes de production d'huile.....	6
I.4. Présentation du travail.....	9
I.5 Conclusion.....	9

## **Chapitre II : Généralités sur les bras de robots et systèmes automatisés**

II.1 Introduction.....	10
II.2 Généralités sur la Robotique.....	10
II.2.1 Définition.....	10
II.2.2 Les disciplines impliquées dans la Robotique.....	10
II.2.3 Types de Robot.....	11
II.2.4 Domaines d'application.....	12
II.3 Architecture d'un bras Manipulateur.....	13
II.3.1 La base.....	14
II.3.2 Articulation.....	14
II.3.3 Segment.....	15
II.3.4 Actionneur.....	15
II.3.5 Organe terminal.....	15
II.3.6 Degrés de Liberté d'un Bras Manipulateur (ddl).....	16

II.4 Morphologies des Bras Manipulateurs.....	16
II.4.1 Structure cartésienne.....	16
II.4.2 Structure cylindrique.....	16
II.4.3 Structure sphérique (polaire).....	16
II.4.4 Structure anthropomorphe.....	17
II.4.5 Structure SCARA (Selective Compliance Assembly Robot Arm).....	17
II.5 Variateurs de vitesse.....	17
II.5.1 Constituants des variateurs de vitesse.....	18
II.5.2 Variateurs de vitesse dans la robotique.....	18
II.6 Systèmes automatisés.....	18
II.6.1 Systèmes automatisés et la Robotique.....	18
II.6.2 Structure d'un système automatisé.....	19
II.7 Généralité sur les Automates Programmable Industrielle (API).....	19
II.7.1 Définition.....	19
II.7.2 Structure interne d'un Automate Programmable Industrielle.....	20
II.7.3 Langages de programmation .....	21
II.7.4 L'exécution des programmes.....	21
II.7.5 Critère et choix des API.....	22
II.7.6 Présentation de Siemens S7-1200.....	22
II.8 Conclusion.....	23

### **Chapitre III : Modélisation, commande et planification de trajectoires d'un bras robotisé**

III.1 Introduction.....	24
III.2 Représentation des coordonnées d'un bras de robot .....	24
III.2.1 Représentation d'un point.....	25
III.2.2 Orientation et matrices de rotation.....	26
III.3 Transformations homogènes.....	26
III.3.1 Matrices de rotation.....	27
III.3.2 Matrice de translation.....	27
III.4 Modélisation géométrique.....	27

III.4.1 Procédure de calcul du MGD par la convention DH.....	28
III.4.2 Modèle géométrique du robot PUMA 560.....	28
III.4.3 Modèle géométrique inverse.....	30
III.5 Modélisation Cinématique.....	31
III.6 Modélisation Dynamique.....	32
III.7 Commande d'un bras robotisé.....	33
III.7.1 Méthodes de commande .....	33
III.7.1.1 Régulation linéaire.....	33
III.7.1.2 Commande en poursuite de trajectoire.....	35
III.8 Planification de trajectoires d'un bras robotisé.....	35
III.8.1 Méthodes de planification dans l'espace articulaire.....	36
III.8.1.1 Méthode du polynôme Cubique.....	36
III.8.1.2 Méthode de polynôme de degré cinq.....	38
III.8.2 Planification de trajectoire pour le bras robotique PUMA 560.....	39
III.9 Simulation.....	40
III.10 Conclusion.....	45

## **Chapitre IV : Implémentation de la commande sous TIA PORTAL**

IV.1 Introduction.....	46
IV.2 Cahier des charges.....	46
IV.3 Architecture générale du système robotisé.....	47
IV.4 Boucle de régulation .....	48
IV.5 Matériels utilisés pour la conception du projet sous TIA Portal.....	49
IV.5.1 Automate programmable industriel (API).....	49
IV.5.2 Variateur de vitesse.....	49
IV.5.3 Servomoteur.....	50



IV.6 Conception du projet.....	50
IV.6.1 Création de projet.....	50
IV.6.2 Configuration du matériel.....	51
IV.6.3 Configuration du réseau de communication PROFINET.....	52
IV.6.4 Configuration de télégramme de communication.....	53
IV.6.5 Définition et Organisation des Variables.....	55
V.6.6 Conception des blocs principaux et traitement des conditions.....	57
IV.6.7 Mouvement des servomoteurs.....	61
IV.6.8 Visualisation des résultats de simulation du processus.....	65
IV.7 Supervision du bras palettiseur à l'aide de l'IHM.....	72
IV.7.1 Choix et configuration de l'IHM.....	72
IV.7.2 Présentation de la vue IHM.....	73
IV.7.3 Résultats de simulation de l'Interface Homme-Machine (IHM).....	74
IV.8 Conclusion.....	75
<b>Conclusion générale.....</b>	<b>76</b>
<b>Liste des bibliographies</b>	

## **Annexes**

## **Résumé**

### Introduction générale

L'automatisation industrielle, en tant qu'acteur majeur des avancées technologiques contemporaines, a radicalement transformé les procédés de production au cours des dernières décennies. L'introduction des robots industriels a permis de réaliser des tâches complexes, longtemps considérées comme non automatisables, entraînant une amélioration significative de la productivité, une réduction des coûts de fabrication et un renforcement de la qualité des produits. Ces systèmes robotiques, capables de fonctionner sans interruption et sans fatigue, sont devenus des éléments incontournables dans de nombreux secteurs industriels. Cette révolution a été amorcée à la fin des années 1970 avec l'apparition des robots industriels de première génération [1].

Cependant, bien que leur adoption soit désormais généralisée et leur développement rapide, la mise en œuvre de solutions robotiques continue de poser des défis techniques importants, en particulier en matière de planification des trajectoires et de commande précise des robots. En effet, la première problématique en robotique consiste à déplacer un robot dans un environnement connu ou inconnu tout en évitant les obstacles fixes ou mobiles, ce qui implique la définition d'une stratégie de mouvement, connue sous le nom de planification [2].

Dans ce cadre, ce travail se concentre sur l'automatisation de la chaîne de production d'huile chez CEVITAL, et plus précisément sur la commande d'un bras robotisé à 3 degrés de liberté (DDL) et une pince pour gérer la palettisation des bouteilles d'huile. Ce bras est contrôlé via un automate Siemens S7-1200 et le logiciel TIA Portal. La logique de commande a été simulée sur PLCSim pour en valider les performances. Le bras robotisé est piloté par des variateurs de vitesse qui commandent les servomoteurs, permettant ainsi des mouvements fluides et précis. Ces variateurs communiquent via PROFINET, garantissant une synchronisation optimale et un contrôle précis des déplacements. Une attention particulière est portée à la gestion des télégrammes pour assurer un échange de données fiable et rapide entre l'automate et les variateurs, ce qui permet une intégration fluide du bras robotisé dans la chaîne de production.

## Introduction générale

---

Pour la présentation de notre travail, notre mémoire a été structuré en quatre chapitres :

- Le premier chapitre présente, tout d'abord l'entreprise qui nous a accueillies, puis s'intéresse à l'unité de conditionnement d'huile ainsi qu'au processus de production associé.
- Le deuxième chapitre présente les notions fondamentales de la robotique industrielle, à travers les définitions, les types et les domaines d'application des robots, ainsi que leur architecture et leur morphologie. Il traite également des systèmes automatisés et des généralités sur les automates programmables.
- Le troisième chapitre est dédié à la modélisation géométrique, cinématique et dynamique des bras robotiques. Il présente également une méthode de planification de trajectoire dans l'espace articulaire, appliquée au robot 3 DDL.
- Le quatrième chapitre constitue la partie pratique du mémoire, consacrée à l'automatisation du bras robotisé (palettiseur), contrôlé par un automate Siemens S7-1200.

Nous débuterons par la présentation du cahier des charges et des différents composants utilisés dans le système. Par la suite, nous expliquerons le logiciel utilisé pour la gestion des commandes. Enfin, nous aborderons les logiques de programmation et les programmes Ladder appliqués, avant de présenter les résultats obtenus lors de la réalisation et de la simulation.

# **Chapitre I : Présentation de l'unité de conditionnement d'huile**

## **Chapitre I : Présentation de l'unité de conditionnement d'huile**

### **I.1 Introduction**

Ce premier chapitre présente d'abord l'entreprise CEVITAL au sein de laquelle on a accompli notre projet Master dans le cadre d'un stage. On donne ensuite une description de l'unité de conditionnement d'huile. On termine par la description de notre travail.

### **I.2 Offre de produits agroalimentaires de CEVITAL**

Le complexe CEVITAL propose une vaste gamme de produits, détaillée par le tableau I.1.

**Tableau I.1 : La gamme des produits CEVITAL agro-alimentaire [3]**

Produit	Marque	Format
Huile de table	Fleurial	1L, 1.8L, 4L
	Fridor pour professionnel	10L
	Fridor DUO	1L, 2L, 5L
Sucre	Sucre blanc Skor	Sachet 1kg, 2kg, 5kg, 10kg 1kg verseur 750gr morceau
	Sucre roux Skor	1kg sachet, 700gr sachet zippé
	Sucre glace Skor	700gr sachet zippé
Margarines	Fleurial	Plaquette 250gr Barquette 500gr et 250 gr
	La parisienne	Plaquette 500gr
	Tendre gourmand	Plaquette 250gr
	Matina	Barquette 250gr
Boissons	Lalla khedidja eau minéral	Pet : 1,5L et 0,5L
	Lalla khedidja eau pétillante	Pet : 1L et 0,33cl
	Tchina plusieurs saveurs	Pet 2L, 1L, 33cl Verre 25cl
	Tchina PEP'S (3 saveurs)	Canette 24cl
Smen	Smen MÉDINA	Barquette 1,8kg, 900g, 500g
Mayonnaise	Mayonnaise fleurial	Pot : 220 gr, 450gr, 680gr Flacon : 200gr, 395gr
	Sauce mayonnaise	Pot : 235gr, 485gr, 700gr Flacon : 220 gr, 420gr
	Moutarde de Dijon Fleurial	Pot : 350gr, 150gr Douce : 230 gr
	Sauce barbecue Fleurial	Flacon : 250gr, 470gr
	Ketchup Fleurial	Flacon : 220gr, 435gr
	Mayonnaise ail et fines herbes Fleurial	Flacon : 200 gr, 395gr
	Sauce vinaigrettes Fleurial plusieurs saveurs	500ml
	Harissa Fleurial	Pot : 150 gr
Sauces pour professionnels	Foody's mayonnaise	Seau : 3 kg Bouteille : 900gr
	Sauce Foody's	Sauce Foody's

### I.3 Présentation de l'entreprise CEVITAL

Cevital est la première entreprise privée algérienne fondée en Mai 1998. Elle se compose de plusieurs unités de production dotées de technologies les plus avancées et poursuit son expansion à travers divers projets en cours de réalisation. Cevital occupe la troisième place parmi les entreprises Algériennes en termes de chiffres d'affaires. Elle emploie 1800 salariés et se distingue comme le leader du secteur agroalimentaire en Afrique, tout en ayant une présence à l'international.

Elle possède également des silos portuaires ainsi qu'un terminal de déchargement portuaire d'une capacité de 2000 tonnes/heure, ce qui en fait le premier terminal de déchargement portuaire en Méditerranée [4].

#### I.3.1 Organigramme de l'entreprise

Le groupe Cevital est organisé en diverses directions, chacune spécialisée dans un secteur particulier de son large éventail d'activités comprenant la direction des finances et comptabilité, la direction des ressources humaines, la direction commerciale, etc.

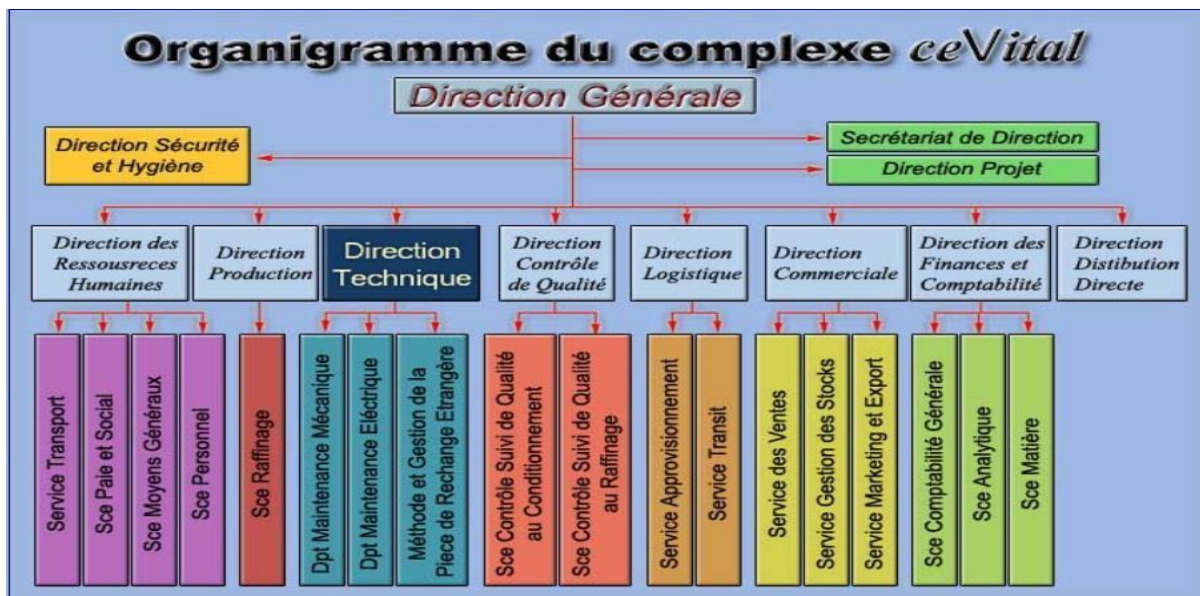


Figure I.1– Organisation du complexe CEVITAL

## **Chapitre I : Présentation de l'unité de conditionnement d'huile**

### **I.3.2 Activités de l'entreprise**

Les activités de CEVITAL sont principalement axées sur la production et la commercialisation d'huiles végétales, de sucre, de margarine et de graisses végétales, de boissons, ainsi que sur la production d'énergie électrique, réparties au sein de plusieurs unités de production [5] :

- 1- Deux raffineries de sucre
- 2- Une unité de sucre liquide
- 3- Une raffinerie d'huile
- 4- Une margarinerie
- 5- Une unité de conditionnement d'eau minérale
- 6- Une unité de fabrication et de conditionnement de boissons rafraîchissantes
- 7- Une conserverie
- 8- Une unité de fabrication de chaux calcinée

### **I.4 Unité de conditionnement d'huile**

L'unité de conditionnement d'huile à CEVITAL compte parmi les plus importantes unités de production du complexe qui a donné une large ampleur à ce groupe [6], est une installation moderne et performante qui assure l'ensemble du processus de transformation des préformes en bouteilles d'huile prêtes à la vente. Elle garantit la qualité et la sécurité alimentaire des produits finis, tout en optimisant l'efficacité de la production et de la distribution.

#### **I.4.1 Organisation de travail à l'unité de conditionnement d'huile**

L'unité emploie environ 334 personnes, réparties entre opérateurs, techniciens et ingénieurs, qui assurent la production en continu grâce à un système de rotation sur trois équipes. Lors des périodes de forte demande, jusqu'à 1300 travailleurs temporaires viennent renforcer les effectifs. Le site de Cevital à Béjaïa compte au total 3800 ingénieurs et collabore régulièrement avec près de 900 prestataires externes pour diverses missions.

#### **I.4.2 Lignes de conditionnement d'huile**

L'unité de conditionnement d'huile comprend plusieurs lignes, comme indiqué par Figure I.2.

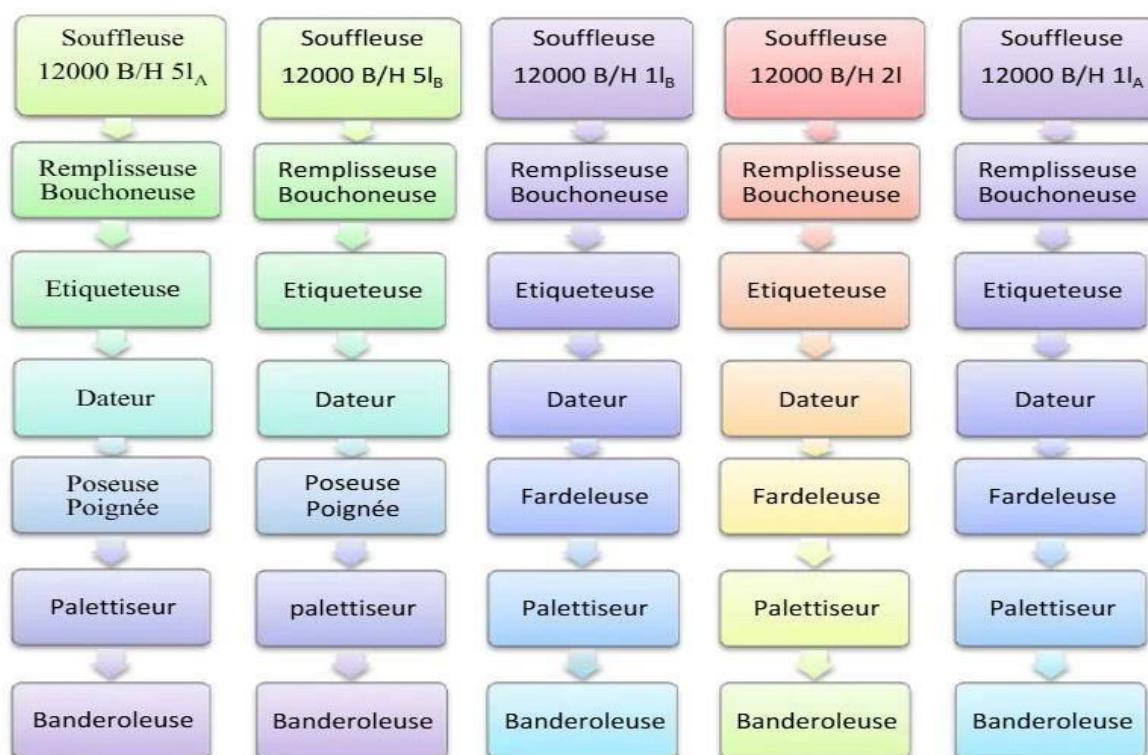


Figure I.2–Différentes lignes de production d'huile

### I.4.3 Etude des lignes de production d'huile

#### • Souffleuse

La machine de soufflage transforme les préformes en bouteilles de différentes tailles grâce à un procédé rapide. Elle produit jusqu'à 12 000 bouteilles/heure. Le processus inclut : chauffage des préformes, étirement, pré-soufflage à 7 bars, soufflage final à 40 bars, démoulage, refroidissement, et dégazage pour garantir une production efficace et précise.

#### • Soutireuse-bouchonneuse

La machine est essentielle dans le processus de conditionnement de l'huile en bouteilles. Elle remplit avec précision les bouteilles vides, les ferme hermétiquement, puis les transporte vers les étapes suivantes via un convoyeur mécanique. Le conditionnement s'effectue en plusieurs étapes :

- 1- Préparation : Nettoyage et installation des composants nécessaires.
- 2- Chargement : Remplissage du réservoir avec l'huile destinée à l'embouteillage.
- 3- Réglage des paramètres : Ajustement du débit et de la température pour un remplissage



## **Chapitre I : Présentation de l'unité de conditionnement d'huile**

homogène.

4- Remplissage : Transfert de l'huile vers les bouteilles.

5- Fermeture : Application des bouchons ou capsules pour garantir l'étanchéité.

### **• Etiqueteuse**

L'étiqueteuse est un équipement clé du conditionnement. Elle applique rapidement et avec précision les étiquettes sur les bouteilles, garantissant leur identification et leur conformité aux normes. Son fonctionnement se déroule en plusieurs étapes :

1- Préparation des étiquettes : Chargement et transfert vers la tête d'application.

2- Réglage des paramètres : Définition des critères d'étiquetage (vitesse, dimensions, quantité) via l'interface de commande

3- Positionnement des bouteilles : Alignement sur le convoyeur pour une pose précise des étiquettes.

4- Détection des bouteilles : Localisation automatique par capteurs pour un étiquetage précis.

5- Application des étiquettes : Fixation uniforme des étiquettes à l'aide de rouleaux ou brosses.

6- Contrôle qualité : Inspection automatisée via capteurs et caméras pour détecter les défauts d'étiquetage.

### **• Dateuse**

La machine de marquage imprime les informations essentielles sur les emballages pour assurer la traçabilité et la conformité des produits. Le processus de marquage comprend :

1- Préparation : Installation des consommables et paramétrage des données à imprimer.

2- Réglage : Configuration des paramètres d'impression (position, format, contenu).

3- Positionnement : Alignement précis des produits sur le convoyeur.

4- Détection : Utilisation de capteurs pour un déclenchement optimal de l'impression.

5- Impression : Marquage des emballages via différentes technologies (jet d'encre, transfert thermique, laser...).

6- Contrôle qualité : Vérification automatique de la lisibilité et de la conformité des marquages.

## **Chapitre I : Présentation de l'unité de conditionnement d'huile**

### **• Fardeleuse**

La machine est automatisée, simple à concevoir et facile d'accès. Elle sert à regrouper les bouteilles avec un film thermo rétractable. De type à barre de soudure, elle superpose des films sur le fond du paquet. Les principaux éléments de cette machine sont :

- 1- Tapis d'alimentation : Chaînes thermoplastiques à faible friction, avec entrée progressive et couloirs pour guider les produits.
- 2- Bobines de films : Une en cours d'utilisation et l'autre en attente, avec contrôle de la tension via un frein progressif.
- 3- Mandrin porte bobine.
- 4- Barre de soudure manuelle : Utilisée pour connecter le film lorsque la bobine est terminée.
- 5- Coupe film : Système de couteau rotatif sans contraste.
- 6- Résistance du tunnel de thermo-rétrécissement.
- 7- Armoire électrique principale : Inclut un conditionneur d'air.
- 8- Système de ventilation : Refroidit les fardeaux à la sortie du tunnel.

### **• Tapis roulant**

Il permet de transporter les fardeaux depuis la sortie de la fardeleuse jusqu'à l'entrée du palettiseur.



Figure I.3–Tapis roulant

## **Chapitre I : Présentation de l'unité de conditionnement d'huile**

### **• Palettiseur**

La machine est conçue pour empiler plusieurs couches de fardeaux ou cartons sur une palette.

Elle se compose des éléments suivants :

- 1- Palettiseur : Assure l'empilement des produits sur la palette.
- 2- Convoyeur : Transporte les produits vers le palettiseur, et les palettes vides sont alimentées, tandis que les palettes pleines sont évacuées.
- 3- Magasin de palettes : Stocke les palettes vides avant utilisation.
- 4- Installation pneumatique : Alimente le système en air comprimé pour faire fonctionner certains composants.
- 5- Tableau électrique : Gère l'ensemble du fonctionnement de la machine.

### **• Banderoleuse**

Cette machine emballe la charge palettisée à l'arrêt en faisant tourner un bras porte-bobine, assurant un enrobage stable et compact avec un film étirable multicouche, non toxique.

## **I.5 Présentation du travail**

Ce projet vise à automatiser la palettisation dans l'unité de conditionnement d'huile grâce à la commande d'un bras robotisé. Sa fonction principale est d'assurer le transfert automatisé de bouteilles de 2 Litres du convoyeur vers la palette.

Ce système remplace les opérations manuelles et semi-automatisées, permettant ainsi d'améliorer l'efficacité, de réduire les erreurs humaines, d'accélérer le cycle de production et d'assurer une précision optimale dans le conditionnement des bouteilles de 2 litres.

## **I.6 Conclusion**

Dans ce chapitre, on a présenté l'entreprise CEVITAL, au sein de laquelle on a accompli notre projet Master et l'unité de conditionnement d'huile qui fait l'objet de notre travail.

Lors de notre stage à l'entreprise, exactement dans l'unité de conditionnement d'huile, on a pu appréhender toutes les étapes du processus de production, depuis la fabrication jusqu'à l'expédition d'huile, ainsi que l'enchaînement et le fonctionnement des différentes machines utilisées

# **Chapitre II : Généralités sur les bras de robots et les systèmes automatisés**

### **II.1 Introduction**

Les bras robotiques et les systèmes automatisés sont essentiels pour améliorer l'efficacité des processus industriels en prenant en charge des tâches répétitives. Cela contribue à augmenter la productivité, la précision et la qualité, tout en réduisant les coûts.

Ce chapitre offre une vue d'ensemble sur la robotique, en définissant ses principes fondamentaux, en explorant les différents types de robots et leurs applications. Il traite également l'architecture d'un bras manipulateur, ses principaux composants et types, ainsi que les variateurs de vitesse et les systèmes automatisés, y compris les automates programmables industriels (API) et leurs langages de programmation.

### **II.2 Généralités sur la robotique**

#### **II.2.1 Définition**

Les bras robotiques sont des systèmes programmables qui imitent les mouvements humains pour automatiser diverses tâches. Le terme "robot" vient du mot tchèque "robota", signifiant "travail forcé". Depuis les années 1960, ils ont révolutionné des secteurs comme l'industrie manufacturière (assemblage, soudage), le médical (chirurgies assistées, prothèses intelligentes), ainsi que la logistique, la robotique spatiale et la défense. Grâce aux progrès technologiques, ces robots deviennent de plus en plus autonomes, intelligents et collaboratifs.

#### **II.2.2 Les disciplines impliquées dans la robotique**

La robotique est un domaine multidisciplinaire qui conçoit des machines capables d'effectuer des tâches de façon autonome ou semi-autonome.

1- mécanique : conçoit les structures, articulations et mécanismes des robots pour assurer stabilité et précision.

2- Électronique : utilise des capteurs et des circuits spécifiques pour détecter et contrôler les moteurs et l'alimentation des robots.

3- Automatique : optimise et régule les mouvements et les trajectoires des robots avec des algorithmes de contrôle.

4- Informatique et intelligence artificielle : programment et renforcent l'autonomie des robots grâce à des algorithmes et des interfaces de supervision.

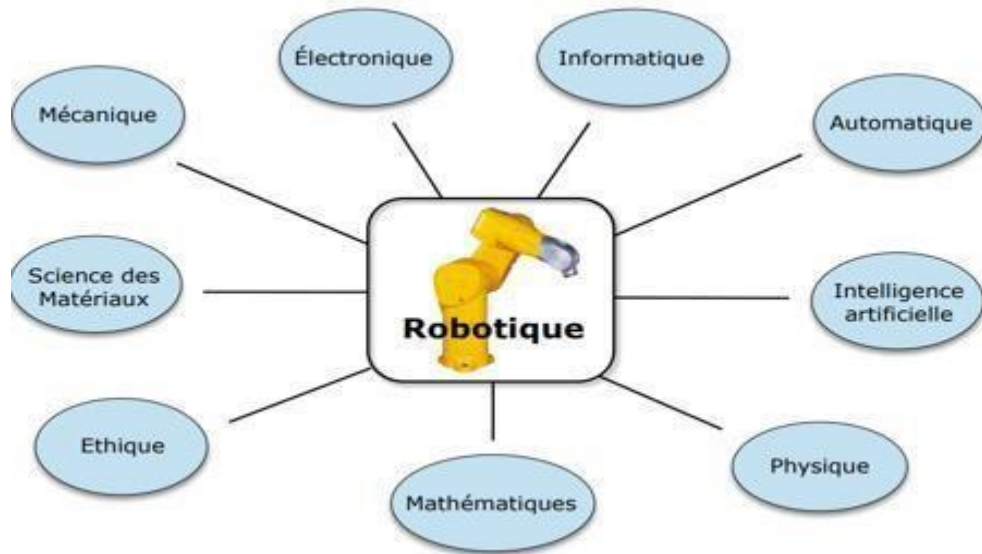


Figure II.1–Multidisciplinarité de la robotique

### II.2.3 Types des robots

Il existe trois catégories regroupent les principaux types de robots, basés sur leur usage, leur capacité de mouvement et leur degré d'autonomie.

#### • Robots manipulateurs

Les robots manipulateurs sont des structures articulées qui effectuent des tâches spécifiques telles que l'emballage, l'assemblage, la manutention ou la soudure. Couramment utilisés dans l'industrie, ils sont généralement fixés à un emplacement pour accomplir des tâches répétitives.



Figure II.2–Robot manipulateur

### • Robots mobiles

Les robots mobiles, équipés de roues, se déplacent dans divers environnements, même dangereux ou inaccessibles. Ils sont utilisés pour des missions d'inspection, de surveillance ou de transport, grâce à des capteurs et des logiciels avancés.



Figure II.3– Robot omnidirectionnel

### • Robots humanoïdes intelligents

Ils exécutent des tâches de façon autonome grâce à des algorithmes, capteurs et logiciels avancés, avec la capacité d'apprendre, de s'adapter à leur environnement et de résoudre des problèmes en temps réel.



FigureII.4–Robot humanoïde

### II.2.4 Domaines d'application

La robotique s'étend sur de nombreux secteurs et offre une variété d'applications. Voici quelques-uns des domaines clés dans lesquels elle est utilisée :

### • Palettisation industrielle

Dans l'industrie, la palettisation désigne l'opération consistant à regrouper un certain nombre de colis sur un support standardisé : la palette. Cette opération de groupage est généralement réalisée de manière automatisée à l'aide d'un équipement appelé palettiseur [7].



Figure II.5-Robot palettiseur



### • Médecine et chirurgie

La robotique médicale facilite des interventions chirurgicales plus précises et moins invasives, grâce à des dispositifs comme le système Da Vinci, qui soutient les chirurgiens dans des procédures complexes.



Figure II.6- Robot chirurgical

### • Transport et logistique

Dans ce domaine, les robots assurent la gestion des stocks, la préparation des commandes et le transport des marchandises, tandis que des véhicules autonomes sont développés pour optimiser la livraison.



Figure II.7– Robot de livraison autonome

### • Agriculture

Les robots agricoles effectuent des tâches comme la plantation, la récolte, le traitement des cultures et leur surveillance, permettant de réduire les coûts et d'améliorer la productivité.



Figure II.8-Robot agricole

## II.3 Architecture d'un bras manipulateur

Un robot manipulateur est essentiellement constitué de la base, des segments, des articulations, des actionneurs et de l'organe terminal. L'architecture globale est illustrée par la figure II.9.





Figure II.9—Architecture d'un bras robotisé

### II.3.1. Base

Elle est installée de manière permanente sur le lieu de travail, ce qui s'applique à la majorité des robots industriels.

### II.3.2 Articulation

Elle lie deux corps successifs en limitant le nombre de degrés de liberté, de l'un par rapport à l'autre [8]. On peut citer deux catégories d'articulations :

1- Articulation rotoïde : Il s'agit d'une articulation de type pivot, notée R, réduisant le mouvement entre deux corps à une rotation autour d'un axe commun. La situation relative entre les deux corps est donnée par l'angle autour de cet axe [8].

2- Articulation prismatique : Il s'agit d'une articulation de type glissière, notée P, réduisant le mouvement entre deux corps à une translation le long d'un axe commun. La situation relative entre les deux corps est mesurée par la distance le long de cet axe [8].

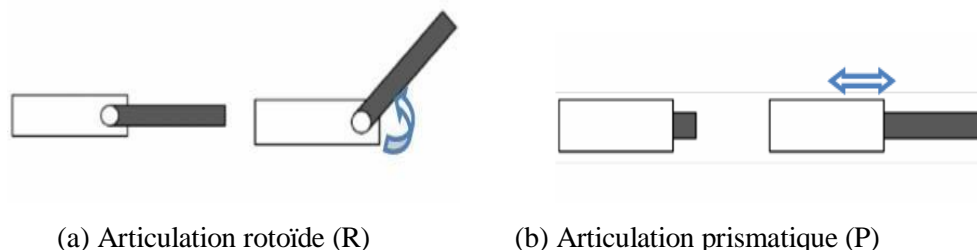


Figure II.10—Types d'articulation

### II.3.3 Segments

Des corps solides rigides capables de se déplacer par rapport à la base du support et entre eux.

### II.3.4 Actionneurs

Les actionneurs permettent au robot d'agir sur son environnement et de réaliser des tâches. Il existe plusieurs types d'actionneurs : électriques, pneumatiques et hydrauliques.

1- Actionneurs électriques : convertissent l'énergie électrique en mouvement mécanique. Les servomoteurs offrent un contrôle précis en boucle fermée, tandis que les moteurs pas à pas assurent un positionnement exact par déplacements fixes, sans capteurs.



Figure II.11– Servomoteur synchrone



Figure II.12– Moteur pas à pas

2- Actionneurs Pneumatiques : présents dans les robots compacts (2 à 4 articulations), génèrent des mouvements rapides et linéaires ou rotatifs grâce à l'air comprimé, adaptés aux tâches simples et rapides de sélection.

3- Actionneurs Hydrauliques : Pour les robots devant manipuler de très lourdes charges, les actionneurs sont le plus souvent hydrauliques, agissant en translation (vérins hydrauliques) ou en rotation [9].

### II.3.5 Organe terminal

C'est la partie qui interagit directement avec l'environnement. Il peut inclure des pinces ou griffes pour saisir et manipuler des objets, des mains robotiques pour des tâches précises, ou des capteurs pour la détection et la mesure, comme des caméras ou des capteurs de force et de température.

### II.3.6. Degrés de liberté d'un bras manipulateur

Correspondent aux mouvements indépendants de l'organe terminal du robot. En chaîne ouverte, chaque articulation ajoute un DDL. En chaîne fermée, les boucles cinématiques imposent des contraintes qui en réduisent le nombre.

## II.4 Morphologie des bras manipulateurs

### II.4.1 Structure cartésienne

Les robots cartésiens se déplacent uniquement sur des axes linéaires, ce qui facilite leur programmation et les rend parfaits pour accomplir des tâches répétitives. Ces robots, constitués de 3 articulations prismatiques en translation, sont appelés robots PPP.

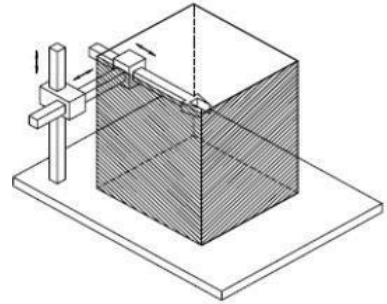


Figure II.13– Structure cartésienne

### II.4.2 Structure cylindrique

Ce type de robot a un porteur généralement composé d'une articulation rotoïde et de deux articulations prismatiques, permettant des déplacements en rotation et en translation, tout en assurant une excellente rigidité mécanique et une simplicité de conception. La géométrie cylindrique diffère de la géométrie cartésienne en ce que la 1<sup>ère</sup> articulation prismatique est remplacé par une articulation rotoïde (RPP) [10].

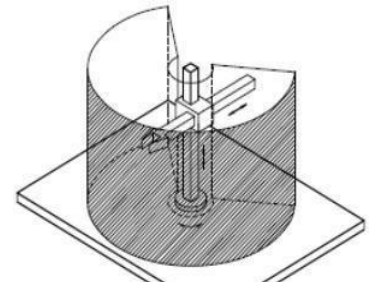


Figure II.14– Structure Cylindrique

### II.4.3 Structure sphérique (polaire)

Elle est composée de deux d'articulations rotoïdes et d'une articulation prismatique. Elle diffère de la géométrie cylindrique en ce que la 2<sup>ème</sup> articulation prismatique est remplacée par une articulation rotoïde (RRP) [10].

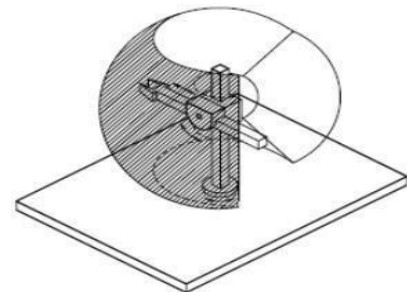


Figure II.15– Structure sphériques

### II.4.4 Structure anthropomorphe

Ce mécanisme possède trois articulations rotoïdes (RRR). La première effectue une rotation autour d'un axe perpendiculaire aux deux suivants. Tandis que les deux autres s'opèrent autour d'axes parallèles. En raison de sa ressemblance avec le bras humain, la 2<sup>e</sup> articulation est appelée l'articulation de l'épaule et la 3<sup>e</sup> l'articulation du coude puisqu'elle relie le bras avec l'avant-bras.

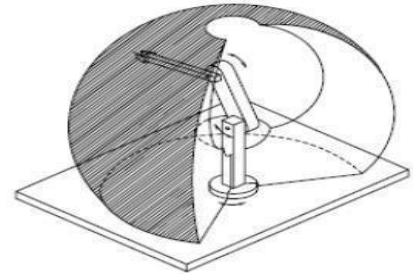


Figure II.16– Structure anthropomorphe

### II.4.5 Structure SCARA

Le robot SCARA (Selective Compliance Assembly Robot Arm) est de type RRP, avec deux liaisons rotoïdes autour d'axes parallèles et une liaison prismatique dans la même direction. Une liaison rotoïde supplémentaire peut être ajoutée à l'extrémité du bras pour augmenter sa mobilité.

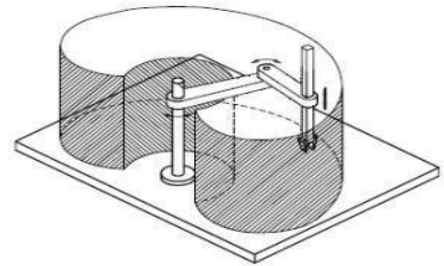


Figure II.17– Structure SCARA

## II.5 Variateurs de vitesse

Les variateurs de vitesse sont des dispositifs électroniques qui permettent de réguler la vitesse de rotation des moteurs électriques de manière précise et efficace. Ils sont souvent utilisés dans les systèmes de contrôle industriel pour optimiser la consommation d'énergie, améliorer les performances des machines et réduire les coûts de maintenance [11].



Figure II.18– Variateur de vitesse

### II.5.1 Constituants des variateurs de vitesse

Les variateurs de vitesse se composent de cinq éléments présentés sur la figure II.19 [11].

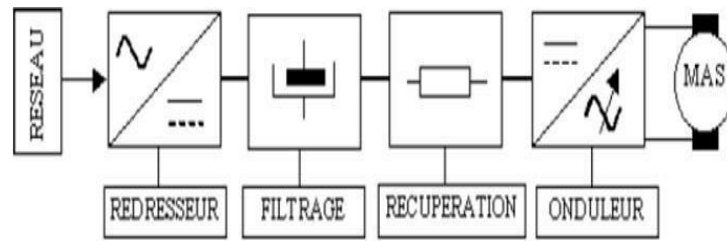


Figure II.19– Synoptique des variateurs de vitesse [11].

### II.5.2 Variateurs de vitesse dans la robotique

Les variateurs de vitesse régulent la vitesse et le couple des moteurs en adaptant l'alimentation électrique selon les besoins de l'application. Voici leur fonction dans les différents types de moteurs utilisés dans les robots :

- 1- Pour les moteurs à courant continu (DC) : un variateur ajuste la tension pour contrôler la vitesse et la direction du moteur, souvent à l'aide d'un pont en H.
- 2- Pour moteurs à courant alternatif (AC) : Il convertit le courant alternatif (AC) en courant continu (DC), puis utilise un onduleur pour moduler la fréquence et ajuster la vitesse du moteur.
- 3- Pour les moteurs pas à pas : Il ajuste la fréquence et la séquence des impulsions pour contrôler la vitesse et la position du moteur.

## II.6 Systèmes automatisés

Un système automatisé est un ensemble de dispositifs mécaniques, électroniques et informatiques, conçu pour exécuter un cycle de travail de manière autonome ou semi-autonome. Son évolution débute avec les automates antiques, s'accélère avec la révolution industrielle, puis se perfectionne avec les API au 20<sup>e</sup> siècle. Aujourd'hui, les technologies intelligentes renforcent son impact dans l'industrie, la robotique et la domotique.

### II.6.1 Système automatisé et la robotique

Les systèmes automatisés sont essentiels en robotique, contrôlant et coordonnant les robots pour des tâches précises et autonomes. Grâce aux capteurs, automates et algorithmes, ils adaptent les actions des robots, améliorant ainsi l'efficacité, la sécurité et l'adaptabilité, même dans des environnements complexes.

### II.6.2 Structure d'un système automatisé

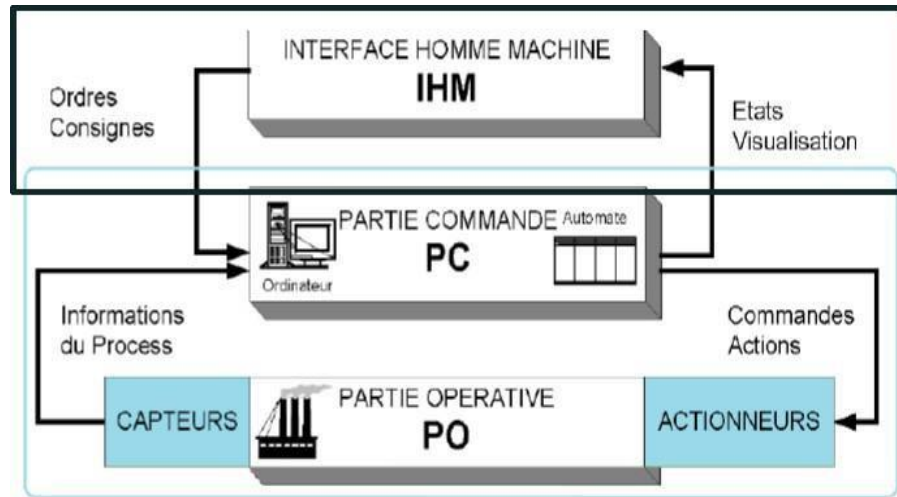


Figure II.20– Structure d'un système automatisé.

1- La partie commande (PC) : constituée d'un contrôleur, elle traite les informations de la partie opérative(PO), génère les ordres pour les actionneurs et reçoit les commandes de l'IHM, tout en transmettant l'état du système pour la supervision en temps réel.

2- La partie opérative (OP) : appelée « partie puissance », elle inclut les capteurs, pré-actionneurs et actionneurs qui collectent les données, exécutent les commandes de la partie commande et transmettant l'état du processus pour un contrôle en temps réel.

3- L'interface Homme Machine (IHM) : est le système de présentation des informations du système automatisé permettant la visualisation du procédé, la prise en compte des consignes et ordre des opérateurs humains sur le procédé [12].

## II.7 Généralités sur les automates programmables

### II.7.1 Définition

Un Automate Programmable Industriel (API) est un dispositif électronique qui automatise et supervise les processus industriels, traitant les signaux des capteurs et pilotant les actionneurs en temps réel pour optimiser les performances et l'efficacité des équipements.

### II.7.2 Structure interne d'un automate programmable industriel

De manière générale, l'automate programmable industriel est composé de plusieurs éléments de base décrits comme suit :

- 1- Un processeur (ou Central Processing Unit, CPU) : sa fonction est d'exécuter le programme, gérer les entrées et sorties, surveiller et diagnostiquer l'automate via des tests réguliers, et établir la communication avec le terminal de programmation.
- 2- Une mémoire : elle stocke les instructions du programme et diverses informations. Elle peut être de type RAM (modifiable, mais effacée lors d'une coupure de courant) ou EEPROM (lecture seule).
- 3- Des interfaces entrées/sorties : Les entrées reçoivent les données des capteurs, et les sorties contrôlent les dispositifs externes. Leur nombre dépend du modèle et des exigences de l'automate.
- 4- L'alimentation : Elle convertit la tension alternative en une faible tension continue (24V) pour alimenter le processeur et les modules. Parfois, l'alimentation n'est pas intégrée et l'automate doit être alimenté par une source externe.
- 5- L'interface de communication : Elle permet l'échange de données entre l'API et d'autres équipements via des réseaux, facilitant la vérification des périphériques, l'acquisition de données, la synchronisation et la gestion des connexions.
- 6- Périphérique de programmation : Elle permet d'insérer le programme dans la mémoire du processeur. Le programme est d'abord développé sur ce périphérique, puis transféré vers la mémoire de l'API.

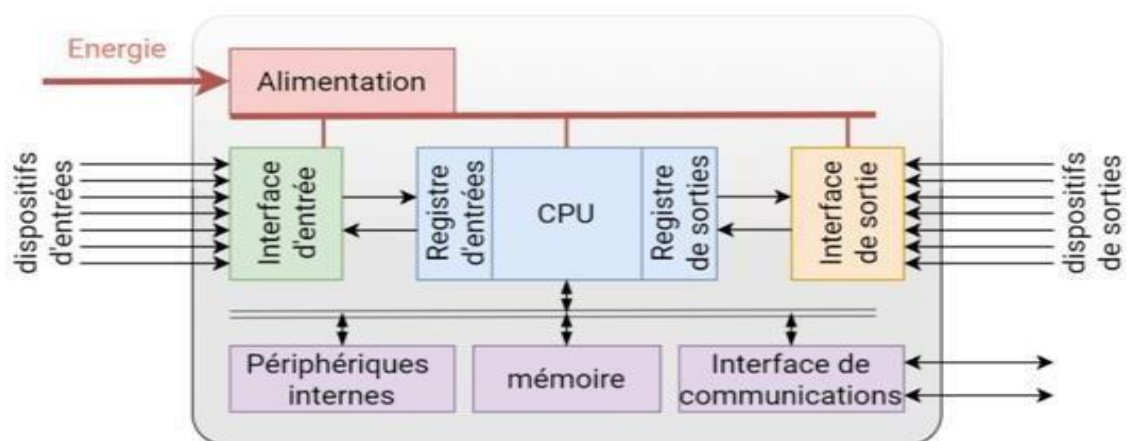


Figure II.21- Structure interne d'un automate programmable industriel

### **II.7.3 Langages de programmation**

La norme industrielle CEI 61131-3 définit plusieurs langages qui peuvent être utilisés pour la programmation des automates. Ces langages se répartissent en deux principales catégories.

#### **• Langages graphiques**

- SFC « Sequential Function Chart » ou GRAFCET : Représenter le fonctionnement séquentiel des automatismes à travers des étapes associées à une ou plusieurs actions, des transitions et des liaisons orientées.

- LD « Ladder Diagram » ou langage à contacts : exprime des équations booléennes (vrai/faux) à l'aide de symboles spécifiques (contacts, temporisateurs, compteurs...), structuré entre deux barres d'alimentation.

- FBD « Function Block Diagram » ou Schéma Par Bloc : Représente des fonctions à l'aide de blocs reliés par des variables d'entrée et de sortie, en utilisant des symboles logiques (portes logiques, bascules...).

#### **• Langages textuels**

-ST « Structured Text » : Langage informatique évolué, structuré en une suite d'énoncés.

- IL « Instruction List » : Proche de l'assembleur, adapté aux petites applications et basé sur la mise à jour d'un résultat courant.

### **II.7.4 L'exécution des programmes**

Deux programmes s'exécutent simultanément dans la CPU : le système d'exploitation et l'application, qui est organisée en blocs.

1- Bloc d'organisation (OB) : détermine l'ordre d'exécution des différentes parties du programme dans l'automate.

2- Bloc fonctionnel (PB) : Subordonné à un OB, structurant le programme en unités et associé à un DB.

3- Bloc de données (DB) : Assure le stockage des variables et la gestion de la mémoire du programme dans un automate.



### II.7.5 Critères et choix des API

Le choix d'un API dépend des besoins du cahier des charges. On doit tenir compte de plusieurs critères à savoir :

- Entrées/sorties intégrées selon le nombre et le type requis.
- Extensions possibles pour ajouter des modules si besoin.
- Processeur à choisir selon la mémoire, la vitesse et les fonctions.
- Communication avec d'autres systèmes via des protocoles standards.
- Prix, garantie et services à évaluer.

### II.7.6 Présentation de l'automate Siemens S7-1200

Le Siemens S7-1200 est une série de PLC destinée à l'automatisation industrielle, adaptée à l'automatisation des processus, à la gestion et aux systèmes de contrôle de la production.

Le S7-1200 propose une interface de programmation intuitive (SIMATIC STEP7 Basic) et peut être étendu avec des modules pour des besoins spécifiques. Il dispose d'un connecteur d'alimentation, de connecteurs amovibles, de LEDs d'état pour les E/S intégrées et d'un connecteur PROFINET pour la communication avec une console de programmation et d'autres systèmes via le réseau PROFINET.



Figure II.22– Automate SIMATIC S7 1200

#### • Modules d'extension du S7-1200

Le S7-1200 offre une large variété de modules d'extension et accessoires pour étendre ses capacités. Parmi eux on trouve :

- CPU : exécute les programmes et gère la communication avec les autres modules.
- Modules I/O : permettent la connexion de signaux physiques (entrées/sorties numériques ou analogiques, combinés).

## **Chapitre II : Généralités sur les bras de robots et les systèmes automatisés**

---

-Modules de communication : protocoles de communication industrielle.

-Modules de sécurité : répondent aux exigences des applications sensibles.

Modules d'extension : ajoutent des capacités supplémentaires.

-Modules d'alimentation : alimentent le système.

-Modules de fonction : dédiés à des tâches spécifiques (température, comptage, etc.).

### **II.8 Conclusion**

Dans ce chapitre, on a présenté les bases des systèmes automatisés et des bras robotisés, en expliquant les concepts clés de la robotique et des API en automatisation industrielle. Ces notions préparent à la mise en œuvre pratique de la planification de trajectoires sous MATLAB et la commande d'un bras robotisé avec TIA Portal V15.

# **Chapitre III : Modélisation, commande et planification de trajectoires d'un bras robotisé**

### III.1 Introduction

La conception et le contrôle des robots reposent sur l'élaboration de modèles mathématiques permettant de décrire et de relier différents aspects de leur fonctionnement. Parmi ceux-ci, on trouve les modèles de transformation entre l'espace articulaire, qui représente la configuration du robot à travers ses articulations, et l'espace opérationnel où l'on définit la position et l'orientation de l'organe terminal.

L'ensemble de ces modèles est indispensable pour permettre la simulation, l'analyse et la commande des mouvements du robot. Comprennent notamment :

- les modèles géométriques directs et inverses qui établissent respectivement la position de l'organe terminal à partir des paramètres articulaires et l'inverse ;
- les modèles cinématiques directs et inverses, qui décrivent la relation entre les vitesses des articulations et celle de l'organe terminal ;
- les modèles dynamiques permettent de formuler les équations du mouvement du robot, en mettant en relation les efforts (forces ou couples) fournis par les actionneurs avec les positions, vitesses et accélérations des articulations.

Ces modèles sont essentiels pour planifier les trajectoires du robot dans l'espace articulaire, en définissant ses mouvements tout en respectant les contraintes mécaniques et dynamiques liées à la tâche à accomplir.

### III.2 Représentation des coordonnées d'un bras de robot

La représentation d'un point dans l'espace est essentielle en robotique pour définir la position et l'orientation de l'effecteur terminal. Elle peut se faire selon différents systèmes de coordonnées, principalement dans l'espace articulaire ou l'espace opérationnel.

- **L'espace articulaire**

La configuration d'un robot est représentée par un vecteur, représentant les déplacements angulaires ou linéaires des articulations selon les liaisons mécaniques.

$$\theta = [\theta_1, \theta_2 \dots, \theta_m]^T$$

où  $m$  est le nombre de degrés de liberté du bras. Ce vecteur prend ses valeurs dans l'espace

articulaire.

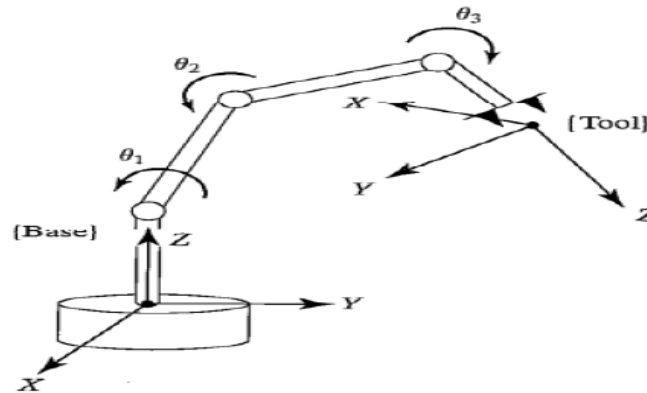


Figure III.1–Représentation de l'espace articulaire

- **l'espace opérationnel**

Définit la position et l'orientation de l'effecteur dans un repère cartésien. Ce vecteur est donné par

$$X = [x_1, x_2, \dots, x_n]^T$$

Où n est la dimension de l'espace opérationnel.

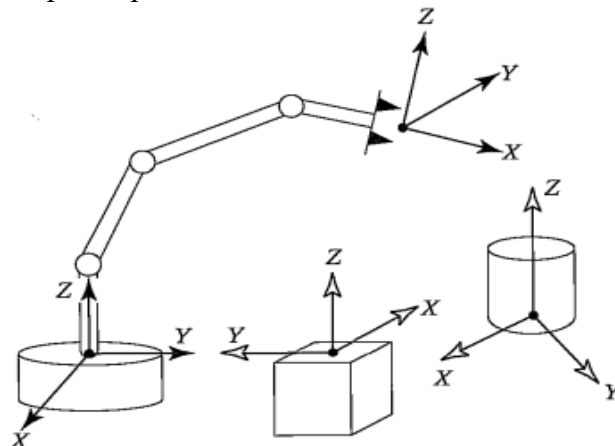


Figure III.2–Représentation de L'espace opérationnelle

### III.2.1 Représentation d'un point

Dans un repère fixe R d'axes X, Y, Z .la position d'un point P associé à un corps rigide est donnée par un vecteur colonne , exprimé dans ce repère.

$$P = [P_x, P_y, P_z]^T$$

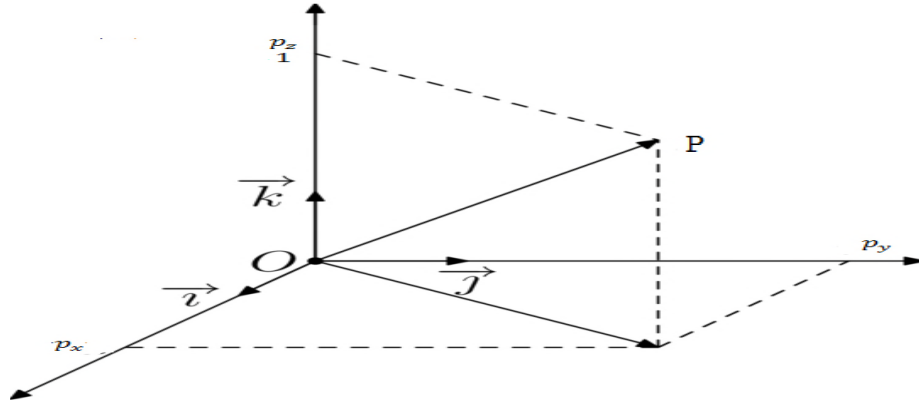


Figure III.3–Représentation d'un point dans l'espace

### III.2.2 Orientation et matrices de rotation

Soient deux repères A et B dont les origines coïncident ( $O_0 = O_1$ ). L'orientation du repère B par rapport au repère A peut être représentée par le produit de trois matrices élémentaires représentant les rotations autour des axes X, Y et Z [13].

#### -Rotation d'un angle $\theta$ autour de X:

$$\text{Rot}(X, \theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{bmatrix}$$

#### -Rotation d'un angle $\theta$ autour de Y:

$$\text{Rot}(Y, \theta) = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix}$$

#### -Rotation d'un angle $\theta$ autour de Z:

$$\text{Rot}(Z, \theta) = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

### III.3 Transformations homogènes

La matrice homogène  $T_j^i$  représente la transformation permettant de passer du repère  $R_i$  au repère

$$R_j [6]. \quad T_j^i = \begin{bmatrix} R_{3 \times 3} & P_{3 \times 1} \\ N_{1 \times 3} & 1 \end{bmatrix}, \quad N_{1 \times 3} = [0 \ 0 \ 0]$$

Avec :  $R_{3 \times 3}$  = Représente la matrice de rotation propre.

### III.3.1 Matrices de rotation

$$\text{Rot}(X, \theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta & 0 \\ 0 & \sin \theta & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix};$$

$$\text{Rot}(Z, \theta) = \begin{bmatrix} \cos \theta & -\sin \theta & 0 & 0 \\ \sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\text{Rot}(Y, \theta) = \begin{bmatrix} \cos \theta & 0 & \sin \theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \theta & 0 & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix};$$

### III.3.2 Matrice de translation

La matrice de translation  $T_r$  permet de déplacer un point ou un repère dans l'espace sans modifier son orientation, en appliquant un vecteur de déplacement  $t_r$  (x, y, z) à ses coordonnées.

$$T_r = \begin{bmatrix} 1 & 0 & 0 & t_{rx} \\ 0 & 1 & 0 & t_{ry} \\ 0 & 0 & 1 & t_{rz} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

### III.4 Modélisation géométrique

Le modèle géométrique permet de calculer la position et l'orientation de l'effecteur, données par le vecteur des variables opérationnelles  $X(t) = [x_1, x_2, \dots, x_n]^T \in R^n$  ou la matrice de transformation homogène équivalente, à partir des variables articulaires  $\theta(t) = [\theta_1, \theta_2, \dots, \theta_m]^T \in R^m$ , avec  $n \leq m$ . Dans le cas d'une chaîne ouverte simple, le modèle géométrique direct peut être représenté par la relation  $x(t) = f[\theta(t)]$  ou par une matrice de transformation  $T_0^n$  :  $T_0^n = T_0^1 \cdot T_1^2 \dots T_{n-1}^n$ .

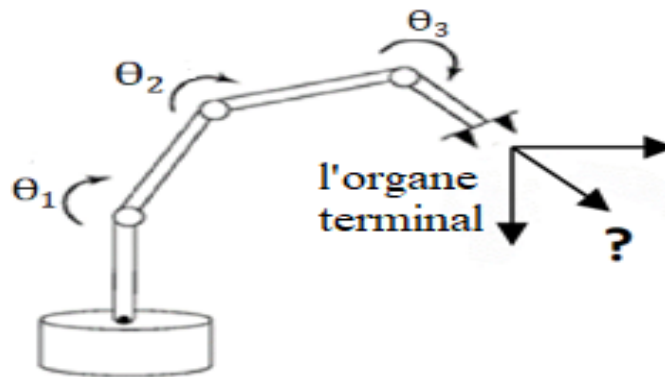


Figure III.4—Représentation du MGD

### III.4.1 Procédure de calcul du MGD par la convention DH

Pour définir les paramètres Denavit-Hartenberg (DH), on suit des règles précises :

1. Placer le repère  $R_{i-1}$  est place a la fin du corps  $c_{i-1}$ .
2. L'axe  $z_{i-1}$  suit l'axe de mouvement de l'articulation  $i$ .
3. L'axe  $x_{i-1}$  est aligné entre les articulations  $i$  et  $i + 1$ .
4. L'axe  $y_i$  est choisi de manière à compléter le repère orthonormé.

Pour passer de  $R_{i-1}$  à  $R_i$  on peut suivre le chemin suivant [14]:

1. Rotation  $(z_{i-1}, \theta_i)$ . avec  $\theta_i$  : angle entre  $x_{i-1}$  et  $x_i$  autour de  $z_{i-1}$ .
2. Translation  $(z_{i-1}, d_i)$ .  $d_i$  : distance de  $O_{i-1}$  à l'intersection de  $z_{i-1}$  avec  $x_i$ , le long de  $z_{i-1}$ .
3. Translation  $(x_i, a_i)$ .  $a_i$ : distance entre  $z_{i-1}$  et  $z_i$ , le long de  $x_i$ .
4. Rotation  $(x_i, \alpha_i)$ .  $\alpha_i$ : angle entre  $z_{i-1}$  et  $z_i$ , autour de  $x_i$ .

Ainsi, la matrice de transformation entre deux articulations successives est :

$$T_i^{i+1} = T_z \cdot T_x = \begin{bmatrix} c\theta_i & -c\alpha_i \cdot s\theta_i & s\alpha_i \cdot s\theta_i & a_i c\theta_i \\ s\theta_i & c\alpha_i \cdot c\theta_i & -s\alpha_i \cdot c\theta_i & a_i s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

### III.4.2 Modèle géométrique du robot PUMA 560

Le PUMA 560 est un bras robotisé articulé doté de 6 axes, capable de manipuler une charge maximale de 2,5 kg. Il possède une portée de 864 mm et offre une excellente précision avec une répétabilité de 0,1mm.

Dans le cadre de ce travail, ce robot a été choisi pour modéliser et simuler un bras à 3 DDL. La Figure III.5 montre la configuration des repères associés aux différentes articulations du robot.



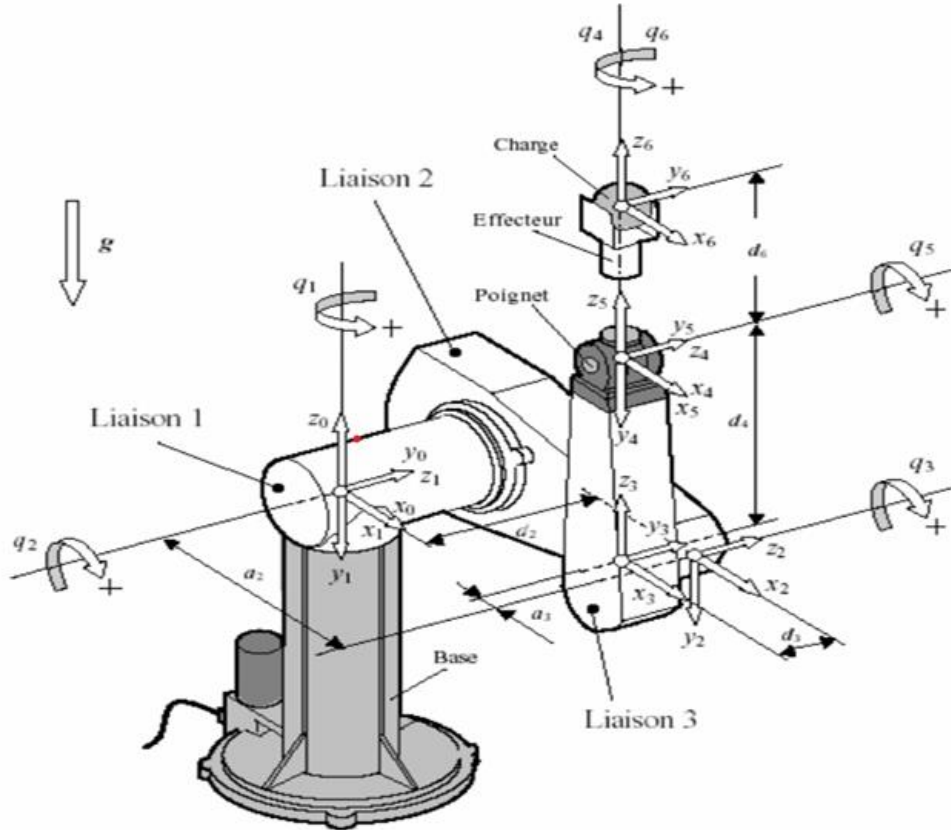


Figure III.5– Association des repères et les paramètres DH pour le bras PUMA 560 [15].

En tenant compte de la position de l'origine du repère de base, il est possible de définir les paramètres de DH.

Tableau III.1 : Configuration des paramètres DH du robot PUMA 560[15]

Articulation	$\theta_i$	$d_i$	$a_i$	$\alpha_i$
$R_0 \rightarrow R_1$	$\theta_1$	0	0	-90
$R_1 \rightarrow R_2$	$\theta_2$	0.2435 mm	0.4318 mm	0
$R_2 \rightarrow R_3$	$\theta_3$	-0.0934 mm	-0.0203 mm	90
$R_3 \rightarrow R_4$	$\theta_4$	0.4331 mm	0	-90
$R_4 \rightarrow R_5$	$\theta_5$	0	0	90
$R_5 \rightarrow R_6$	$\theta_6$	-	0	0

Dans cette étude, l'influence dynamique des trois dernières articulations est jugée négligeable en comparaison avec celle des trois premières. L'étude se concentre donc sur le comportement des

trois premiers axes du robot PUMA 560, en considérant que le poignet reste fixe à sa position initiale ( $\theta_4 = 0, \theta_5 = 0, \theta_6 = 0$ ).

À partir des trois premières lignes de la table des paramètres DH, il est possible de construire les matrices de transformation homogène reliant le repère  $R_0$  au repère  $R_3$  :

$$T_0^1 = \begin{bmatrix} C1 & 0 & -S1 & 0 \\ S1 & 0 & C1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; T_1^2 = \begin{bmatrix} C2 & -S2 & 0 & a2C2 \\ S2 & C2 & 0 & a2C2 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; T_2^3 = \begin{bmatrix} C3 & 0 & -S3 & a2C2 \\ S3 & 0 & C3 & a2C2 \\ 0 & 1 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Le MGD du robot est donné par :

$$T_0^3 = T_0^1 \cdot T_1^2 \cdot T_2^3 = \begin{bmatrix} C1C23 & -S1 & C1S23 & C1(a2C2 + a3C23) - (d2 + d3)S1 \\ S1C23 & C1 & S1S23 & S1(a2C2 + a3C23) + (d2 + d3)C1 \\ -S23 & 0 & C23 & -(a2S2 + a3S23) \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (III.1)$$

Avec  $Ci = \cos \theta_i$ ;  $Si = \sin \theta_i$ ;  $Cij = \cos (\theta_i + \theta_j)$ ;  $Sij = \sin (\theta_i + \theta_j)$

#### III.4. Modèle géométrique inverse

Le MGI d'un robot consiste à calculer les coordonnées articulaires correspondant à une situation particulière de l'organe terminal. Le modèle s'écrit [14] :

$$\theta = f^{-1}(x) \quad (III.2)$$

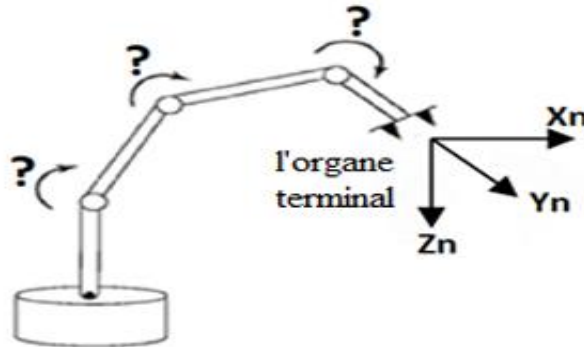


Figure III.6—Représentation du MGI

Après avoir calculé la matrice de transformation, celle-ci est utilisée pour le chemin inverse, c'est-à-dire pour déterminer les variables articulaires à partir de la position et de l'orientation de l'effecteur.

### Modèle géométrique inverse du robot PUMA 560

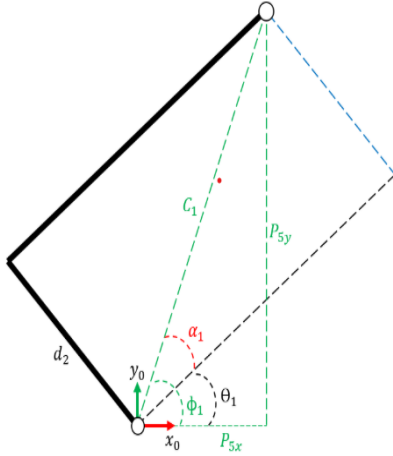


Figure III.7—plan x0 – y0

(Vue de dessus du PUMA560) [16]

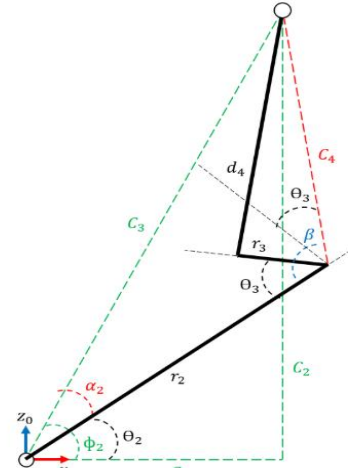


Figure III.8—plan x0 – z0

(Vue latérale du PUMA560) [16]

À partir des figures III.7 et III.8 nous obtiendrons les résultats suivants : [16]

$$\theta_1 = \text{atan2} (P_{5y}, P_{5x}) - \text{atan2} (D_1, \sqrt{1 - D_1^2}) \quad (\text{III.3.a})$$

$$\theta_2 = \text{atan2} (C_2, C_1) - \text{atan2} (\sqrt{1 - D_2^2}, D_2) \quad (\text{III.3.b})$$

$$\theta_3 = \text{atan2} (\sqrt{1 - D_3^2}, D_3) - 90 \quad (\text{III.3.c})$$

### III.5 Modélisation Cinématique

Soit  $X = [X_P^T \ X_r^T]^T$  une représentation quelconque dans le repère  $R_0$  de la situation du repère  $R_n$  fixé à un solide , les éléments  $X_P$  et  $X_r$  désignant respectivement la position et l'orientation opérationnelles du solide. Les relations entre les vitesses  $\dot{X}_P$  Et  $\dot{X}_r$  et les vecteurs vitesses  $v_0^n$  et  $\omega_0^n$  on du repère  $R_n$  sont telles que :

$$\begin{bmatrix} \dot{X}_p \\ \dot{X}_r \end{bmatrix} = \begin{bmatrix} \Omega_p & 0_3 \\ 0_3 & \Omega_r \end{bmatrix} \begin{bmatrix} v_n^0 \\ \Omega_n^0 \end{bmatrix} = \Omega \begin{bmatrix} v_n^0 \\ \Omega_n^0 \end{bmatrix} \quad (\text{III.4})$$

Les matrices  $\Omega_p$  et  $\Omega_r$  dépendant de la représentation choisie respectivement pour la position et pour l'orientation. Le modèle cinématique inverse a pour forme générale :

$$\dot{\theta} = J^T \dot{X} \quad (\text{III.5})$$

Dans le cas régulier où la matrice jacobéenne  $J$  est carrée d'ordre  $n$  et son déterminant est non nul, la méthode la plus générale consiste à la matrice inverse de  $J$  [16].

## III.6 Modélisation Dynamique

La modélisation dynamique d'un robot permet de décrire son comportement temporel en établissant une relation entre les actions mécaniques exercées, telles que les forces ou couples, et les mouvements produits au niveau de ses articulations, incluant positions, vitesses et accélérations. Elle joue un rôle fondamental dans les domaines de la commande et de la simulation, car elle permet soit de prévoir l'évolution des grandeurs articulaires à partir des couples appliqués, soit de déterminer les couples nécessaires pour générer un mouvement donné.

Le modèle dynamique direct (ou modèle de simulation) exprime les accélérations articulaires en fonction des positions, vitesses et couples des articulaires. Il est représenté par la relation suivante [16] :

$$\ddot{\theta} = g(\theta, \dot{\theta}, F_e) \quad (\text{III.6})$$

On représente le modèle dynamique inverse par une relation [16]

$$\tau = f(\theta, \dot{\theta}, \ddot{\theta}, F_e) \quad (\text{III.7})$$

Avec :

$\tau$  : Vecteur des couples / forces articulaires ;

$\theta$  : Vecteur des positions articulaires ;

$\dot{\theta}$  : Vecteur des vitesses articulaires ;

$\ddot{\theta}$  : Vecteur des accélérations articulaires ;

$F_e$  : Vecteurs de l'effort extérieur qu'exerce le robot sur l'environnement.

### III.7 Commande d'un bras robotisé

La commande a pour but de déplacer l'outil ou l'effecteur d'un bras robotique depuis une position initiale vers une position finale, tout en respectant une trajectoire définie. Pour cela, des méthodes de commande linéaires et non linéaires sont mises en œuvre.

#### III.7.1 Méthodes de commande

##### III.7.1.1 Régulation linéaire

Elle permet de contrôler précisément le mouvement d'un bras robotisé, en s'appuyant sur un modèle linéaire ou linéarisé du système.

Parmi les techniques classiques les plus utilisées, on retrouve :

- La commande PD, qui corrige l'erreur à partir de la position et de la vitesse.
- La commande PID, qui ajoute un terme intégral pour éliminer les erreurs persistantes.

#### Commande Proportionnelle Dérivée (PD)

Cette commande repose sur l'hypothèse que le bras robotique peut être représenté par un modèle simplifié, ce qui est valable dans des conditions de fonctionnement à basse vitesse et avec des composants de faible masse. Dans cette approche, l'objectif est de faire passer le bras robotique d'un état initial  $\theta_0$  à un état cible  $\theta_d$ . La stratégie de commande repose sur la mesure de l'écart entre la position actuelle du système et la position finale souhaitée.

#### Structure de la commande :

La commande PD est basée sur l'erreur entre l'état finale désiré et l'état instantané réel [13] :

$$e(t) = (\theta_d - \theta(t)) \quad \text{et} \quad \dot{e}(t) = -\dot{\theta}(t) \quad (\text{III.8-a})$$

L'état final désiré est généralement ramené vers 0, par translation de l'origine. Ainsi :

$$e(t) = -\theta(t) \quad \text{et} \quad \dot{e}(t) = -\dot{\theta}(t) \quad (\text{III.8-b})$$

Les profils des vitesses et des accélérations angulaires ne sont pas imposés. Ils dépendent de la loi de commande. Les couples requis pour les articulations sont calculés à base de la loi PD. Pour

la simulation, les trajectoires réelles du robot sont calculées en utilisant son modèle dynamique direct.

Dans le cas où l'état initial et l'état final sont définis dans l'espace opérationnel, le MGI est utilisé pour calculer les valeurs correspondantes dans l'espace articulaire :

$$\theta_d = MGI(X_d). \quad (\text{III.9})$$

En utilisant une loi de commande PD :

$$\Gamma = -k_p * e - k_v * \dot{e} \quad (\text{III.10-a})$$

avec  $\Gamma = [\tau_1 \ \tau_2 \ \tau_3]^T$  et  $e = [e_1 \ e_2 \ e_3]^T$ .

$K_p$  et  $K_v$  sont des matrices diagonales des paramètres PD.

On peut considérer le système comme trois boucles de commande indépendantes. Au niveau de chaque articulation, on a :

$$\tau_i = k_{p_i} * e_i + k_{v_i} * \dot{e}_i, \quad (i=1, \dots, 3) \quad (\text{III.10-b})$$

$k_{p_i}$  et  $k_{v_i}$  sont les éléments diagonaux de  $K_p$  et  $K_v$ .

Dans la majorité des applications industrielles, le robot est considéré comme trois articulations indépendantes. Chacune est décrite par un modèle très simple :

$$\tau_i = \ddot{\theta}_i \quad (i=1, \dots, 3) \quad (\text{III.11})$$

En remplaçant (III.10-b) dans (III.11), on obtient l'équation dynamique du système en boucle fermée :

$$\ddot{\theta}_i + k_{v_i} * \dot{\theta}_i(t) + k_{p_i} * \theta_i(t) = 0 \quad (i=1, \dots, 3)$$

Pour assurer une réponse stable sans oscillations, on utilise un amortissement critique ( $\xi=1$ ) avec une pulsation  $\omega_n=25$  rad/s. Pour ce faire, on fixe les gains :

$$k_p = \omega_n^2 = 625 \quad \text{et} \quad k_v = 2 * \omega_n = 50.$$

Ces valeurs permettent de générer les couples nécessaires pour chaque articulation à partir de l'erreur de position et de vitesse. La pulsation est choisie suffisamment basse pour éviter les effets de résonance mécanique.

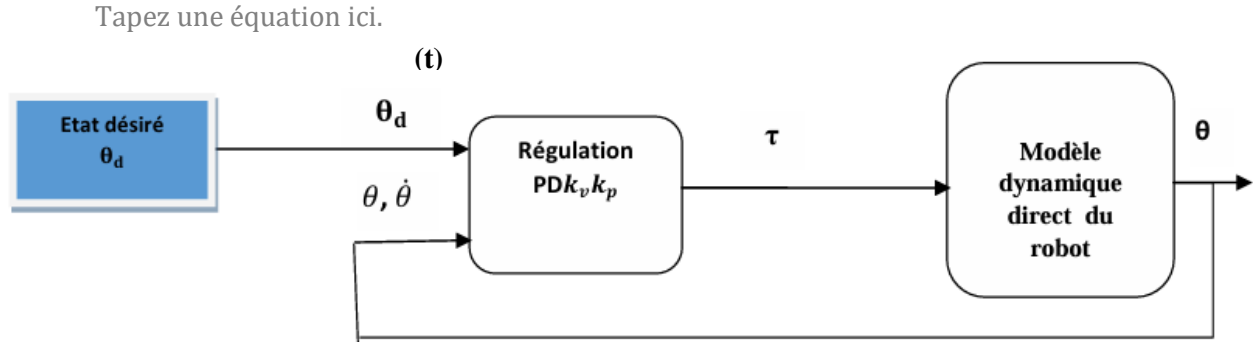


Figure III.14– Schéma de simulation de la commande PD du bras de robot

### III.7.1.2 Commande en poursuite de trajectoire

Dans le cas où le système est appelé à poursuivre une trajectoire  $(\theta_d(t), \dot{\theta}_d(t), \ddot{\theta}_d(t))$ , générée par un module de génération de trajectoires (MGT), on définit l'erreur de poursuite :

$$e(t) = (\theta_d(t) - \theta(t)) \quad \text{et} \quad \dot{e}(t) = (\dot{\theta}_d(t) - \dot{\theta}(t))$$

On exprime le couple de commande sous la forme suivante :

$$\tau = \ddot{\theta}_d(t) + k_p * e + k_v * \dot{e} \quad (\text{III.12})$$

En remplaçant cette loi de commande PD avec un terme d'anticipation dans le modèle  $\tau_i = \ddot{\theta}_i$ , on obtient l'équation dynamique de l'erreur:

$$\ddot{e} + k_v * \dot{e} + k_p * e = 0 \quad (\text{III.13})$$

Pour  $\zeta=1$  et en choisissant une pulsation naturelle  $\omega_n$ , les gains de la commande peuvent être définis comme suit :

$$k_v = 2 \omega_n \quad \text{et} \quad k_p = \omega_n^2$$

## III.8 Planification de trajectoires d'un bras robotisé

La planification de trajectoires a pour objectif de déterminer le mouvement que le robot doit effectuer pour atteindre une cible ou accomplir une tâche, tout en respectant ses limitations mécaniques et dynamiques. Elle peut être effectuée soit dans l'espace articulaire, soit dans l'espace cartésien (ou opérationnel).

Dans l'espace articulaire, la trajectoire décrit l'évolution des angles, des vitesses et des accélérations des articulations du robot en fonction du temps. Elle est généralement représentée par une série de points calculés à intervalles réguliers, avec une fréquence de mise à jour comprise entre 60 et 2000 Hz dans le cas des bras manipulateurs.

En revanche, dans l'espace opérationnel, la trajectoire décrit le mouvement de l'outil (ou effecteur) exprimé en termes de position et d'orientation par rapport au repère fixe de la station de travail. Générer un tel mouvement consiste à déterminer les consignes de position, de vitesse et d'accélération nécessaires pour que le robot suive la trajectoire définie.

### III.8.1 Méthodes de planification dans l'espace articulaire

Dans ce type de planification, chaque point de la trajectoire correspond à un angle spécifique que chaque articulation doit atteindre. Pour assurer un mouvement fluide et sans à-coups, une fonction continue est déterminée pour chaque articulation. Cette fonction relie les différents points intermédiaires et aboutit à la position finale, tout en minimisant les vibrations potentielles.

Afin de synchroniser le mouvement, la durée exigée à chaque segment de la trajectoire est identique pour toutes les articulations. Ainsi, chaque articulation atteint simultanément les points de passage. Plusieurs types de fonctions peuvent être utilisées pour relier ces points, mais leur choix doit respecter un ensemble de contraintes définies par la dynamique ou la mécanique du robot.

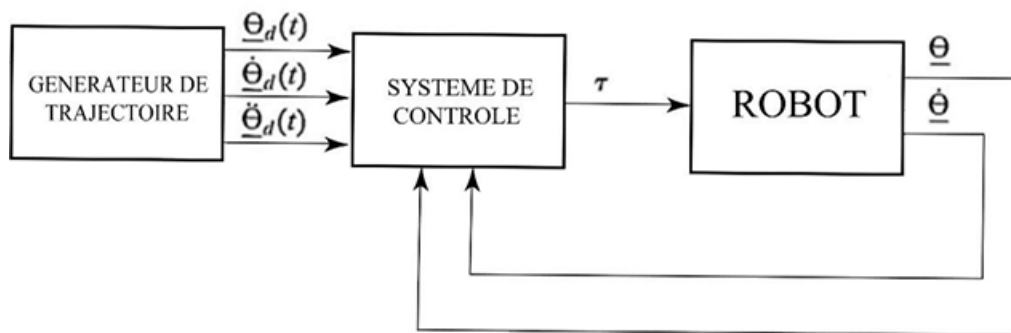


Figure III.9– Génération de mouvement dans l'espace articulaire

#### III.8.1.1 Méthode du polynôme Cubique



Elle permet de créer des trajectoires continues et régulières, aussi bien en position qu'en vitesse. Cela est essentiel pour assurer un mouvement fluide, sans interruptions ni vibrations.

On peut distinguer deux classes de mouvements :

- Mouvement entre deux points avec une trajectoire libre entre les deux extrémités,
- Mouvement passant par des points intermédiaires (points via).

#### 1-Mouvement entre deux points :

Dans le cas de deux points (initial-final), on peut imposer les quatre contraintes suivantes :

Contraintes sur les valeurs des fonctions :

$$\theta(0) = \theta_0, \quad \theta(t_f) = \theta_f \quad (\text{III.14})$$

Contraintes sur la continuité de la vitesse :

$$\dot{\theta}(0) = 0, \quad \dot{\theta}(t_f) = 0 \quad (\text{III.15})$$

Pour satisfaire ces contraintes, il faut un polynôme de degré  $\geq 3$ ; d'où l'usage de polynômes cubiques (de degré 3) ayant quatre paramètres :

$$\theta(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3$$

Avec

$$\dot{\theta}(t) = a_1 + 2 a_2 t + 3 a_3 t^2$$

$$\ddot{\theta}(t) = 2 a_2 + 6 a_3 t \quad (\text{III.16})$$

Les contraintes (1) et (2) permettent d'écrire :

$$\begin{aligned} \theta_0 &= a_0, & \theta_f &= a_0 + a_1 t_f + a_2 t_f^2 + a_3 t_f^3 \\ 0 &= a_1, & 0 &= a_1 + 2 a_2 t_f + 3 a_3 t_f^2 \end{aligned} \quad (\text{III.17})$$

Ce qui donne :

$$\begin{aligned} a_0 &= \theta_0 & a_2 &= \frac{3}{t_f^2} (\theta_f - \theta_0) \\ a_1 &= 0 & a_3 &= -\frac{3}{t_f^2} (\theta_f - \theta_0) \end{aligned} \quad (\text{III.18})$$

Cette solution ne concerne que le cas où le bras démarre et finit à vitesse nulle [14].

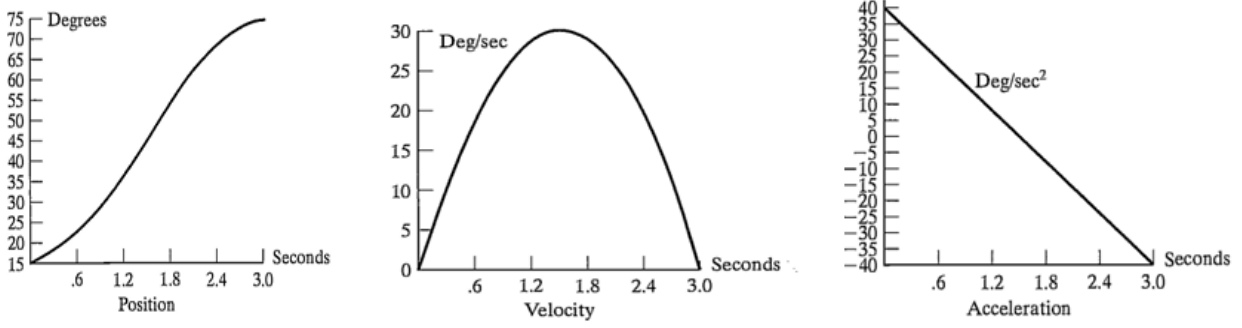


Figure III.10– Allures de la position, de la vitesse et de l'accélération pour  $\theta_0 = 15^\circ, \theta_f = 75^\circ$  et  $t_f = 3s$  [14].

## 2-Mouvement continu entre points via

On suit le même principe, mais en considérant des vitesses non nulles aux point via:

$$\dot{\theta}(0) = \dot{\theta}_0 \text{ et } \dot{\theta}(t_f) = \dot{\theta}_f \quad (\text{III.19})$$

On écrit alors :

$$\theta_0 = a_0, \quad \theta_f = a_0 + a_1 t_f + a_2 t_f^2 + a_3 t_f^3$$

$$\dot{\theta}_0 = a_1, \quad \dot{\theta}_f = a_1 + 2 a_2 t_f + 3 a_3 t_f^2 \quad (\text{III.20})$$

Soit :

$$a_0 = \theta_0 \quad a_2 = \frac{3}{t_f^2} (\theta_f - \theta_0) - \frac{2}{t_f} \dot{\theta}_0 - \frac{1}{t_f} \dot{\theta}_f$$

$$a_1 = \dot{\theta}_0 \quad a_3 = -\frac{3}{t_f^2} (\theta_f - \theta_0) + \frac{1}{t_f^2} (\dot{\theta}_0 + \dot{\theta}_f) \quad (\text{III.21})$$

De cette façon, on calcule le polynôme cubique entre deux points avec des vitesses initiales et finales quelconques [14].

### III.8.1.2 Méthode de polynôme de degré cinq

Génère des trajectoires lisses en contrôlant position, vitesse et accélération, avec six contraintes à satisfaire pour assurer un démarrage et un arrêt en douceur, idéal pour les mouvements précis des bras robotisés, à l'aide d'un polynôme de degré cinq :

$$\theta(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5 \quad (\text{III.22})$$

Les contraintes:

$$\begin{aligned}\theta_0 &= a_0; & \theta_f &= a_0 + a_1 t_f + a_2 t_f^2 + a_3 t_f^3 + a_4 t_f^4 + a_5 t_f^5 \\ \dot{\theta}_0 &= a_1; & \dot{\theta}_f &= a_1 + 2a_2 t_f + 3a_3 t_f^2 + 4a_4 t_f^3 + 5a_5 t_f^4 \\ \ddot{\theta}_0 &= 2a_2; & \ddot{\theta}_f &= 2a_2 + 6a_3 t_f + 12a_4 t_f^2 + 20a_5 t_f^3\end{aligned}\quad (\text{III.23})$$

Les paramètres :

$$a_0 = \theta_0 ; \quad a_1 = \dot{\theta}_0 ; \quad a_2 = \frac{\ddot{\theta}_0}{2} \quad (\text{III.24})$$

$$\begin{aligned}a_3 &= \frac{20\theta_f - 20\theta_0 - (80\dot{\theta}_f + 12\dot{\theta}_0)t_f - (3\ddot{\theta}_0 - \ddot{\theta}_f)t_f^2}{2t_f^3} ; & a_4 &= \frac{30\theta_0 - 30\theta_f + (14\dot{\theta}_f + 16\dot{\theta}_0)t_f + (3\ddot{\theta}_0 - 2\ddot{\theta}_f)t_f^2}{2t_f^4} \\ a_5 &= \frac{12\theta_f - 12\theta_0 - (6\dot{\theta}_f + 6\dot{\theta}_0)t_f - (\ddot{\theta}_0 - \ddot{\theta}_f)t_f^2}{2t_f^5}\end{aligned}\quad (\text{III.25})$$

#### III.8.2 Planification de trajectoire pour le bras robotique PUMA 560

Le robot PUMA dispose initialement de 6 degrés de liberté. Pour simplifier l'analyse tout en conservant une dynamique représentative, seuls trois axes, correspondant aux articulations principales du bras, ont été considérés.

Le système est ainsi modélisé comme un robot à 3 DDL, soumis à une planification de trajectoires et à une commande de type PD. Son comportement est régi par l'équation dynamique suivante [13] :

$$\Gamma = M(\theta)\ddot{\theta} + V(\theta, \dot{\theta})\dot{\theta} + G(\theta) \quad (\text{III.26})$$

où :

- $M(\theta)$  : matrice d'inertie dépendante de la configuration articulaire.
- $V(\theta, \dot{\theta})$  : vecteur regroupant les effets de Coriolis et les forces centrifuges.
- $G(\theta)$  : vecteur des forces gravitationnelles.
- $\Gamma$  : vecteur des couples moteurs appliqués.

Les trajectoires articulaires ont été générées à l'aide d'un polynôme cubique, dont les paramètres ont été calculés et tracés à l'aide de fonctions MATLAB (voir annexe A.1). Afin d'assurer une simulation réaliste, la modélisation intègre la dynamique non linéaire du robot, notamment à travers les expressions analytiques de la matrice d'inertie et du vecteur gravitationnel, fournies en annexe A.3.

### III.9 Simulation

Dans cette section, Nous présentons une étude par simulation pour la planification des trajectoires et la commande PD en poursuite pour les 3 premières articulations du robot. Quant aux 3 articulations du poignet, elles seront maintenues à une position fixe, définie par :

$$\theta_4 = 0, \theta_5 = 0 \text{ et } \theta_6 = 0.$$

TableauIII.2 : Mouvement des articulations entre 2 points

	Valeur initiale $\theta_i$	Valeur finale $\theta_f$
1 <sup>ère</sup> articulation $\theta_1$	0°	45°
2 <sup>ème</sup> articulation $\theta_2$	80°	45°
3 <sup>ème</sup> articulation $\theta_3$	0°	60°

Pour assurer le suivi précis de la trajectoire de bras, 3 fonctions principales ont été développées : fcn(MGT), PD\_1 et Dynamic\_model.

- **Fcn**: génère une trajectoire lisse pour chaque articulation à l'aide d'un polynôme cubique, garantissant des vitesses nulles aux points de départ et d'arrivée. Elle fournit à chaque instant les valeurs de position ( $\theta$ ), la vitesse ( $\dot{\theta}$ ) et l'accélération ( $\ddot{\theta}$ ) souhaitées, servant de références pour la commande (voir annexe A.1).
- **PD\_1**: implémente une commande proportionnelle-dérivée enrichie d'un terme anticipatif (feedforward) basé sur l'accélération désirée. Elle calcule le couple moteur nécessaire pour corriger les écarts en position et vitesse, tout en améliorant la stabilité et la réactivité du système (voir annexe A.2).
- **DYNAMIC\_MODEL**: simule la dynamique du robot à trois articulations en utilisant les équations de la dynamique directe. Elle déduit les accélérations articulaires en fonction des couples appliqués, tout en tenant compte de l'inertie, des effets de Coriolis et de la gravité, ce qui permet d'anticiper l'évolution du système (voir annexe A.4).

#### Interprétation des Résultats :

Les courbes illustrant l'évolution des positions et des vitesses des articulations obtenues lors d'une poursuite de trajectoire à l'aide d'un régulateur PD. Les figures III.11 et III.12 montrent que la position et la vitesse réelles de la 1<sup>ère</sup> articulation suivent globalement les valeurs désirées,

ce qui indique que le régulateur PD assure un suivi efficace de la trajectoire avec une légère différence initiale.

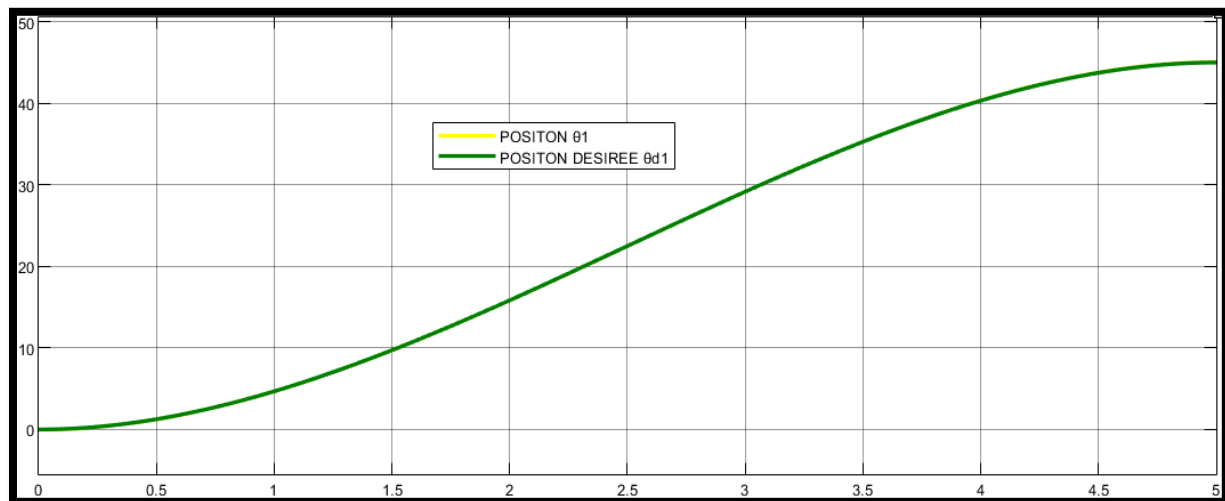


Figure III.11–L'évolution des positions de 1<sup>ère</sup> articulation

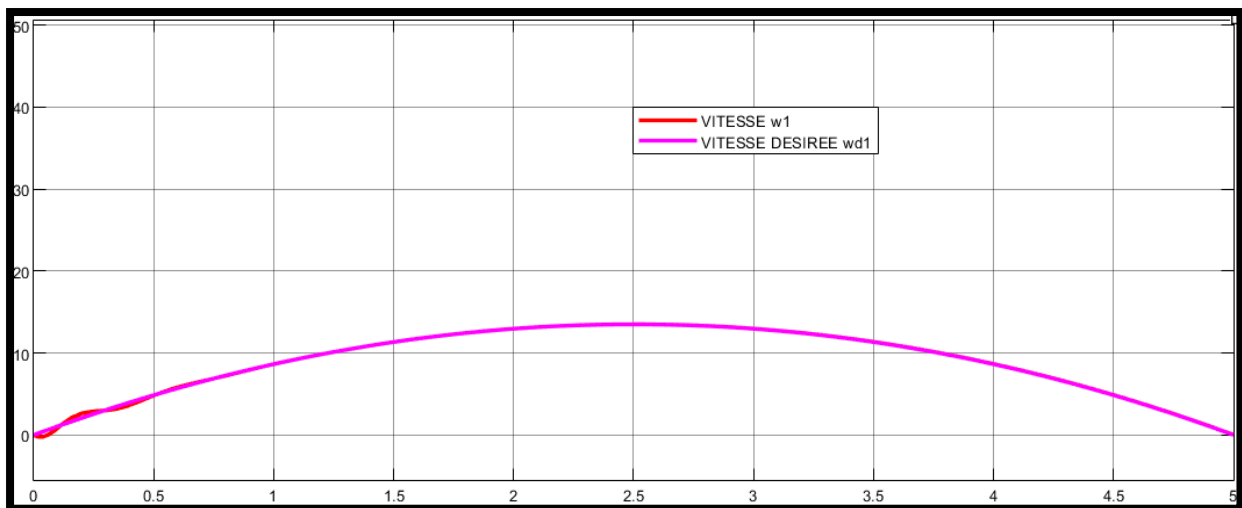


Figure III.12–L'évolution des vitesses de 1<sup>ère</sup> articulation

- La figure III.13 montre que l'accélération suit une décroissance linéaire passant d'une valeur positive à une valeur négative, ce qui reflète un ralentissement progressif du mouvement. Cette évolution assure une transition fluide, limitant les efforts mécaniques sur l'articulation.

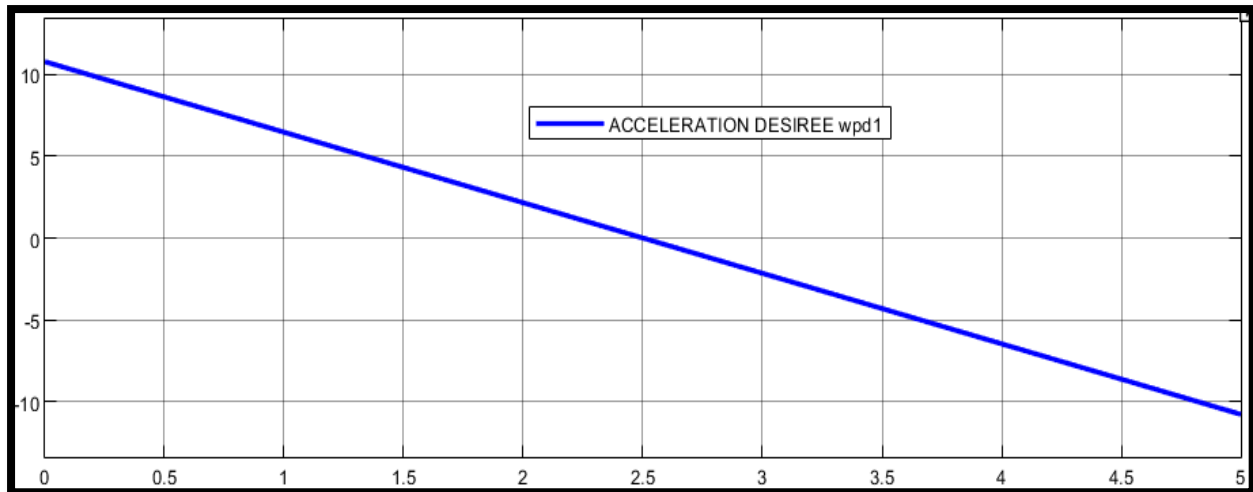


Figure III.13–L'évolution de l'accélération de 1<sup>ère</sup> articulation

- Les figures III.14 et III.15 reflètent une bonne performance du régulateur PD qui permet à l'articulation 2 du bras de suivre correctement la trajectoire imposée, avec une concordance satisfaisante entre les valeurs réelles et désirées malgré un léger retard dans la réponse de vitesse.

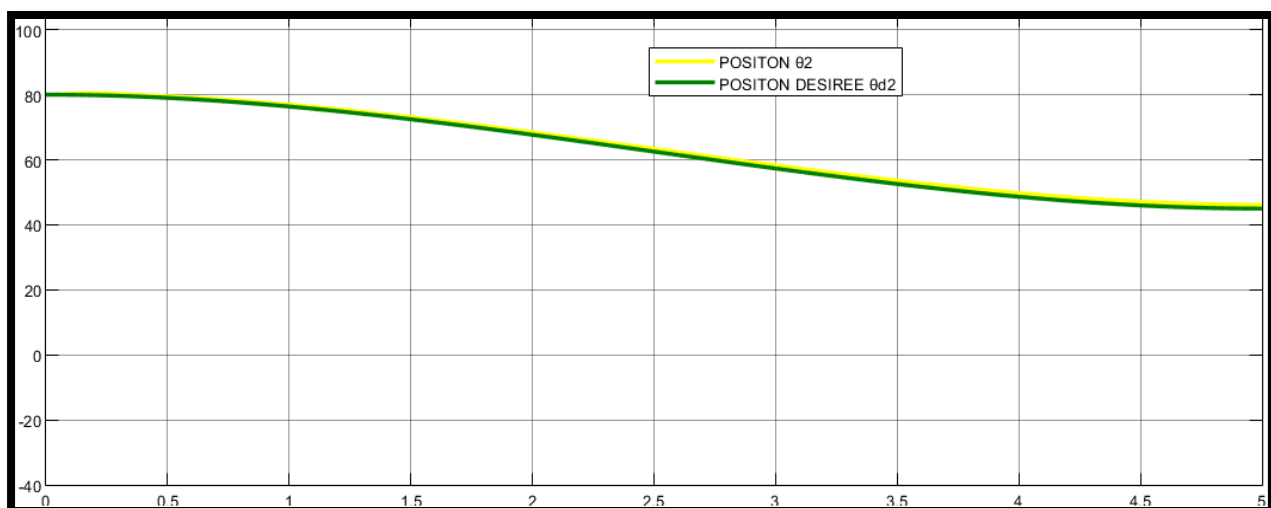


Figure III.14–L'évolution des positions de 2<sup>ème</sup> articulation

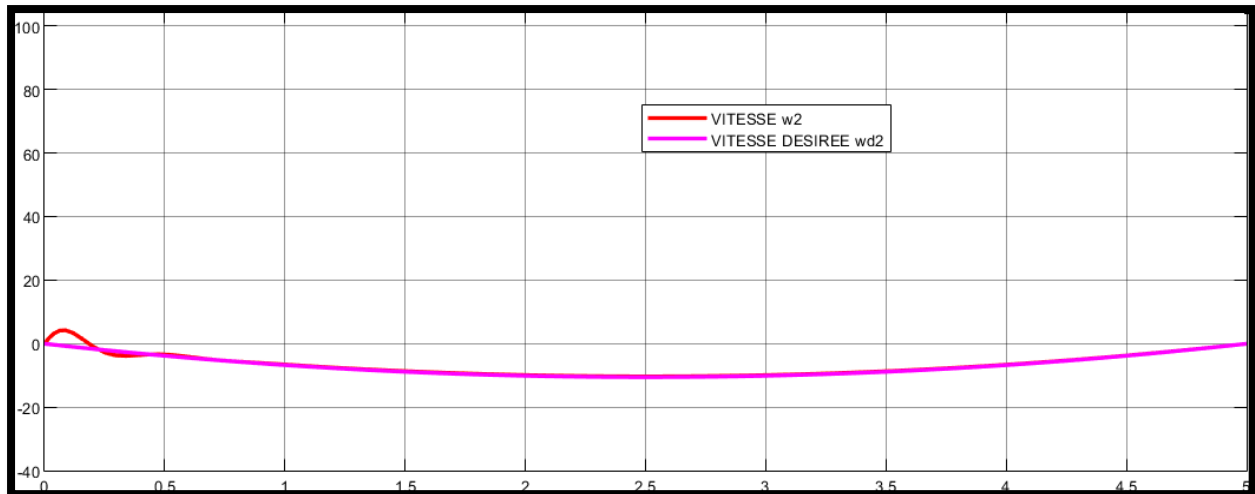


Figure III.15–L'évolution des vitesses de 2<sup>ème</sup> articulation

- La figure III.16 montre une variation de l'accélération désirée, signe d'un comportement dynamique avec des ajustements continus. Cette fluctuation traduit une réponse active aux modifications de trajectoire.

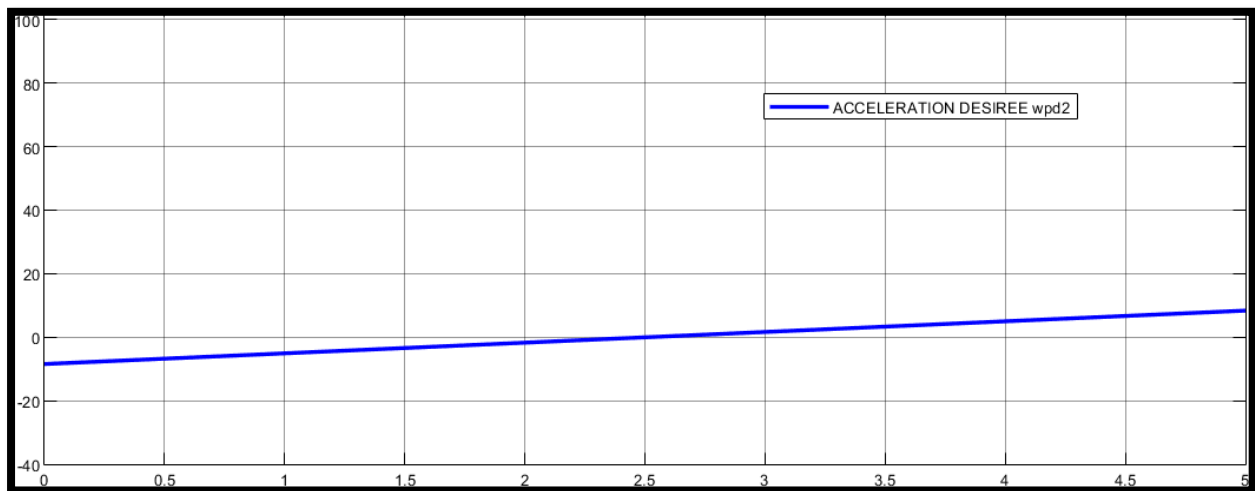


Figure III.16–L'évolution de l'accélération de 2<sup>ème</sup> articulation

- Les figures III.17 et III.18 mettent en évidence un suivi globalement efficace de la trajectoire pour la 3<sup>ème</sup> articulation, avec une bonne cohérence entre les courbes réelles et désirées malgré un léger décalage initial.

Le système de commande semble bien maîtriser l'évolution des positions et des vitesses.

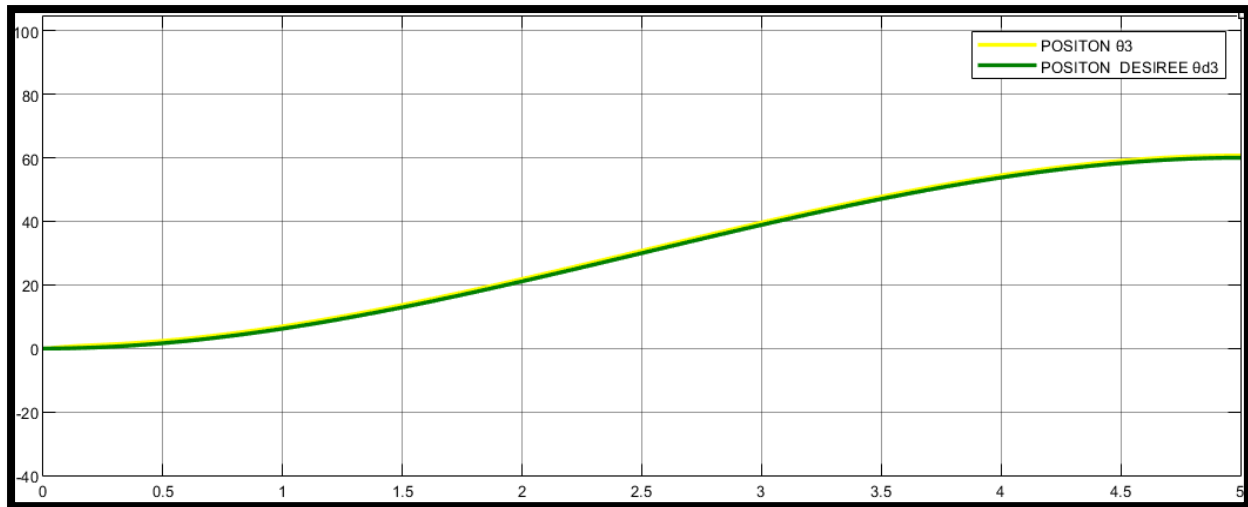


Figure III.17–L'évolution des positions de 3<sup>ème</sup> articulation

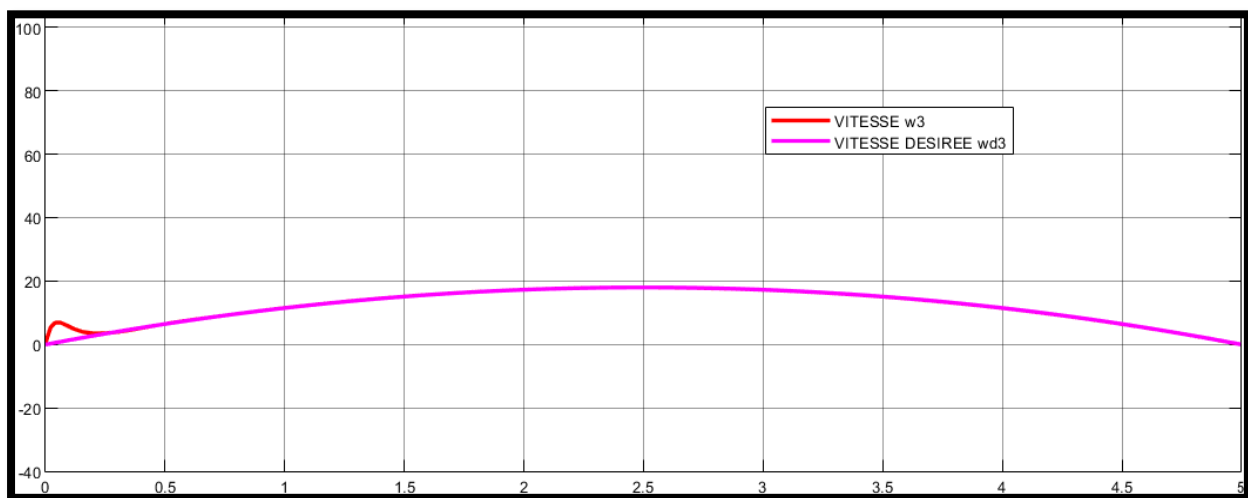


Figure III.18–L'évolution des vitesses de 3<sup>ème</sup> articulation

- La figure III.19 montre que l'accélération diminue graduellement au fil du temps. Ce changement indique une phase de ralentissement maîtrisé, conçue pour garantir une transition fluide tout en évitant les contraintes mécaniques excessives.



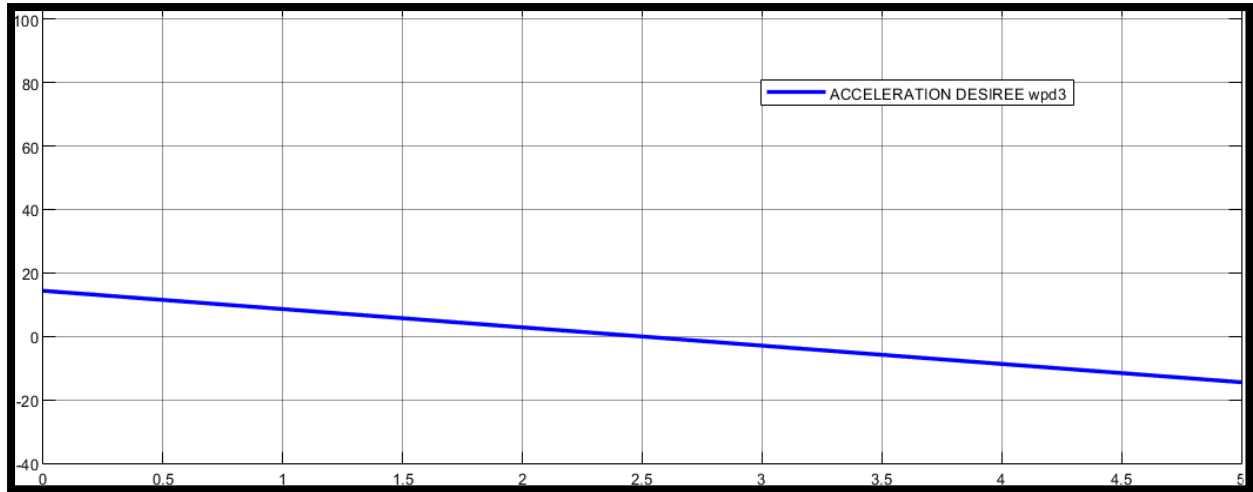


Figure III.19–L'évolution de l'accélération de 3<sup>ème</sup> articulation

### III.10 Conclusion

Ce chapitre est consacré à l'étude de la modélisation, de la commande et de la planification de trajectoires d'un bras robotisé. Trois types de modèles y ont été développés : le modèle géométrique, le modèle cinématique et le modèle dynamique, chacun présenté dans ses versions directe et inverse. Par la suite, nous avons implémenté quelques techniques de commande des bras de robot. Puis nous avons mentionné l'utilisation de la méthode du polynôme cubique pour planifier les trajectoires dans l'espace articulaire et traité la planification des trajectoires d'un robot à trois degrés de liberté, et on a terminé par la simulation des résultats.

# **Chapitre IV : Implémentation de la commande sous TIA PORTAL**

### **IV.1 Introduction**

Ce chapitre présente l'automatisation d'un bras palettiseur à trois degrés de liberté, contrôlé par un API Siemens S7-1200 et programmé avec TIA Portal V15.1. Il commence par l'exposé du cahier des charges, puis la présentation de l'architecture et de la régulation du système, la programmation des blocs, le contrôle des servomoteurs, les résultats de simulation et l'interface de supervision.

### **IV.2 Cahier des charges**

Le robot commence en position initiale : base orientée à  $0^\circ$ , épaule inclinée à  $80^\circ$ , coude (bras) à  $0^\circ$  et pince (tête) complètement fermée ( $0^\circ$ ).

Avant de lancer le cycle de mouvement, le système doit vérifier que 60 bouteilles sont prêtes à être manipulées et qu'une palette vide est disponible. La palette étant composée de 4 étages, chacun d'eux doit être rempli successivement avant que le robot se redémarre un nouveau cycle.

Une fois ces conditions vérifiées, le cycle commence lorsque le bouton Start est pressé. IL se déroule en deux phases principales :

#### **1-Prise des bouteilles :**

Le cycle débute par la rotation de la base à  $45^\circ$  vers la gauche afin d'aligner le bras avec les bouteilles. L'épaule se redresse ensuite à  $45^\circ$ , tandis que le coude s'abaisse de  $60^\circ$  pour rapprocher la pince des bouteilles. La pince s'ouvre complètement ( $90^\circ$ ) pour saisir la charge, puis se referme afin de maintenir les bouteilles.

#### **2-Dépose des bouteilles sur la palette**

L'épaule et le coude remontent vers leur position initiale, tandis que la base effectue un retour à la position  $0^\circ$ . L'épaule et le coude redescendent ensuite, et la pince s'ouvre pour déposer les bouteilles sur la palette.

Pour des raisons de sécurité, un arrêt d'urgence est programmé pour se déclencher automatiquement en cas de situation anormale, telle qu'une surcharge de palettes ou de bouteilles.

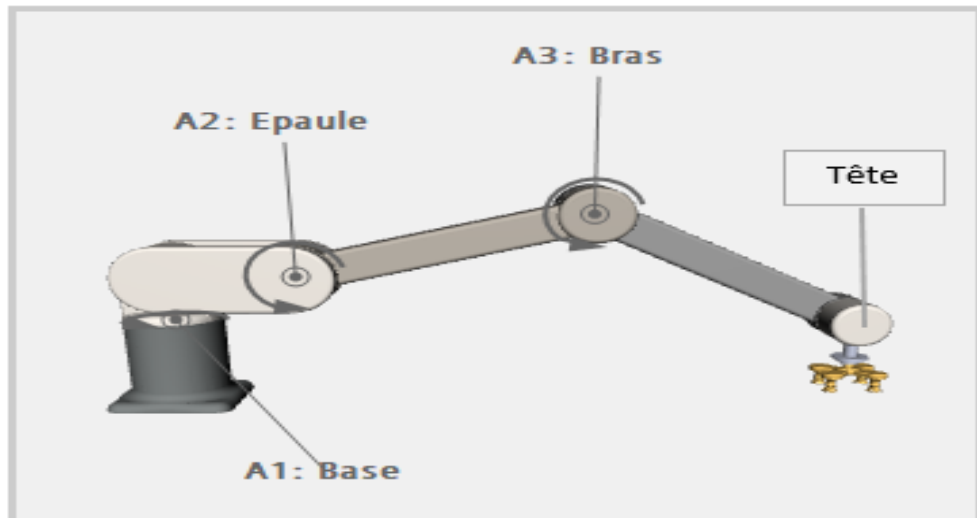


Figure IV.1– Bras de Robot [17]

### IV.3 Architecture générale du système robotisé

Le système robotisé industriel se compose de deux blocs principaux : la partie commande et la partie opérative, reliés par des interfaces d'entrées/sorties pour la transmission des signaux.

**1-Partie commande** : C'est un ensemble d'éléments comprenant l'API, l'IHM permettant à l'utilisateur de programmer les tâches, ainsi que les modules d'E/S.

-API : gère le traitement, l'exécution et l'envoi des ordres dans le système de commande.

-IHM : à travers laquelle l'utilisateur programme les tâches que le robot doit exécuter [9].

-E/S (Entrée/Sortie) : permettent la communication entre le système et les équipements externes.

**2-Partie opérative** : comprend les composants exécutant les tâches : actionneurs, système mécanique et capteurs, permettant les mouvements et l'ajustement du robot.

-Pré-actionneur (variateur de vitesse) : transformation de l'énergie électrique en mouvement.

-Actionneurs (servomoteur) : production d'effort.

-Système mécanique articulé : exécute les tâches.

-Capteurs (codeurs incrémentaux) : renvoient l'état des articulations [17].

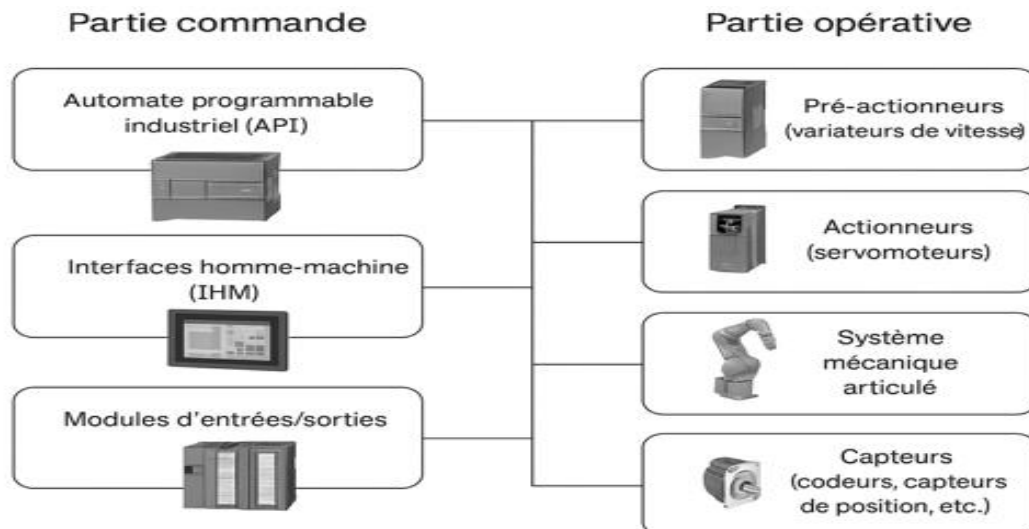


Figure IV.2– Architecture générale d'un système Robotisé

### IV.4 Boucle de régulation

La régulation des mouvements du bras robotisé est assurée par une boucle fermée, où les capteurs mesurent en temps réel la position et la vitesse des articulations. En fonction de la consigne donnée, la CPU calcule le profil de vitesse et ajuste les paramètres via le variateur, garantissant ainsi une vitesse optimale. Chaque articulation est contrôlée par deux boucles : l'une pour la position et l'autre pour la vitesse, assurant un suivi précis et efficace des mouvements.

-La boucle de position constitue la boucle principale du système. Elle inclut une CPU qui envoie les ordres au variateur, lequel contrôle le moteur grâce aux informations du codeur.

- La boucle de vitesse, qui est une boucle interne intégrée dans le même variateur, permet d'ajuster la vitesse du moteur en temps réel.



Figure IV.3– Boucle de régulation du système robotisé

### IV.5 Matériels utilisé pour la conception du Projet sous TIA Portal

#### IV.5.1 Automate programmable industriel (API)

Nous avons choisi l'API S7-1200 1212C DC/DC/DC avec le firmWare V4.2, idéale pour des automatismes flexibles et performants.

Il dispose de 8 entrées numériques et 6 sorties numériques et 2 entrées analogiques 0-10 V. La tension d'alimentation est de 20,4 à 28.8 V avec une mémoire de programmation de données de 75 ko et une interface de communication PROFINET [18].

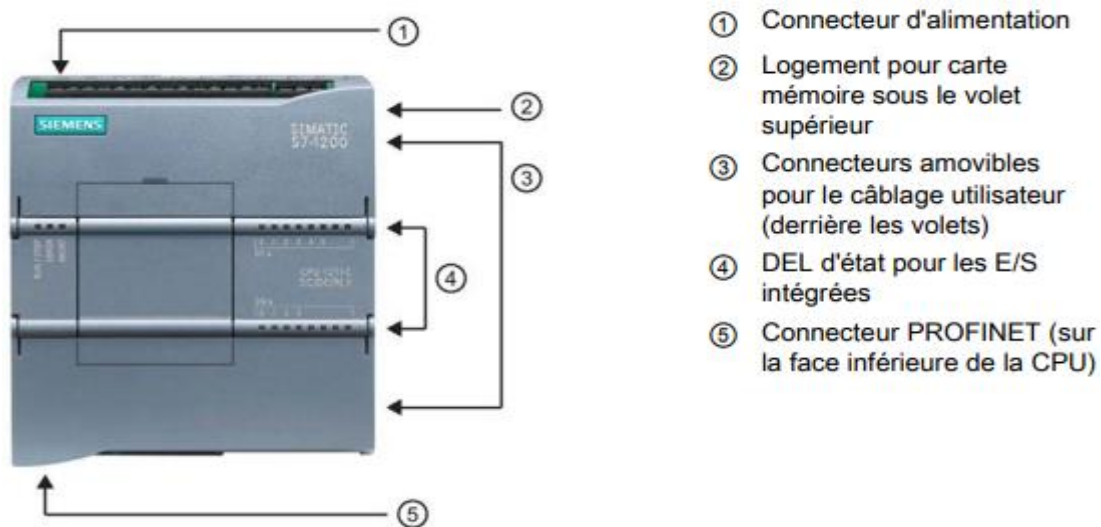


Figure IV.4– CPU S7 1200[18]

#### IV.5.2 Variateur de vitesse

Le variateur SINAMICS V90 a une entrée monophasée 200–240V a été choisi pour sa compatibilité avec l'alimentation disponible et sa facilité d'intégration via PROFINET dans TIA Portal, permettant un contrôle précis du servomoteur [19].



Figure IV.5– Variateur de vitesse SINAMICS V90.

### IV.5.3 Servomoteur

Nous avons opté le servomoteur SIMOTICS S-1FL6 pour sa compatibilité avec les exigences de notre système qui a une entrée monophasée 230V ; puissance 2000W ; 3000rpm ; 0,64Nm.

Il est équipé d'un codeur incrémental TTL avec une rotation de 2500 points/tour [19].



Figure IV.6– Servomoteur SIMOTICS S-1FL6

## IV.6 Conception du projet

Pour répondre aux exigences du cahier des charges et automatiser le contrôle du bras robotisé, le projet suit les étapes suivantes :

- 1-Création du projet dans TIA Portal.
- 2- Configuration du matériel, incluant le choix de la CPU et des variateurs Sinamics V90 (base, épaule, coude, pince) nécessaires pour l'application.
- 3- Configuration du réseau PROFINET pour assurer la communication.
- 4- Configuration des télégrammes de communication pour chaque variateur.
- 5- Création des blocs de programme de contrôle des servomoteurs.
- 6- Association des blocs de commande aux variateurs et réglage des paramètres de communication.

### IV.6.1 Création de projet

Nous présentons ci-après les deux étapes initiales de la création de notre projet sous TIA Portal.

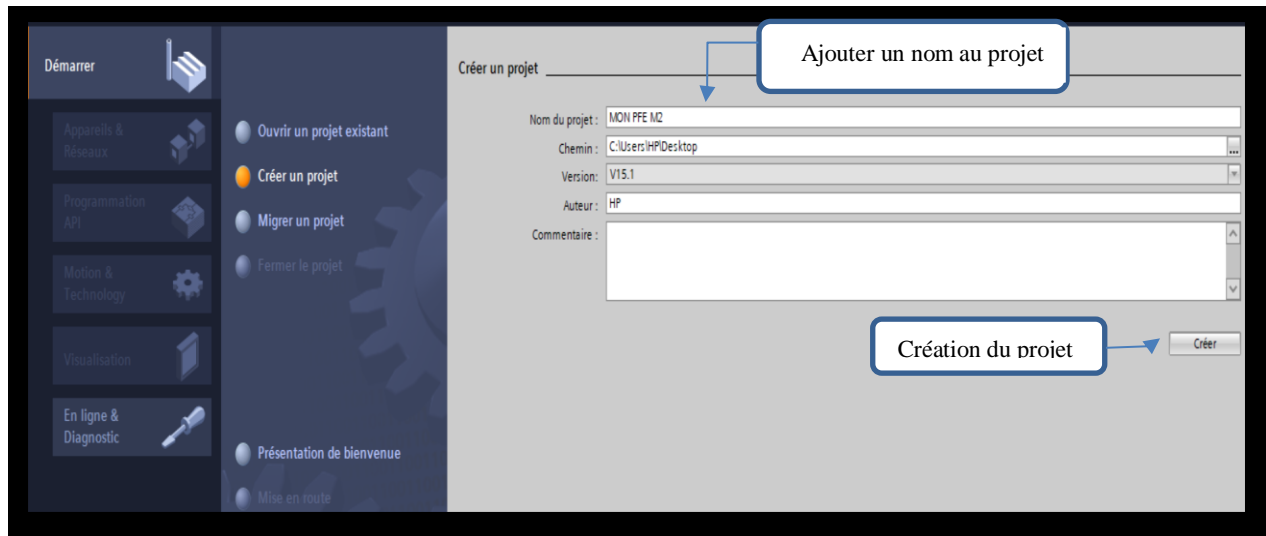


Figure IV.7– Création du projet TIA Portal

### IV.6.2 Configuration du matériel

Après la création du projet, nous sélectionnons une CPU 1212C DC/DC/DC et ajoutons les modules d'entrée/sortie, comme illustré ci-après.

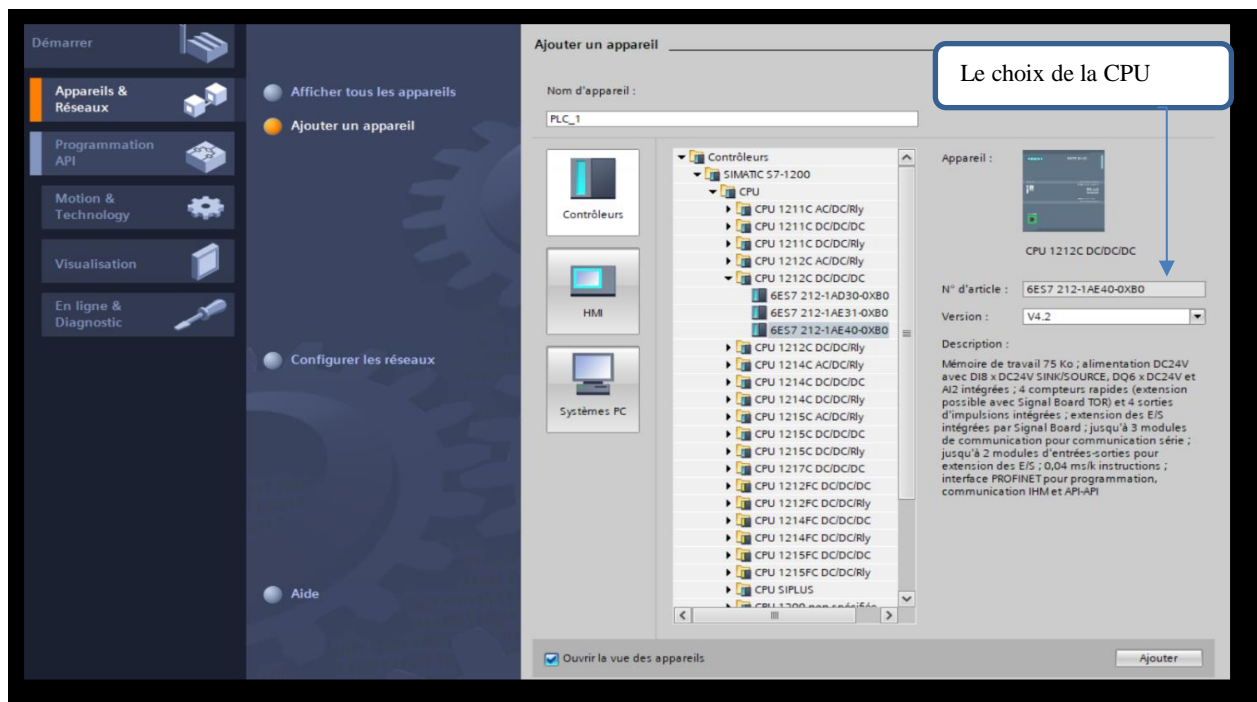


Figure IV.8–Le choix de la CPU



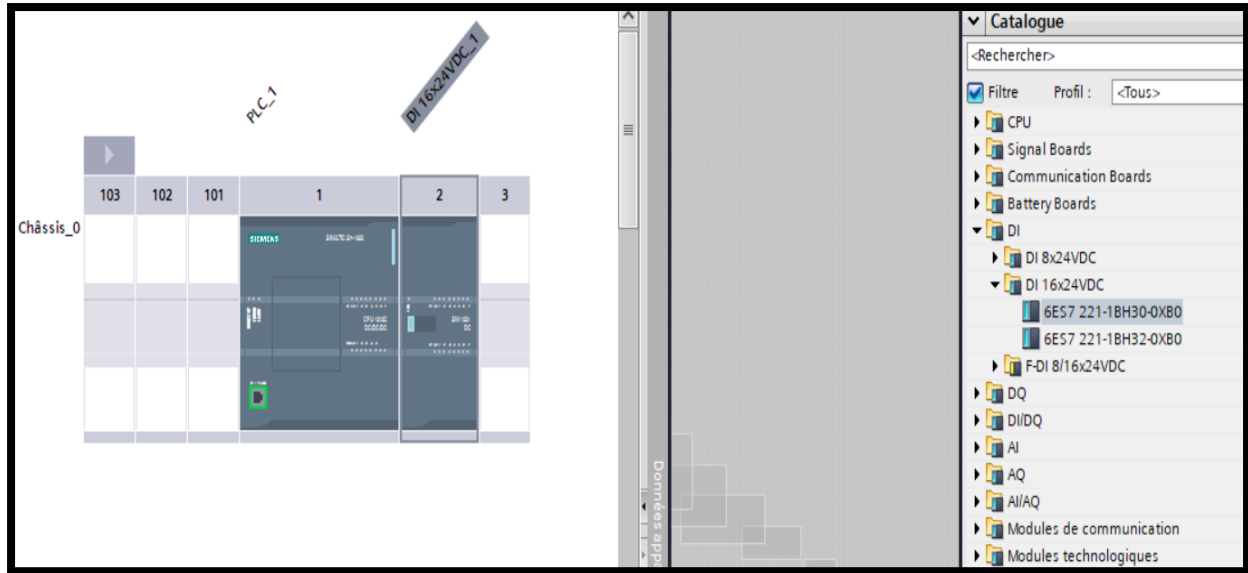


Figure IV.9– Configuration de la CPU

Après avoir configuré la CPU, nous avons inséré quatre variateurs SINAMICS V90 à partir de la section « Autres appareils de terrain », en sélectionnant ensuite l’option « Drives ».



Figure IV.10–Insertion des variateurs dans la vue réseau

### IV.6.3 Configuration du réseau de communication PROFINET

Chaque équipement PROFINET doit avoir une adresse IP pour communiquer sur le réseau. Après la configuration de la CPU, les connexions réseau sont définies dans « Appareils & Réseaux », puis le projet est chargé pour appliquer les adresses IP.

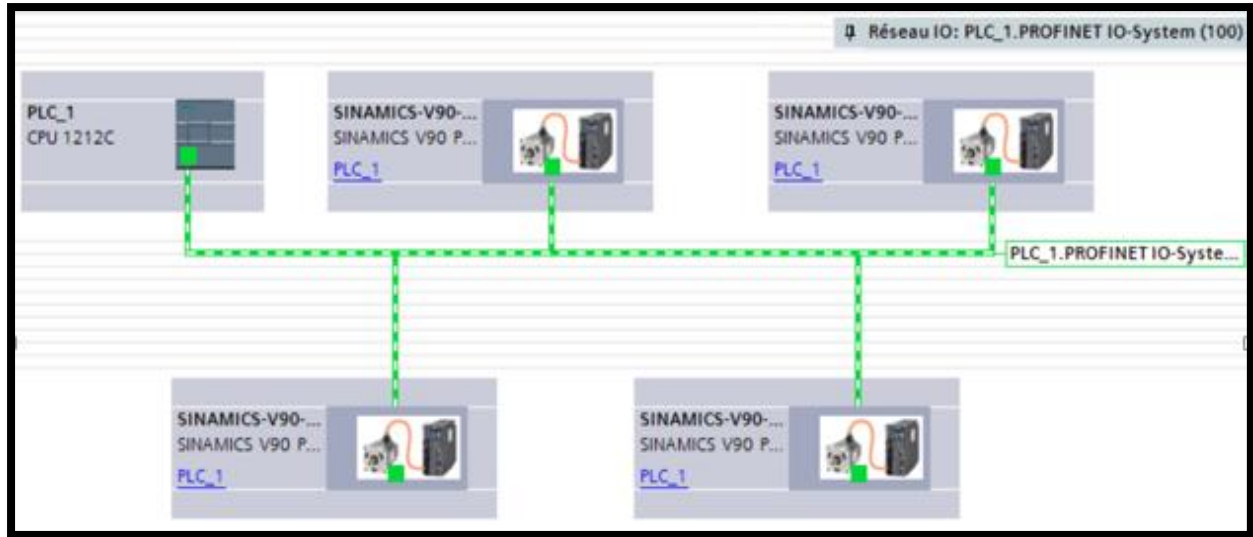


Figure IV.11-Configuration des liaisons de communication PROFINET

### IV.6.4 Configuration de télégramme de communication

Le SINAMICS V90 PN utilise des télégrammes standards et Siemens pour les modes de régulation de vitesse et de positionnement. Le télégramme est sélectionné via le paramètre P0922. Nous avons choisi le télégramme Standard 3, PZD-5/9 pour sa simplicité et son adéquation avec les besoins de communication entre le variateur et l'automate. (L'annexe B.1).

Module	Châssis	Empla..	Adresse I	Adresse...	Type	N° d'arti...
MOTEUR DE BASE	0	0			SINAMICS V90 PN V...	6SL3 21...
PN-IO	0	0 X150			SINAMICS-V90-PN	
Entraînement_1	0	1			Entraînement	
Module Access Point	0	1 1			Module Access Point	
sans PROFIsafe	0	1 2			sans PROFIsafe	
Télégramme standard 3, ...	0	1 3	68...85	64...73	Télégramme stand...	
	0	2				

Figure IV.12– Insertion du télégramme standard 3

## Chapitre IV : Implémentation de la commande sous TIA PORTAL

### Remarque

- **P0922** : est un paramètre qui nous permet de sélectionner le télégramme pour la communication PROFINET.
- **PZD** : est un mot qui est utilisé pour envoyer des ordres et consigne ou pour recevoir des signaux et mesures (PZD = ProZess Daten).
- **Le SINAMICS V90 PN** : est en mode régulation de vitesse (Annexe B.2) [17].

Tableau IV.1 : Mots de commande envoyés aux variateurs (adresse I)

			MOTEUR DE BASE	MOTEUR EPAULE	MOTEUR COUDE	MOTEUR PINCE
PZD1	STW1	Mot de commande	64.....65	74.....75	84.....85	94.....95
PZD2 PZD3	NSOLL_B	Consigne de vitesse	66.....69	76.....79	85.....89	96.....99
PZD4	STW2	2 <sup>eme</sup> Mot de commande	70.....71	80.....81	90.....91	100.....101
PZD5	G1_STW	Mot de commande du codeur	72.....73	82.....83	92.....93	102.....103

Tableau IV.2 : Mots d'état envoyés aux API (adresse Q)

			MOTEUR BASE	MOTEUR EPAULE	MOTEUR COUDE	MOTEUR PINCE
PZD1	ZSW1	Mot d'état principal	68.....69	86.....87	104.....105	122.....123
PZD2 PZD3	NIST_B	Valeur de vitesse	70.....73	88.....91	106.....109	124.....127
PZD4	ZSW2	2 <sup>eme</sup> Mot d'état	74.....75	92.....93	110.....111	128.....129
PZD5	G1_ZSW	Mot d'état du 1 <sup>er</sup> codeur	76.....77	94.....95	112.....113	130.....131
PZD6 PZD7	G1_XIST1	Position réelle du 1 <sup>er</sup> codeur	78.....81	96.....99	114.....117	132.....135
PZD8 PZD9	G1_XIST2	Position réelle du 2 <sup>eme</sup> codeur	82.....85	100.....103	118.....121	136.....139

### IV.6.5 Définition et organisation des variables

1-Les entrées de l'automate : collectent les données des capteurs et boutons, qui sont ensuite traitées pour générer les commandes adaptées et ajuster le processus.

2- Les sorties de l'automate : une fois les données d'entrée traitées, l'automate génère des signaux de sortie envoyés aux actionneurs pour exécuter les actions nécessaires sur l'installation.

3- Le memento : ou mémoire interne stocke des informations essentielles et intermédiaires pour le processus. La mémoire de la CPU permet l'accès en lecture et en écriture, utilisant des unités comme le bit, l'octet, le mot et le double mot.

4- L'adressage : Dans TIA Portal, les variables globales telles que les entrées, les sorties et les mémentos, sont attribuées à la fois une adresse symbolique et une adresse absolue. L'adresse absolue désigne l'identificateur d'opérande (I, Q, M) ainsi que son adresse et son numéro de bit, tandis que l'adresse symbolique représente le nom de la variable choisi par l'utilisateur (par exemple "Bouton marche"). [20]

5- Table des variables : est un tableau qui répertorie les variables utilisées dans le programme de l'automate, précisant leur type, adresse, rôle (entrée, sortie, memento) et description. Elle permet d'organiser les données de manière claire et structurée, facilitant leur gestion et leur accès tout au long de l'exécution du programme.

## Chapitre IV : Implémentation de la commande sous TIA PORTAL

Table de variables standard									
	Nom	Type de données	Adresse	Réma...	Acces...	Ecritu...	Visibl...	Comment	
1	START	Bool	%I0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
2	STOP	Bool	%I0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
3	STOP_URG	Bool	%M0.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
4	CAPTEUR_BOUT	Bool	%I0.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
5	NOMBRE_BOUT_DSR	Bool	%M2.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
6	NOMBRE_BOUT	Int	%MW50	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
7	CONVYR_BOUT	Bool	%Q0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
8	NOMBRE_ETG_DSR	Bool	%M4.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
9	NOMBRE_ETG	Int	%MW60	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
10	CAPTEUR_PALT	Bool	%I0.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
11	CONVYR_PALT	Bool	%Q0.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
12	ACTIVATION_SYS	Bool	%M1.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
13	SYS_ACTIVE	Bool	%Q0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
14	NMBR_BOUT	Bool	%M3.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
15	Tag_1	Bool	%M5.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
16	Tag_2	Bool	%M6.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
17	VAL_BRUTE_B	Int	%IW78	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
18	VAL_NORM_B	Real	%MD10	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
19	POSITION_BASE	Real	%MD14	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
20	PLG_BASE	Bool	%M7.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
21	VAL_BRUTE_E	Int	%IW96	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
22	VAL_NORM_E	Real	%MD18	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
23	POSITION_EPAULE	Real	%MD22	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
24	PLG_EPAULE	Bool	%M8.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
25	VAL_BRUTE_C	Int	%IW114	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
26	VAL_NORM_C	Real	%MD26	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
27	POSITION_COUDE	Real	%MD30	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
28	PLG_COUDE	Bool	%M9.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
29	VAL_BRUTE_P	Int	%IW132	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
30	VAL_NORM_P	Real	%MD34	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
31	POSITION_PINCE	Real	%MD38	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
32	PLG_PINCE	Bool	%M10.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
33	Tag_3	Bool	%M11.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
34	Tag_4	Bool	%M12.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
35	Tag_5	Bool	%M13.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
36	Tag_6	Bool	%M14.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
37	Tag_7	Bool	%M15.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
38	Tag_8	Bool	%M16.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
39	Tag_9	Bool	%M17.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
40	Tag_10	Bool	%M18.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
41	ROTATION_BASE	Word	%QW64	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
42	ROTATION_EPAULE	Word	%QW74	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
43	ROTATION_COUDE	Word	%QW84	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
44	ROTATION_PINCE	Word	%QW94	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
45	CONSIGNE_VIT_BASE	Word	%QW66	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
46	CONSIGNE_VIT_EPAULE	Word	%QW86	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
47	CONSIGNE_VIT_COUDE	Word	%QW76	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
48	CONSIGNE_VIT_PINCE	Word	%QW96	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
49	Tag_11	Bool	%M19.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
50	PINC	Bool	%Q0.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
51	CONSIGNE ANGLE ROT	Real	%MD100	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
52	CONSIGNE ANGLE EPAUL	Real	%MD104	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
53	CONSIGNE ANGLE COUDE	Real	%MD108	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
54	CONSIGNE ANGLE PINCE	Real	%MD112	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
55	Clock_1Hz	Bool	%M0.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
56	Tag_12	Word	%MW48	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
57	Tag_13	Word	%MW44	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
58	Tag_14	Time	%MD40	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		

Figure IV.13– Table des variables

### IV.6.6 Conception des blocs principaux et traitement des conditions

Le bloc OB1, point d'entrée du programme, exécute la logique principale, vérifie les conditions et gère les séquences de démarrage, d'arrêt et le suivi des capteurs. Cette logique a été développée en utilisant le langage Ladder.

#### 1-Démarrage et activation du processus :

Le système s'active via le bouton START (%I0.0) et peut se désactiver soit manuellement via le bouton STOP (%I0.1), soit automatiquement via l'arrêt d'urgence (%M0.2).

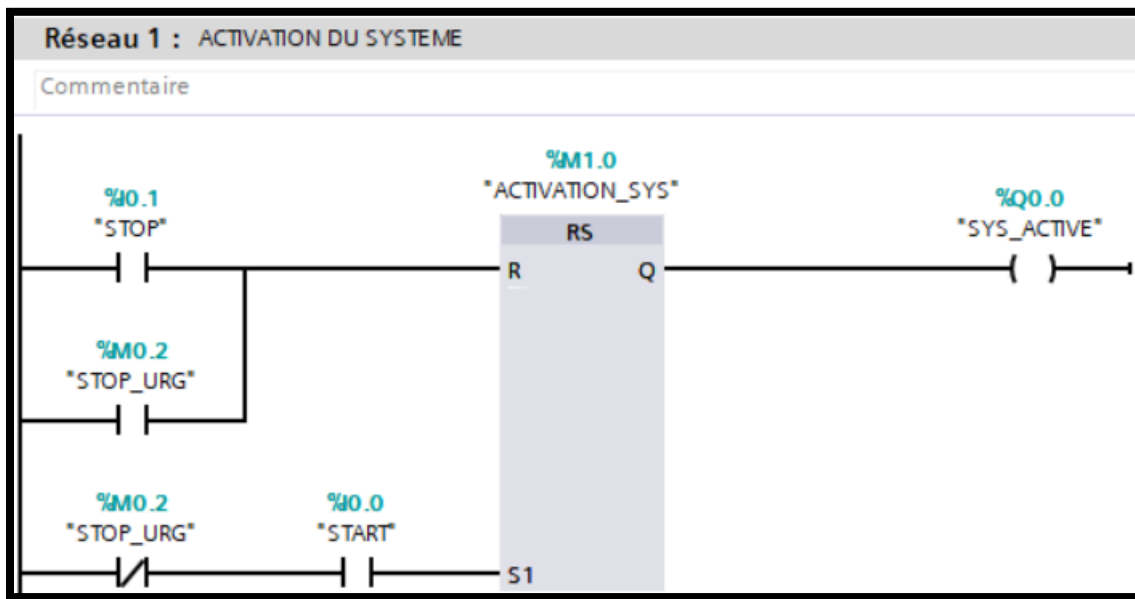


Figure IV.14–Logique d'activation du système

L'arrêt d'urgence est déclenché automatiquement en cas de situation anormale, notamment lorsque le nombre de bouteilles dépasse 60 ou le nombre d'étages est supérieur à 4.

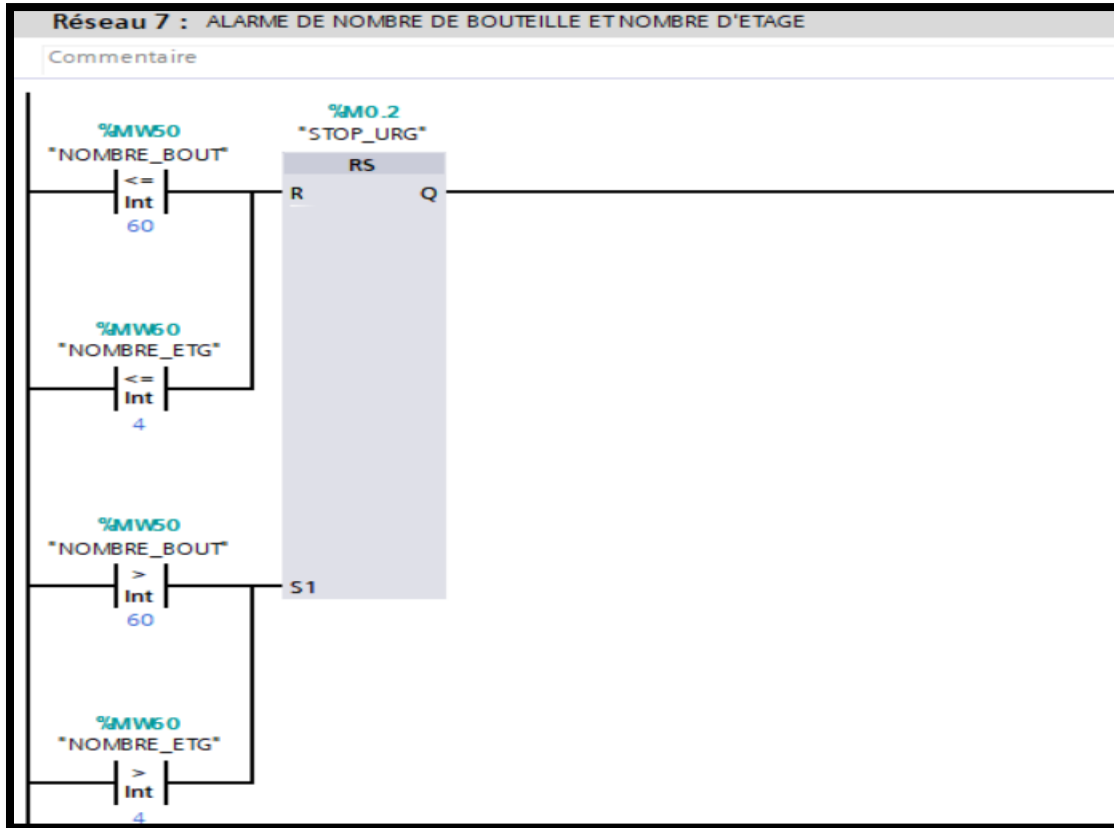
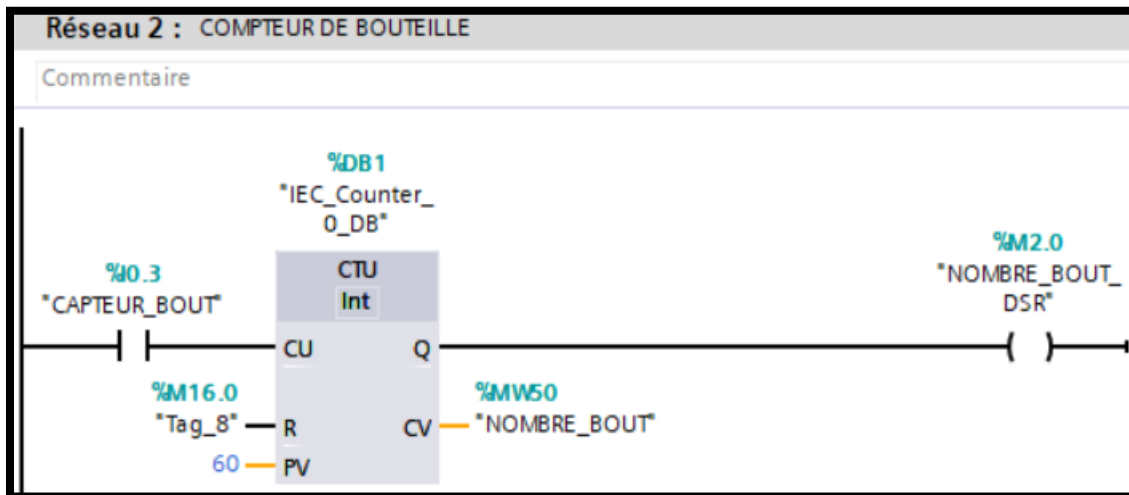


Figure IV.15-Logique de gestion de l'arrêt d'urgence

### 2-Vérification du nombre de bouteilles :

Le réseau de la figure IV.16 assure le comptage des bouteilles à l'aide du capteur %I0.3, jusqu'à atteindre le nombre souhaité(60), défini par le bit mémoire %M2.0.



IV.16- Programme Ladder de compteur de bouteille

## Chapitre IV : Implémentation de la commande sous TIA PORTAL

Pour la réservation de nombre de bouteille, la bascule RS s'active quand le nombre de bouteilles est atteint et se réinitialise à l'ouverture de la pince.

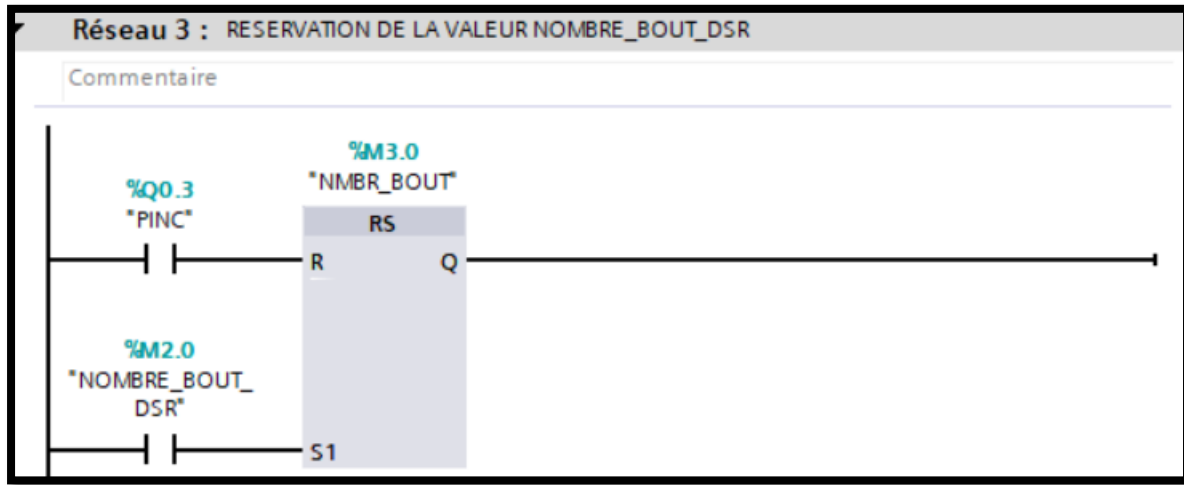


Figure IV.17 Programme Ladder de la réservation de nombre de bouteille désiré

Le réseau de la figure IV.18 assure la marche et l'arrêt du convoyeur de bouteille, en l'arrêtant automatiquement dès que le total de 60 bouteilles est atteint.

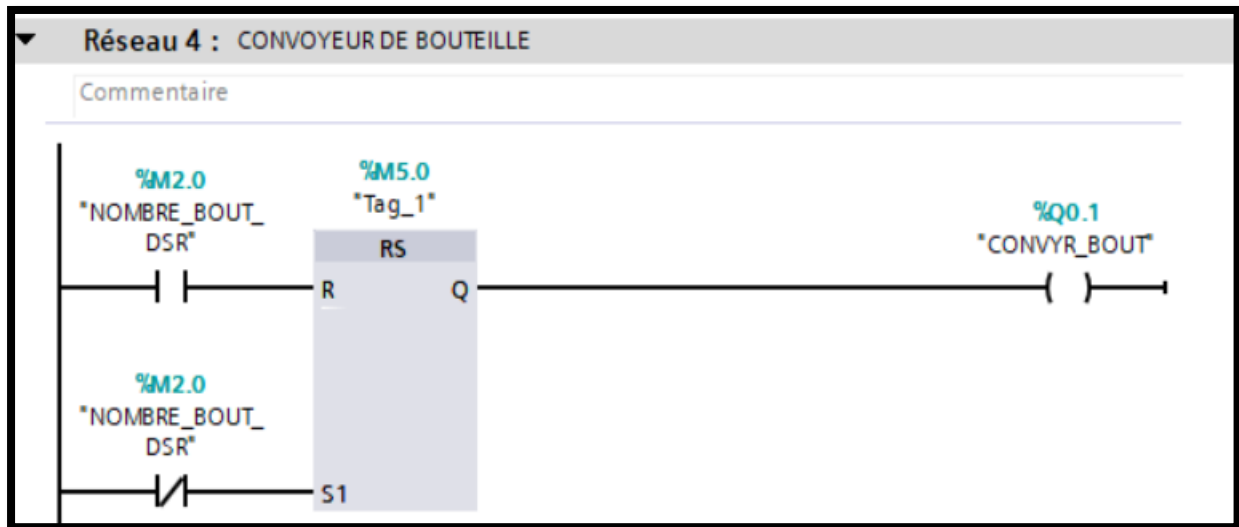


Figure IV.18– Programme Ladder de la commande du convoyeur de bouteille

### 3-Vérification de la présence de la palette et du nombre d'étages :

Le comptage du nombre d'étages commence dès que la pince (%Q0.3) est ouverte pour la dépose, le processus se poursuit jusqu'à ce que le nombre d'étages atteigne la valeur maximale de 4.



## Chapitre IV : Implémentation de la commande sous TIA PORTAL

Une temporisation de 15 secondes, gérée par le bit %M200.0, permet ensuite de relancer un nouveau cycle de comptage.

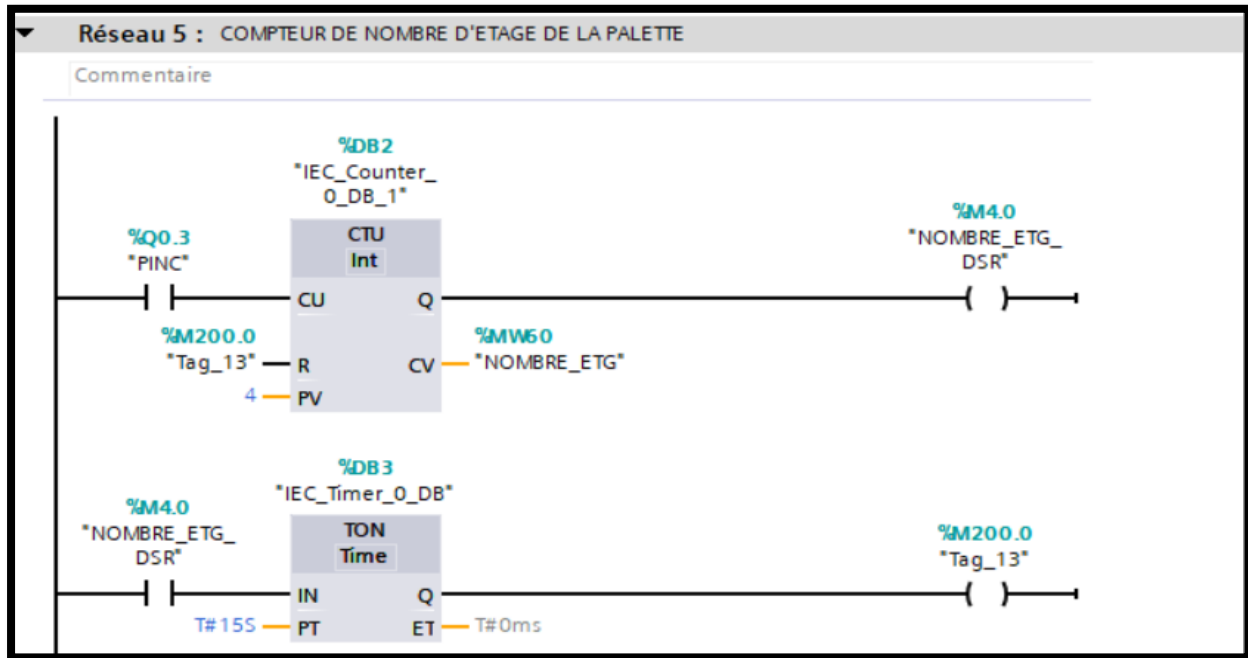


Figure IV.19– Programme Ladder de compteur de nombre d'étage

Le convoyeur de palette est activé lorsqu'aucune palette n'est détectée ou lorsque le nombre d'étages atteint la valeur limite de 4 (%M4.0). Il s'arrête automatiquement dès qu'une palette est détectée par le capteur (%I0.4).

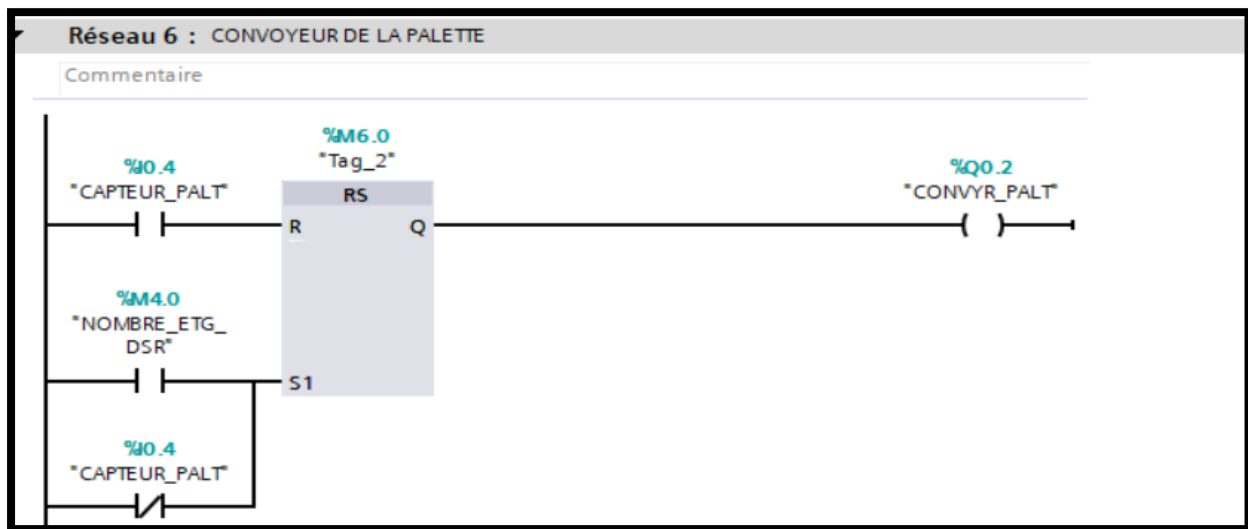


Figure IV.20– Programme Ladder de la commande du convoyeur

### IV.6.7 Mouvement des servomoteurs

Le déplacement du servomoteur (base) est géré par un bloc fonction activé en fonction des conditions de positionnement (mot de contrôle), voir annexe B.3.

#### 1- Traitement et stockage de la valeur du codeur :

L'information récupérée à partir de PIW78 (valeur lue du codeur) a d'abord été normalisée à l'aide de l'instruction NORM\_X, puis mise à l'échelle avec l'instruction SCALE\_X. Une fois traitée, la valeur ajustée a été enregistrée dans la mémoire %MD14 prête à être utilisée dans les étapes suivantes du programme.

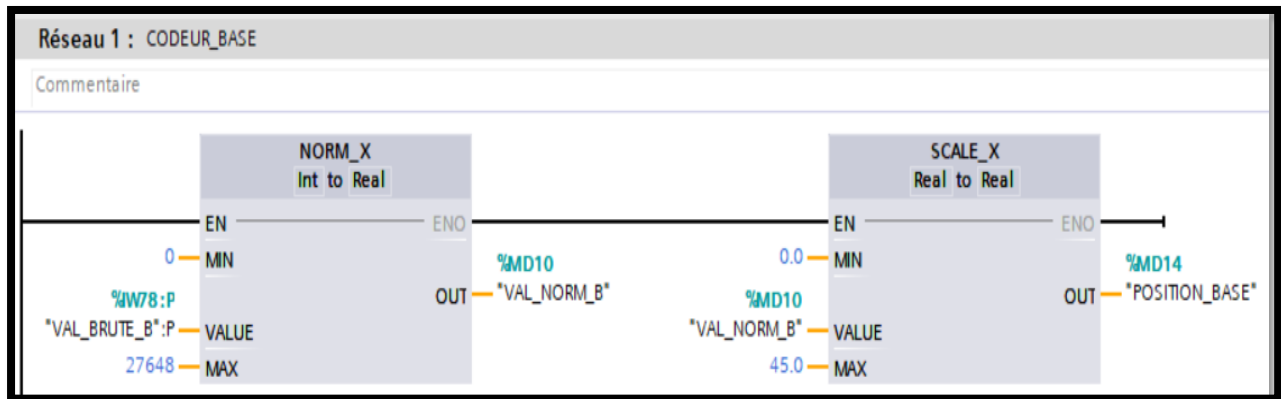


Figure IV.21– Calcul de la position angulaire de la base

#### 2-Contrôle de la marche et arrêt avant (à gauche) :

La première action du cycle consiste à effectuer une rotation de la base vers la gauche jusqu'à atteindre la consigne d'angle de rotation (45°). Cette action est déclenchée uniquement lorsque la base et l'épaule sont en position initiale, et que le compteur de bouteilles indique 60. Le mouvement s'interrompt automatiquement dès que la base atteint l'angle prévu.

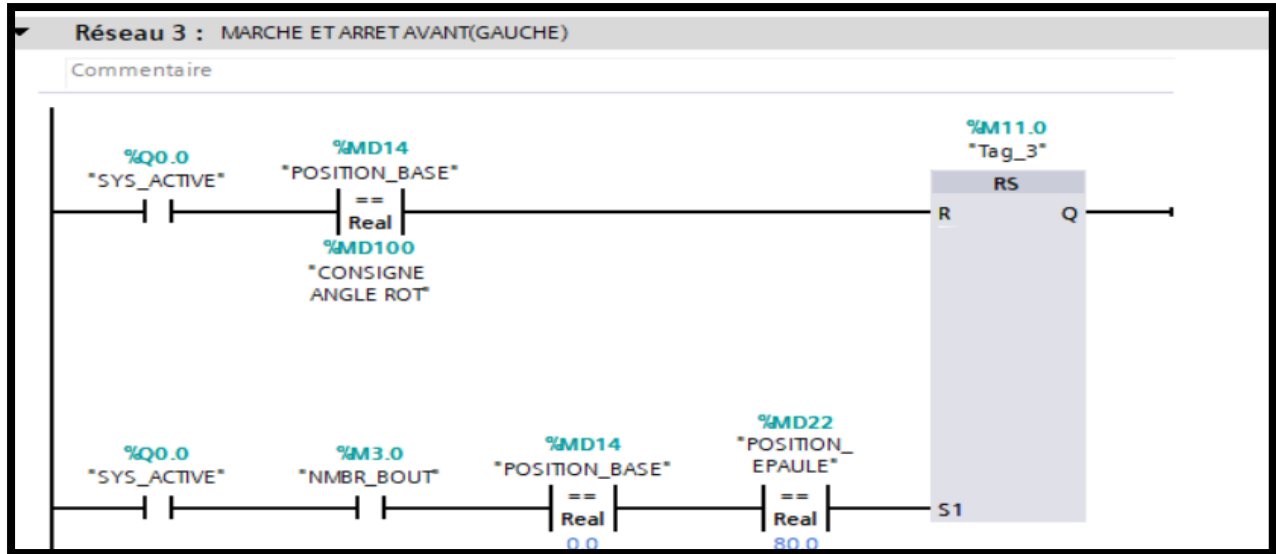


Figure IV.22– Commande de la marche et l’arrêt avant de la base

### 3-Consigne de vitesse pour la marche avant :

Le réseau suivant assure la transmission de l'ordre de démarrage(1151) vers la sortie PQW64, tandis que la sortie PQW66 envoie la consigne de vitesse nécessaire(2000) pour activer le servomoteur (base).

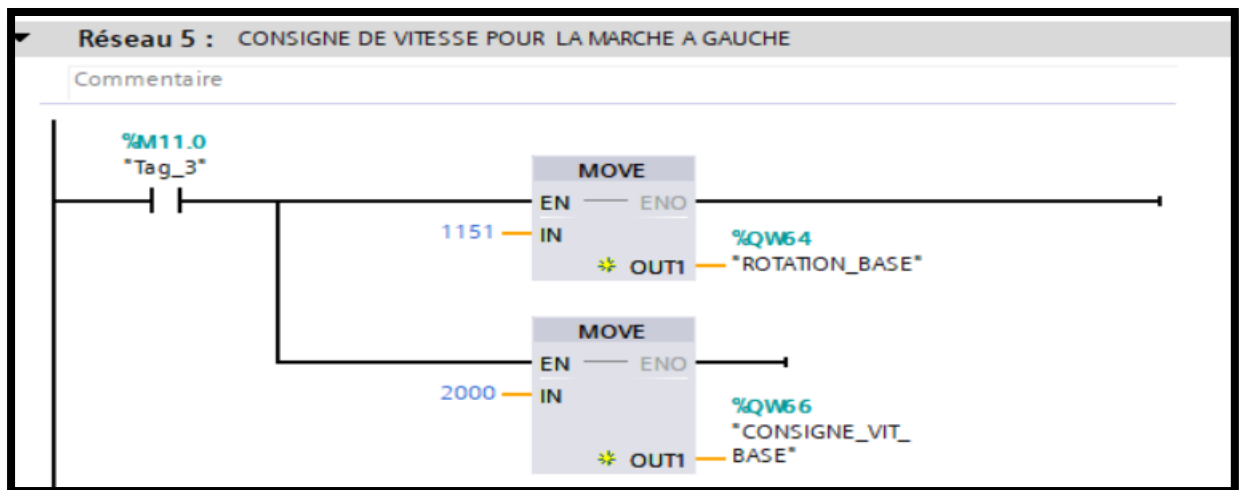


Figure IV.23– la marche avant de la base

### 4-Consigne de vitesse pour l’arrêt avant :

Dans ce réseau, l’ordre d’arrêt(1024) est transmis au servomoteur de la base par la sortie PQW64, et la sortie PQW66 est utilisée pour envoyer la consigne de vitesse(0).

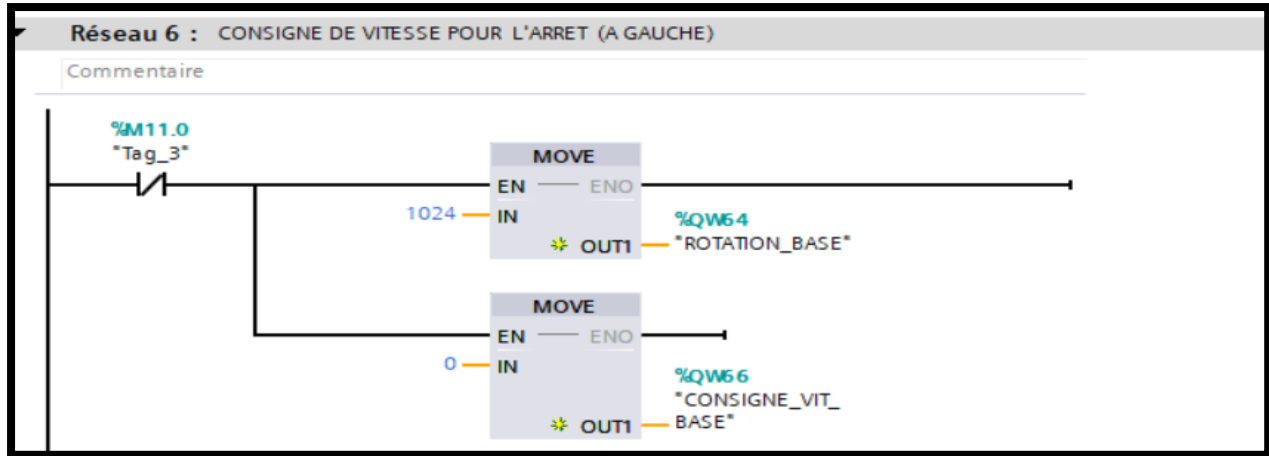


Figure IV.24– l'arrêt avant de la base

### 5-Contrôle de la marche et arrêt arrière (à droite) :

Après la prise des bouteilles, la base effectue une rotation en sens inverse (marche arrière) pour revenir à la position initiale (zone de dépose). Ce fonctionnement démarre lorsque la base est à 45°, que l'épaule est en position initiale, Le système s'arrête automatiquement lorsque la base revient à 0°.

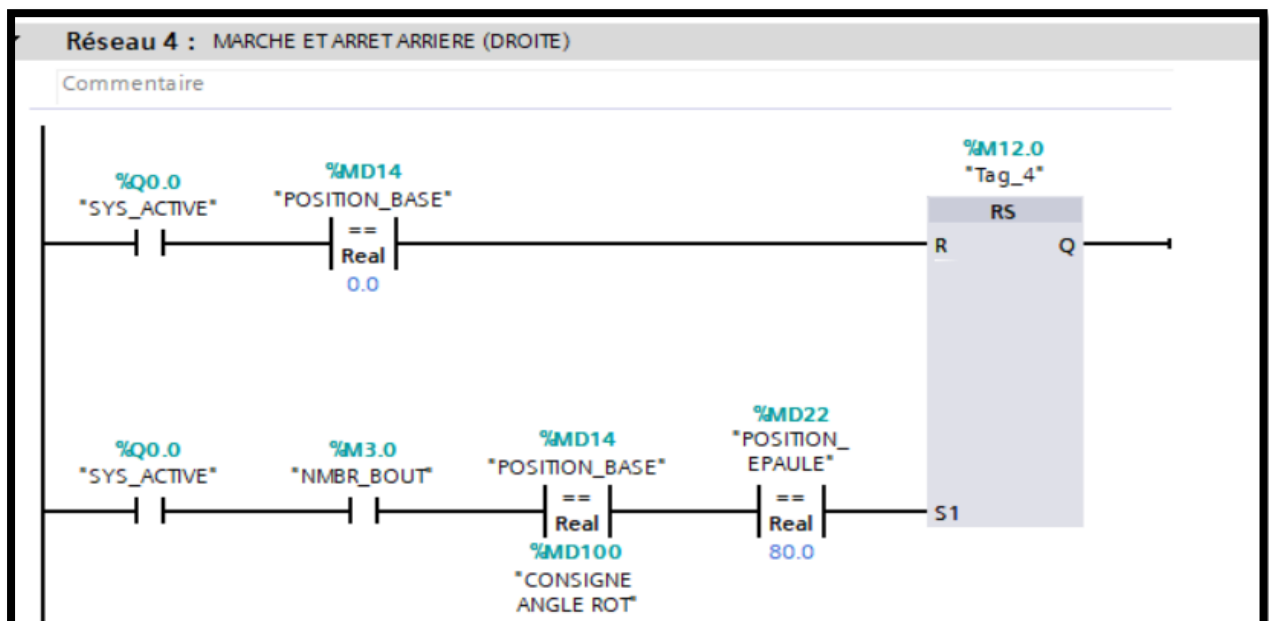


Figure IV.25– Commande de la marche et l'arrêt arrière de la base

### 6-Consigne de vitesse pour la marche arrière :

Ce réseau transmet l'ordre de démarrage(3199) à la sortie PQW64 et envoie la consigne de vitesse(2000) via la sortie PQW66 pour activer le servomoteur.

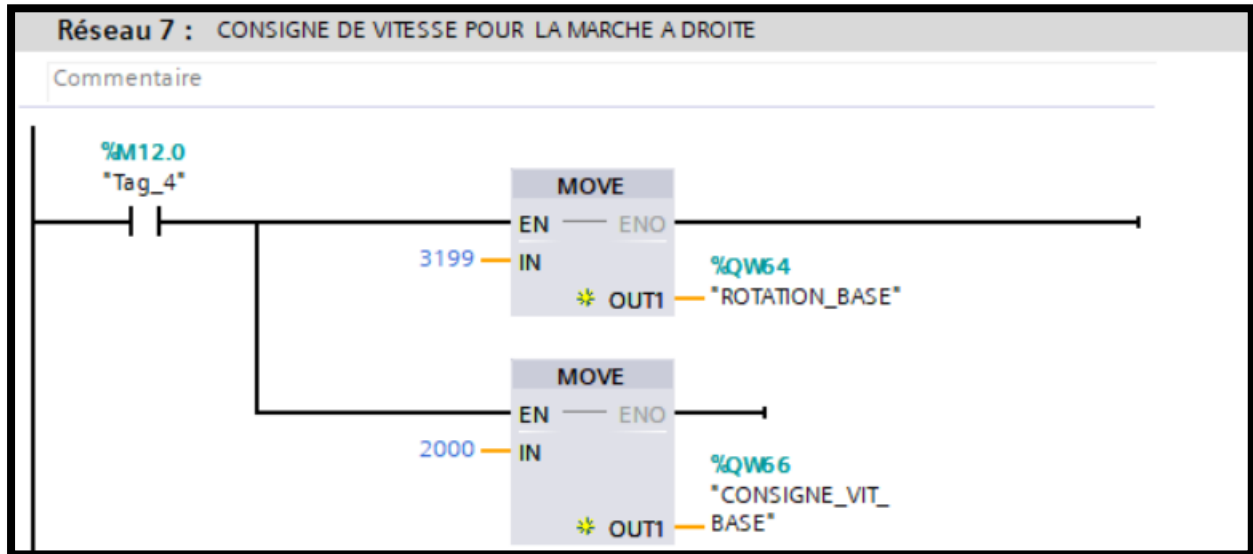


Figure IV.26– la marche arrière de la base

### 7-Consigne de vitesse pour l'arrêt arrière :

Ce réseau transmet l'ordre d'arrêt(3072) à la sortie PQW64 et envoie la consigne de vitesse (0) via la sortie PQW66 pour activer le servomoteur.

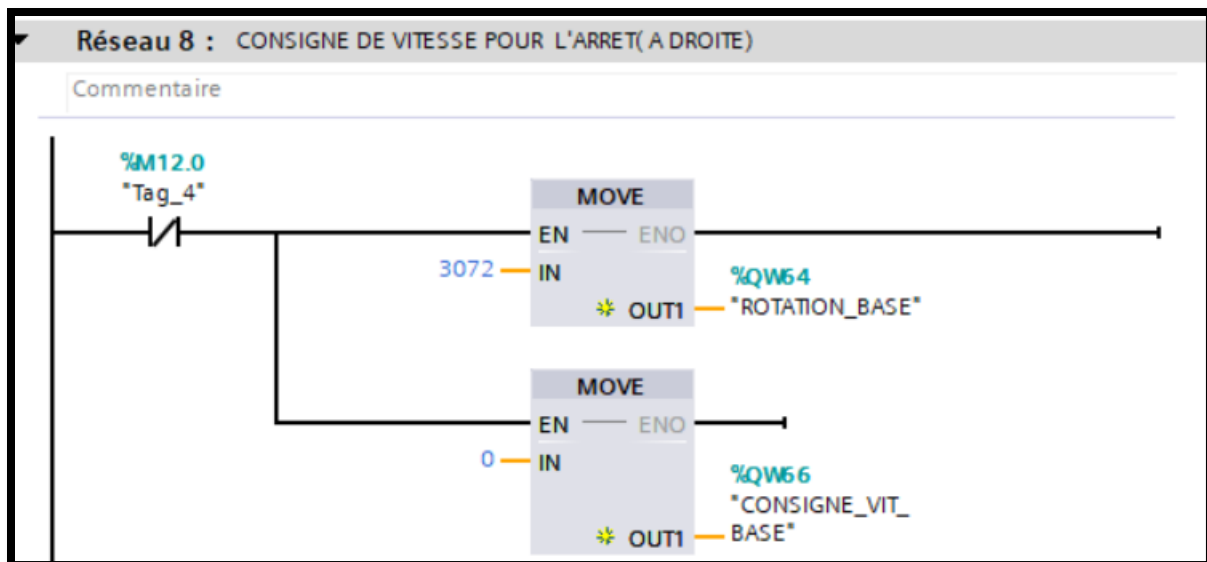


Figure IV.27– l'arrêt avant de la base

## Chapitre IV : Implémentation de la commande sous TIA PORTAL

### 8-Appel de la fonction dans le programme principal (Main) :

L'appel de la fonction de rotation de base dans le programme principal structure le code, facilitant sa gestion, réutilisation et maintenance tout en réduisant les erreurs.



Figure IV.28– Programme Ladder de la fonction de rotation de la base

### IV.6.8 Visualisation des résultats de simulation du processus

#### 1-Démarrage et activation du processus :

L'activation du bouton Start déclenche le démarrage du système, à condition qu'aucune situation d'urgence ne soit détectée.

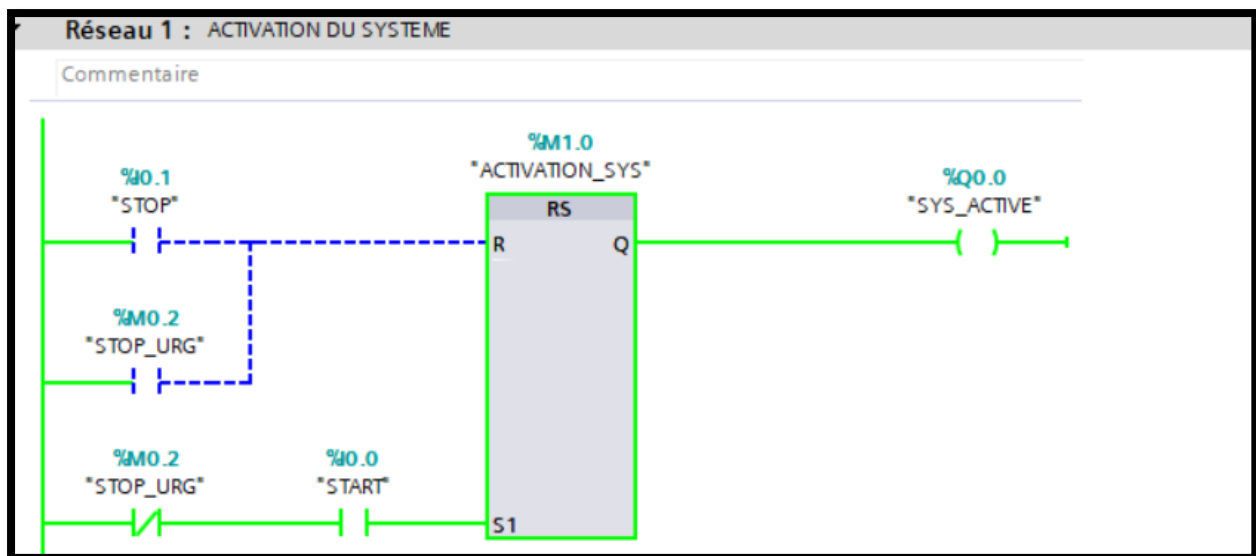


Figure IV.29-Simulation du bouton Start

L'activation de bouton Stop entraîne l'arrêt immédiat du système.

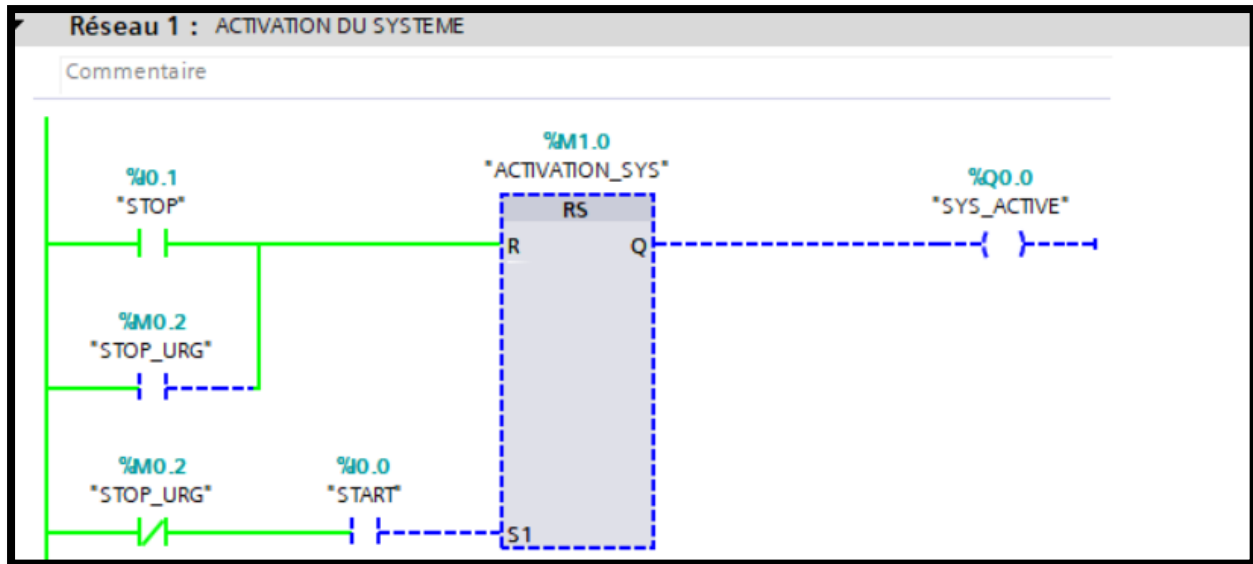


Figure IV.30- Simulation du bouton Stop

L'activation de l'arrêt d'urgence est déclenchée lorsque le nombre d'étages dépasse la valeur limite de 4.

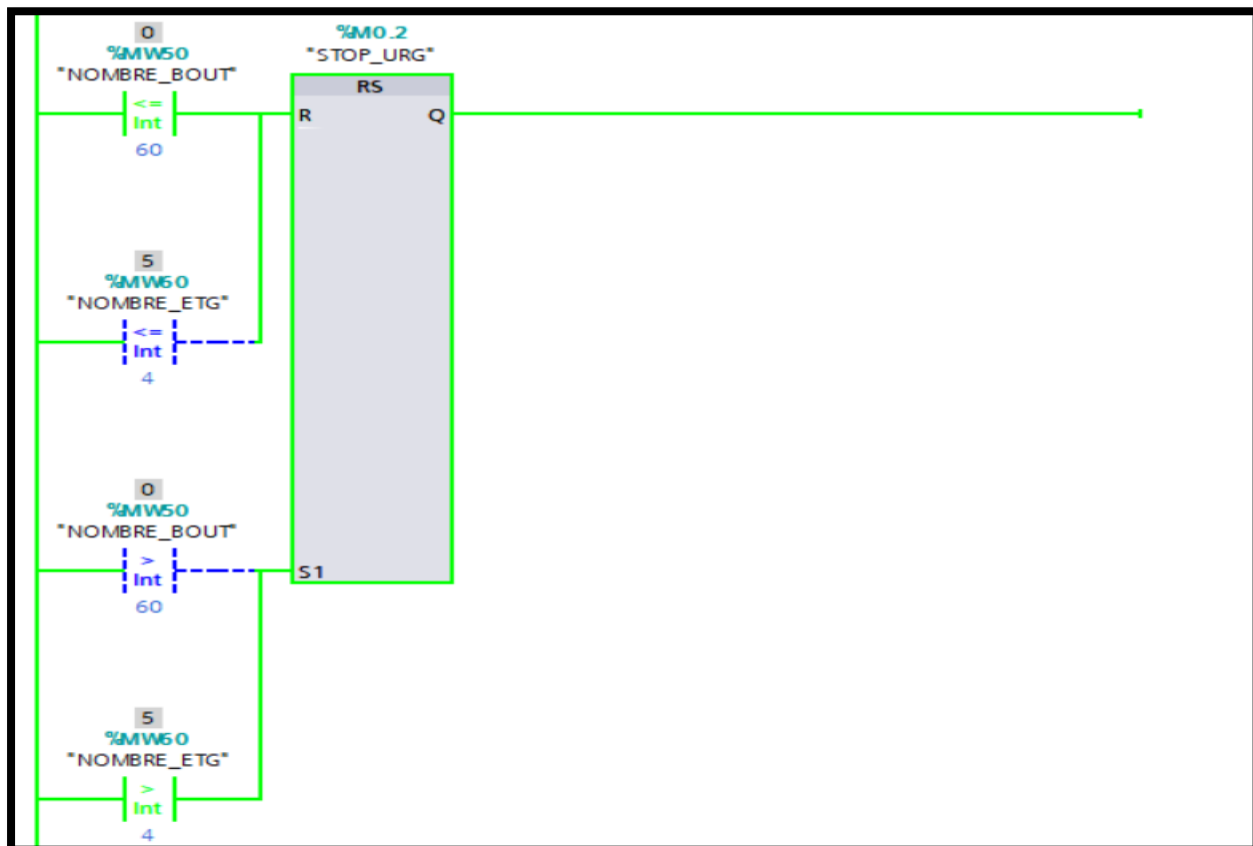


Figure IV.31- Simulation d'une condition d'arrêt d'urgence

## Chapitre IV : Implémentation de la commande sous TIA PORTAL

Dans la figure suivante, le système est désactivé en raison de l'activation de l'arrêt d'urgence, et ce malgré l'activation du bouton Start.

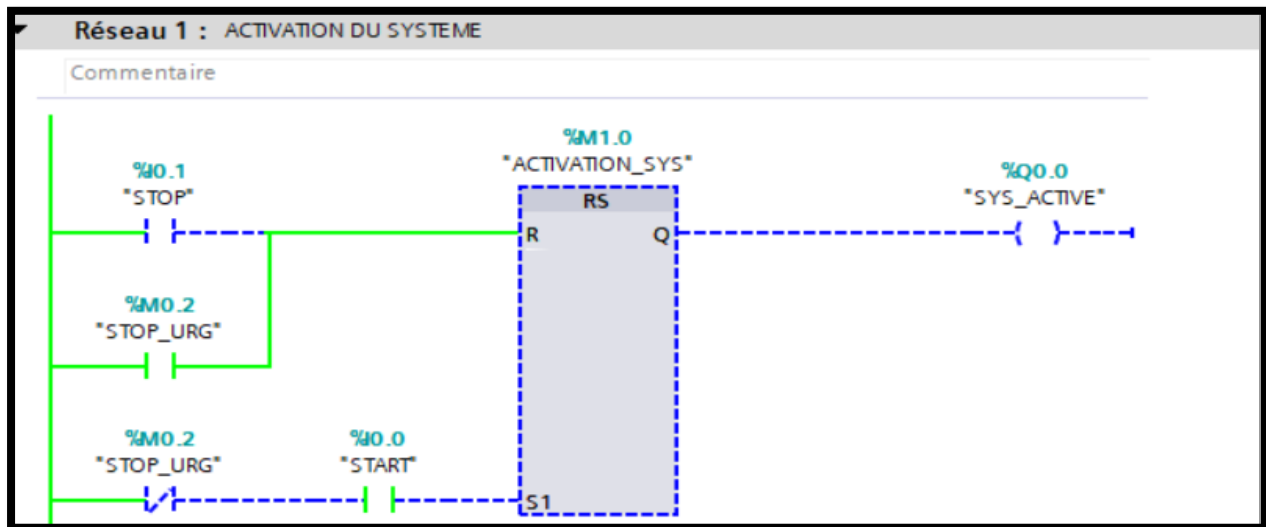


Figure IV.32- Simulation de l'arrêt d'urgence

### 2-Vérification du nombre de bouteilles :

La figure suivante illustre le réseau où le compteur enregistre un total de 60 bouteilles.

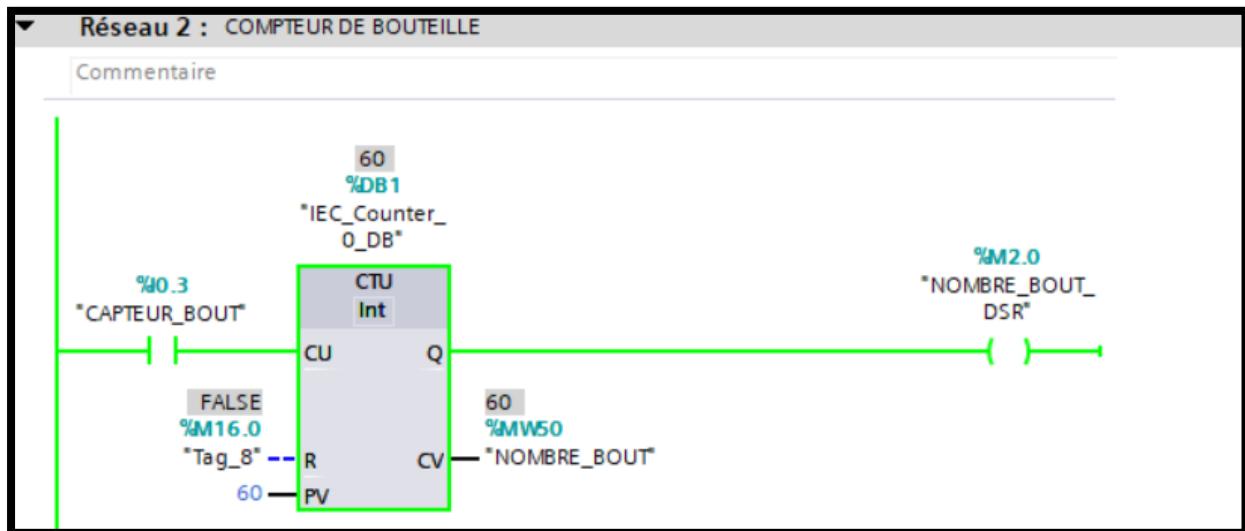


Figure IV.33- Simulation de compteur de bouteille

La réservation du nombre de 60 bouteilles dans le réseau suivant.



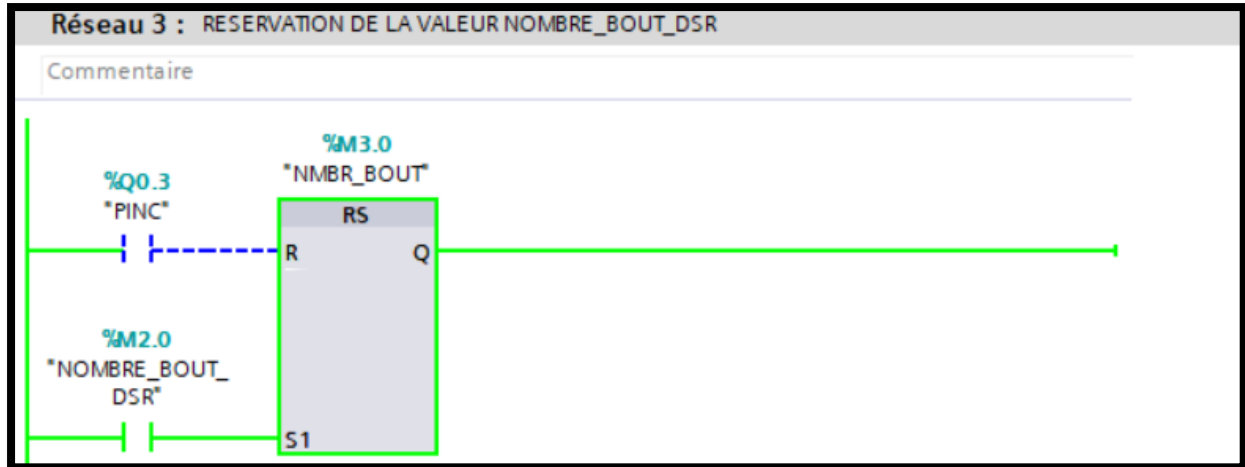


Figure IV.34-Simulation de la réservation de nombre de bouteille

Désactivation du convoyeur de bouteille lorsque le nombre de bouteilles atteint 60.

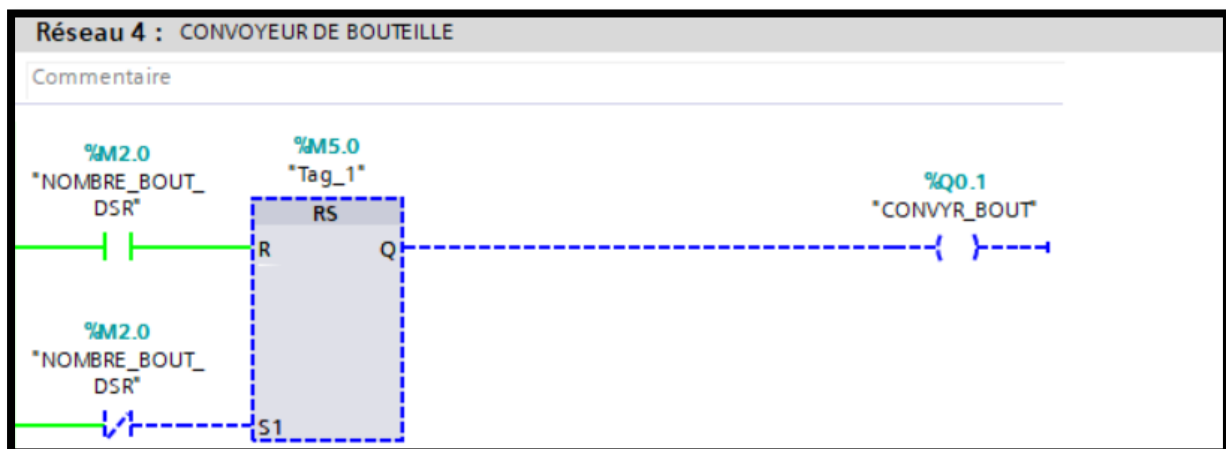


Figure IV.35-Simulation de la commande du convoyeur bouteille

### 3-Vérification de la présence de la palette et du nombre d'étages :

La figure suivante représente le réseau pour le cas où le nombre d'étages est égal à 4. Lorsque le compteur atteint 4 étages, un temporisateur est activé. Celui-ci démarre à 0 seconde et compte jusqu'à 15 secondes, délai nécessaire avant de réinitialiser le compteur et de démarrer un nouveau cycle de comptage pour la palette suivante.

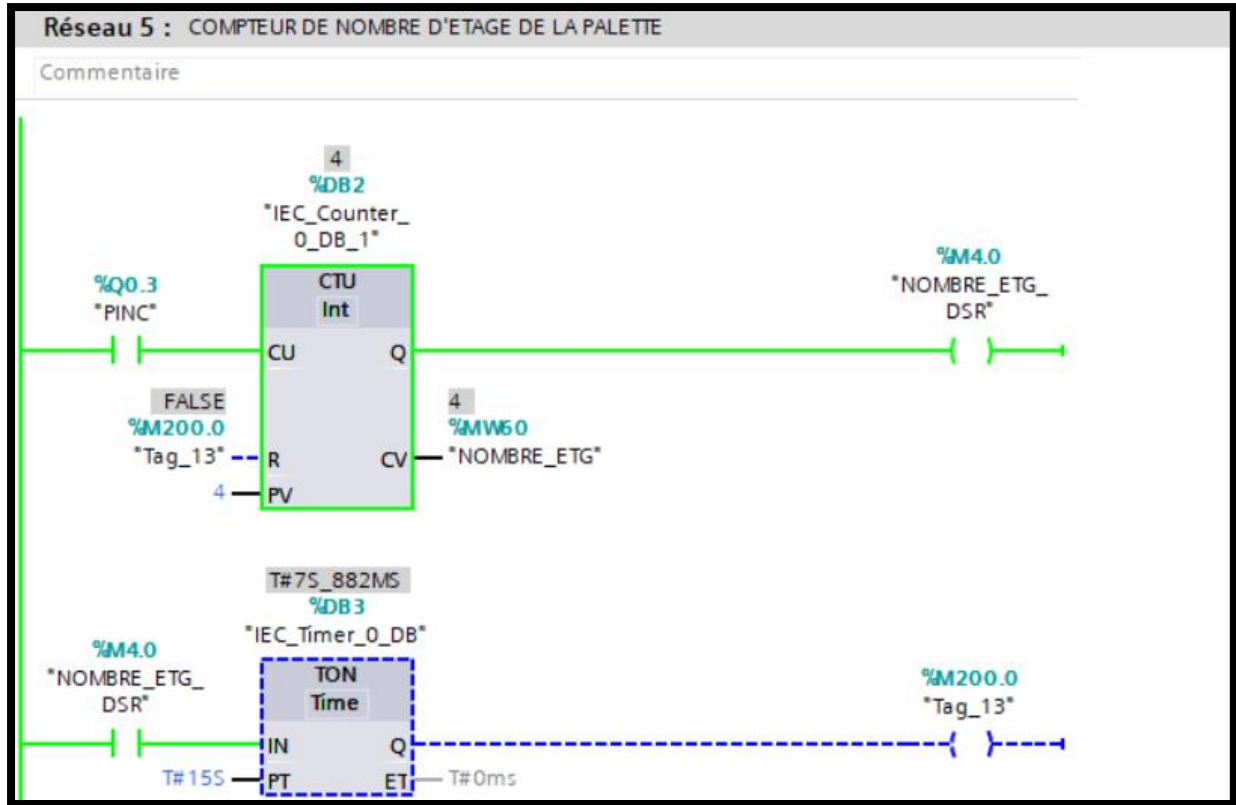


Figure IV.36-Simulation de compteur de nombre d'étage

Activation du convoyeur de palette en réponse à l'absence de palette afin d'en acheminer une nouvelle.

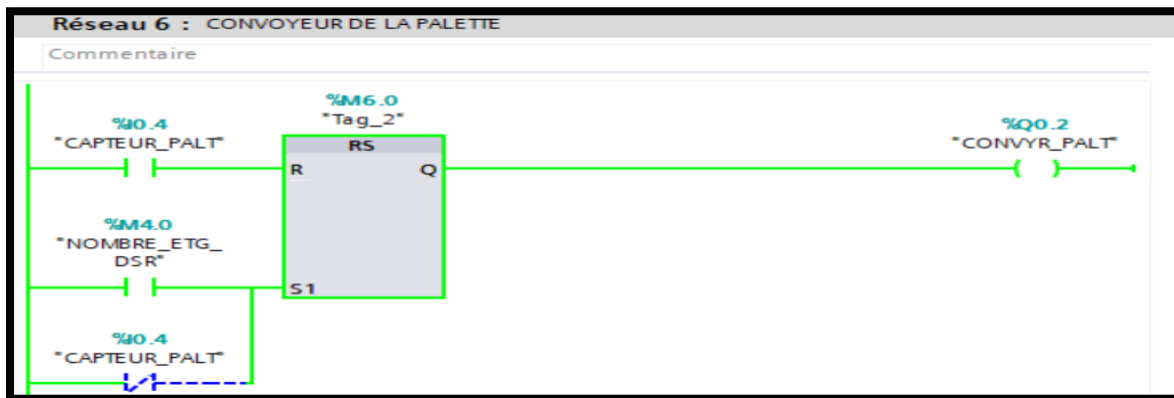


Figure IV.37-Simulation de convoyeur de palette

#### 4-Mouvement des servomoteurs (base) :

Récupération de la position de la base (0°).

## Chapitre IV : Implémentation de la commande sous TIA PORTAL

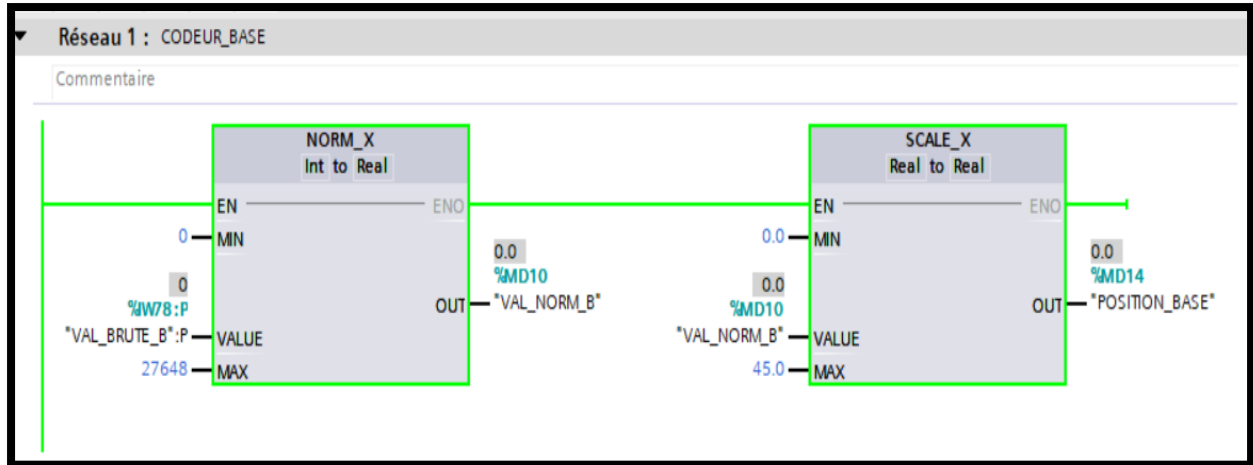


Figure IV.38-Simulation de la position angulaire de la base

Dans la figure IV.39, la marche de la base débute lorsque les conditions de la base sont remplies afin d'atteindre la consigne d'angle de rotation ( $45^\circ$ ) pour la base.

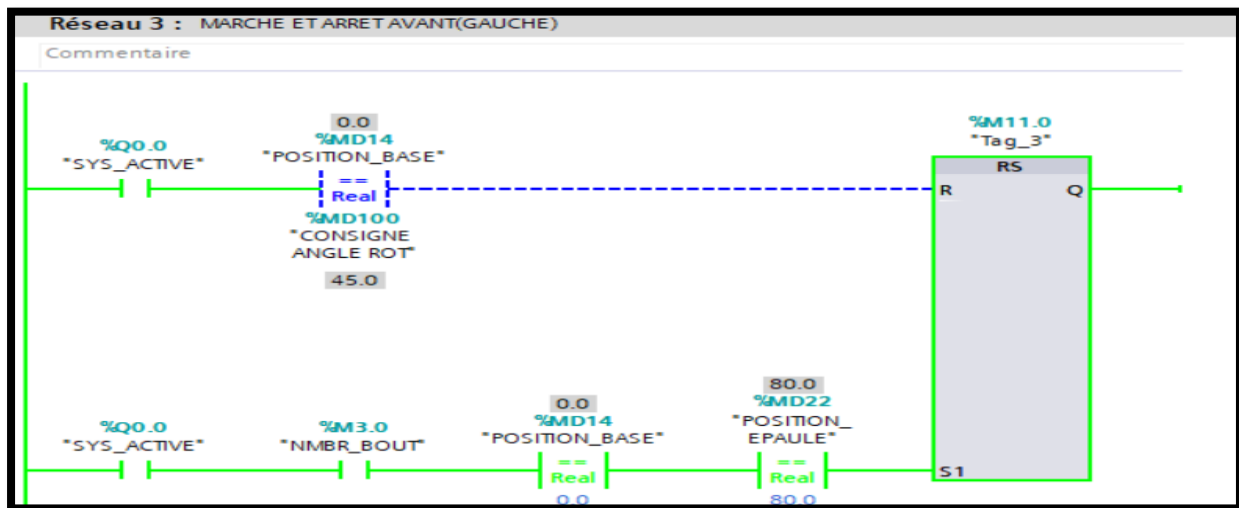


Figure IV.39-Simulation de la commande de la marche et l'arrêt avant de la base

Déclenchement de la marche du servomoteur de la base après activation de la commande.

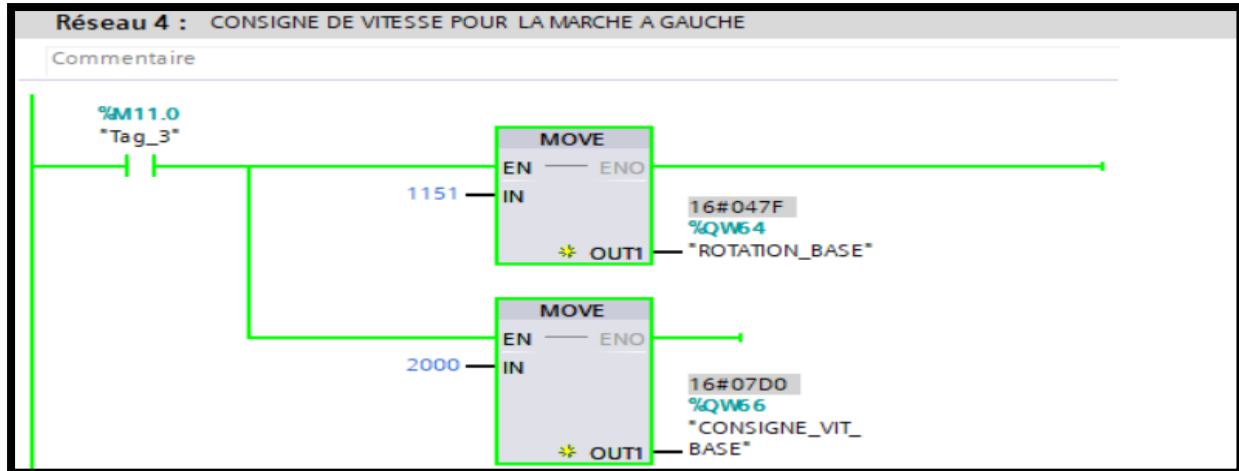


Figure IV.40-Simulation de la marche avant (à gauche) de la base

Arrêt du servomoteur de la base suite à la désactivation de la commande ce qui indique que l'angle de 45° a été atteint. .

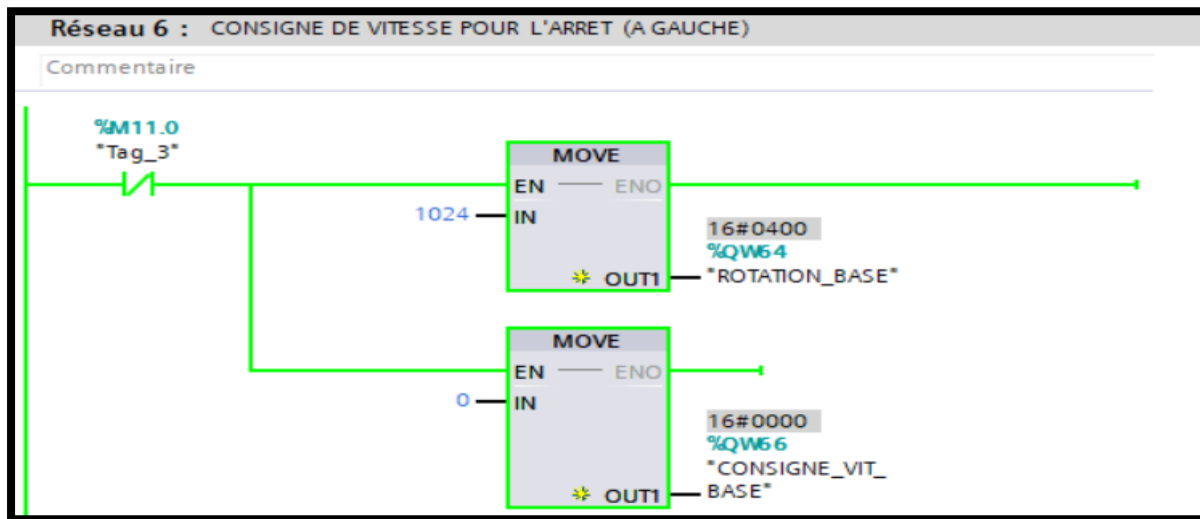


Figure IV.41-Simulation de l'arrêt avant (à gauche) de la base

### 5-Appel de la fonction dans le programme principal (Main) :

Appel de la fonction pour le contrôle du servomoteur de la base selon l'état de la commande.



Figure IV.42-Simulation des fonctions de rotation de la base

### IV.7. Supervision du bras palettiseur à l'aide de l'IHM

#### IV.7.1 Le choix et configuration de l'IHM

On commence par le choix du panneau IHM. Celui-ci, de type KTP1200 Comfort, est sélectionné depuis la bibliothèque matérielle de TIA Portal, puis connecté à l'API pour assurer la communication entre l'automate et l'interface, comme illustré dans Figures IV.43 et IV.44.

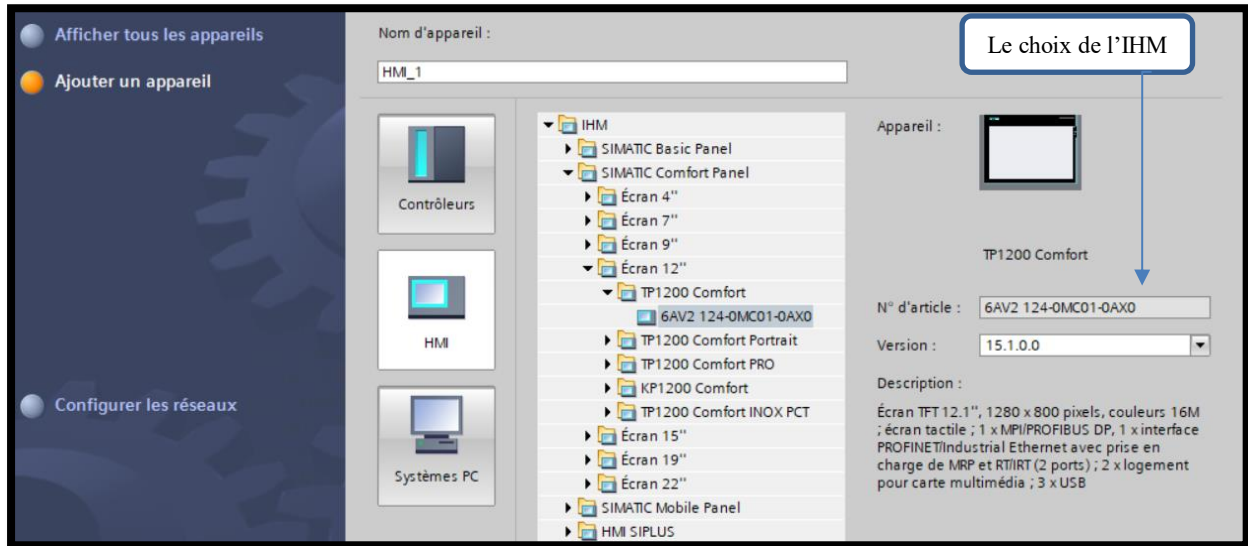


Figure IV.43-Le choix de la vue IHM

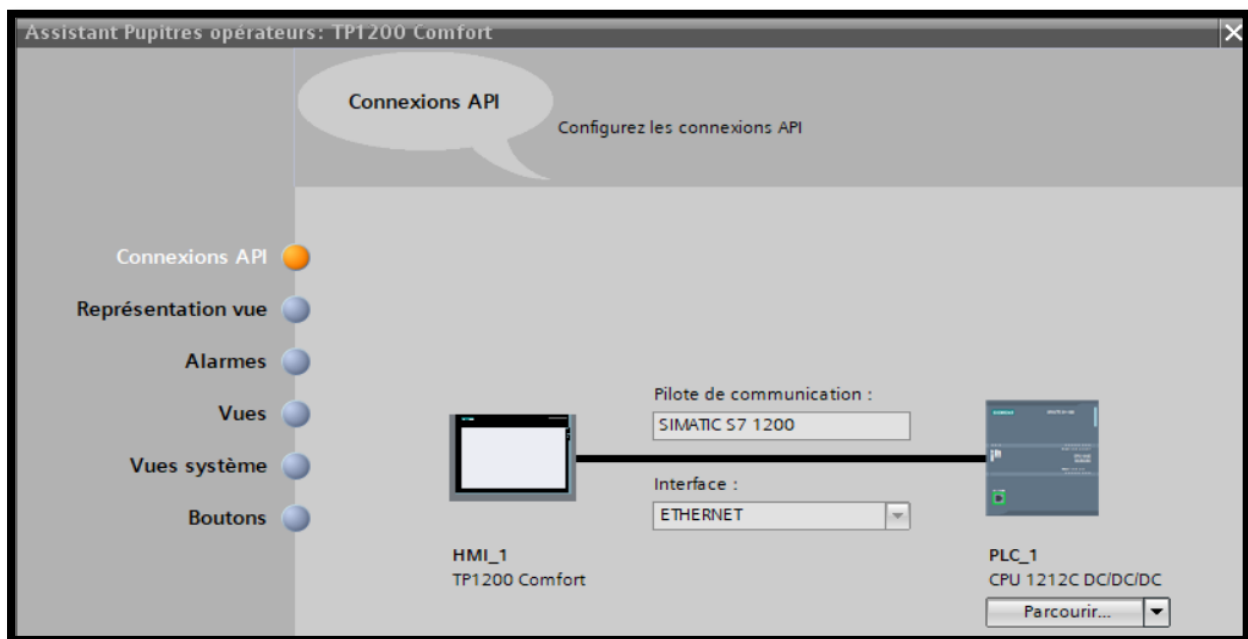


Figure IV.44- Configuration de la liaison de communication ETHERNET

### IV.7.2 Présentation de la vue IHM

L'IHM permet de superviser le système automatisé, qui comprend deux convoyeurs (pour les bouteilles et les palettes), deux capteurs de détection, deux moteurs assurant l'entraînement des convoyeurs, un bras robotisé à trois articulations plus la pince, un compteur de bouteilles, des zones de réglage des consignes d'angle pour chaque articulation, deux boutons de commande (Start/Stop) et un voyant signalant un arrêt d'urgence en rouge en cas de problème.

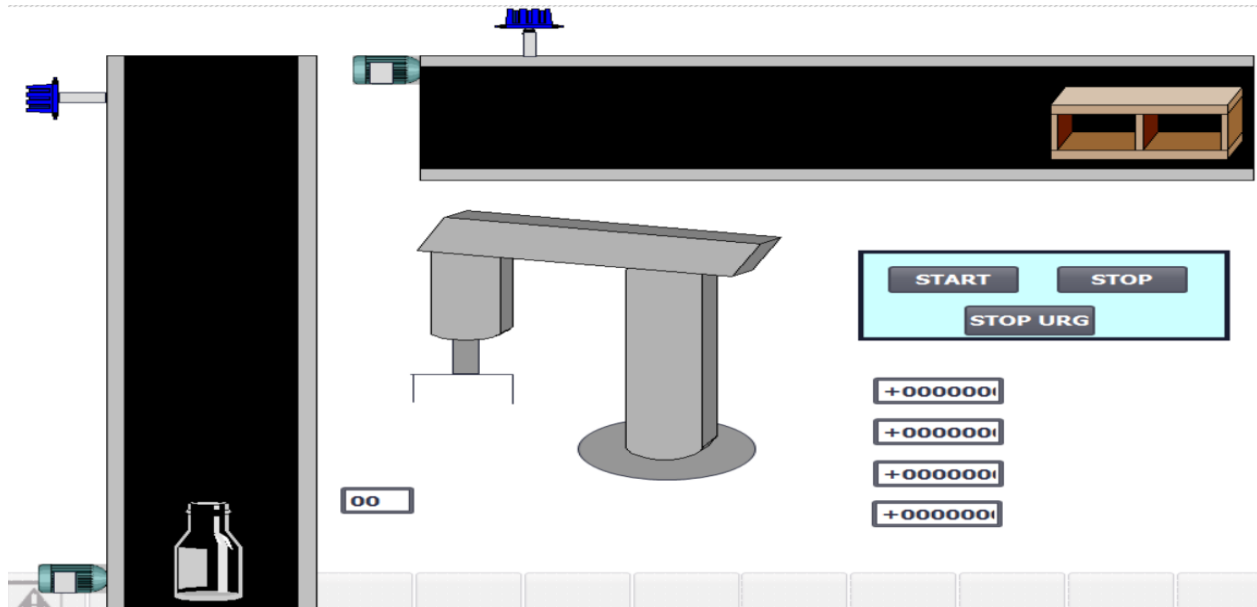


Figure IV.45- Interface de supervision de la palettisation

Les réseaux Ladder ci-dessous illustrent le contrôle des deux convoyeurs, l'un pour les bouteilles et l'autre pour les palettes. Chaque réseau gère l'activation des moteurs en fonction des capteurs de détection et des commandes des convoyeurs mentionnés précédemment (Figures IV.18 et IV.20).

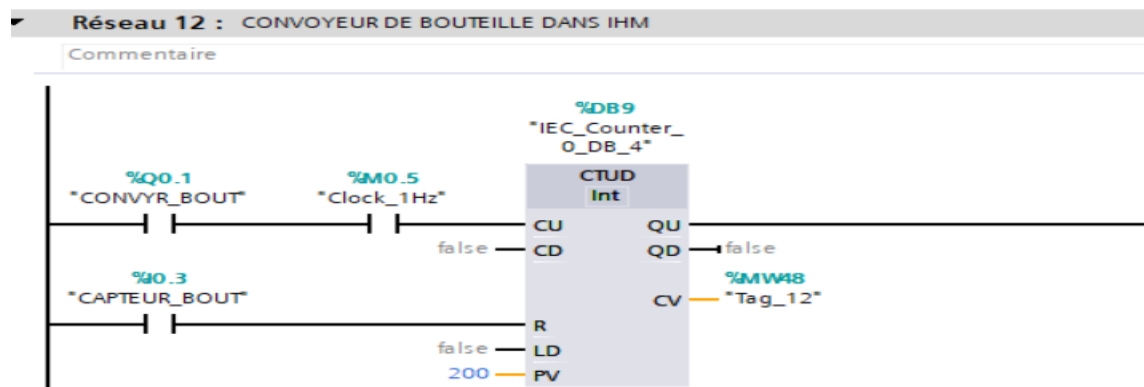


Figure IV.46- Réseau Ladder pour la commande du convoyeur de bouteilles

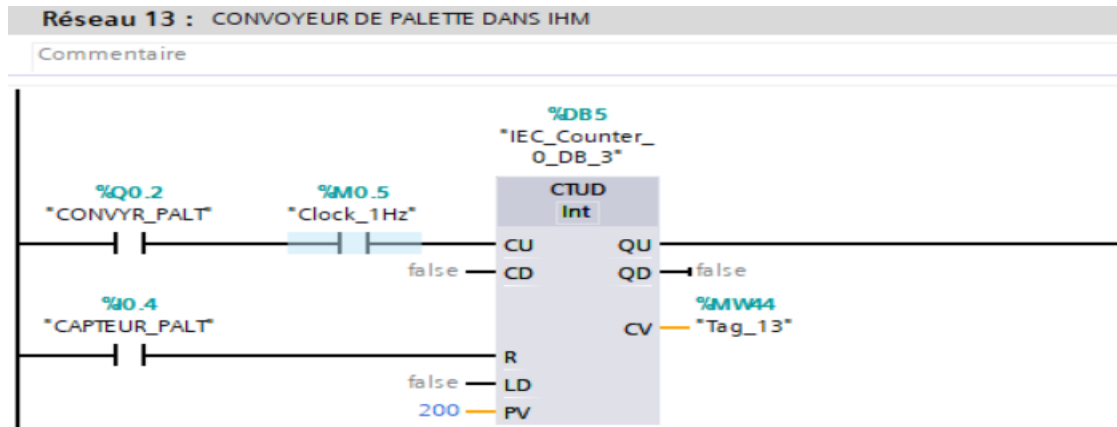


Figure IV.47- Réseau Ladder pour la commande du convoyeur de palette

### IV.7.3 Résultats de simulation de l'Interface Homme-Machine (IHM)

**Remarque :** La couleur verte dans l'IHM indique ou de la fonction correspondante

Au démarrage du cycle, les deux convoyeurs sont activés. Le capteur de détection des bouteilles est activé, une bouteille étant déjà positionnée à côté du capteur, ce qui entraîne l'augmentation du compteur, qui indique un total de 12 bouteilles. Le convoyeur de bouteilles continue à fonctionner jusqu'à ce que 60 bouteilles soient détectées, tandis que le convoyeur de palettes se désactive lors de la détection d'une palette.

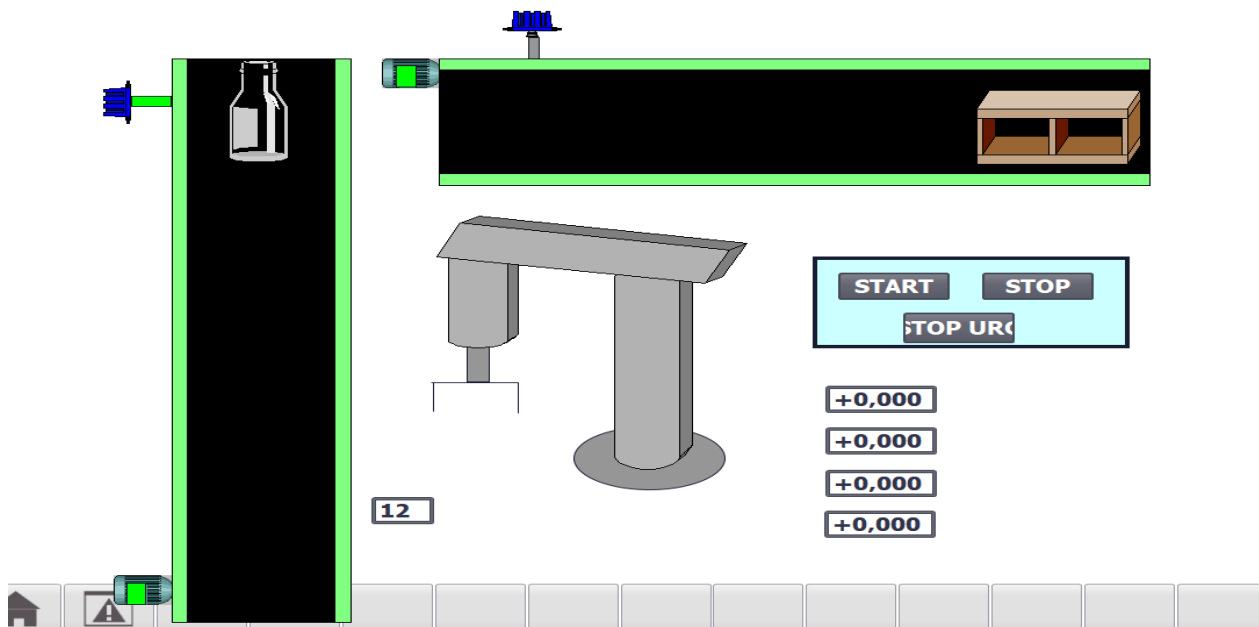


Figure IV.48- fonctionnement des convoyeurs et capteurs de détection

## Chapitre IV : Implémentation de la commande sous TIA PORTAL

La figure suivante représente l'activation de l'arrêt d'urgence en cas de dépassement du seuil de 60 bouteilles. Lorsque ce seuil est atteint, l'arrêt d'urgence se déclenche automatiquement, désactivant les convoyeurs et les moteurs, garantissant ainsi la sécurité du système.

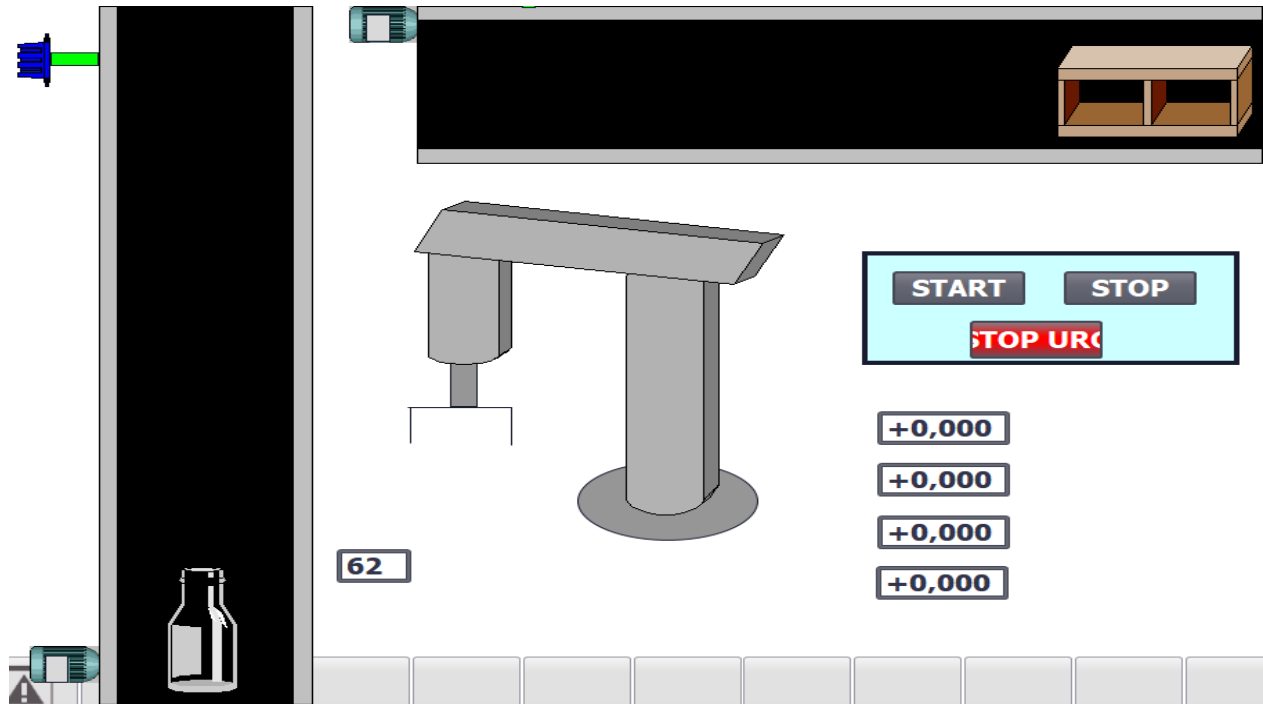


Figure IV.49- Activation de l'arrêt d'urgence

### IV.8 Conclusion :

Dans ce chapitre, nous avons présenté les principales étapes de notre contribution à l'automatisation d'un bras robotisé palettiseur au sein de l'entreprise Cevital. À travers la conception du programme sous TIA Portal V15, l'intégration sur l'automate S7-1200, la simulation du fonctionnement et le développement d'une interface homme-machine, nous avons mis en place un système cohérent et opérationnel, répondant aux exigences du processus de palettisation.



### Conclusion générale

Ce projet de fin d'études nous a permis de mettre en pratique nos connaissances théoriques dans un cadre industriel, en participant à l'automatisation d'une étape importante du processus de production d'huile chez CEVITAL. Nous avons conçu et développé un système de palettisation utilisant un bras robotisé à trois degrés de liberté, commandé par un automate Siemens S7-1200 à l'aide du logiciel TIA Portal.

Notre travail s'est divisé en deux parties principales. La première partie concerne la modélisation du bras robotisé : nous avons étudié sa géométrie, ses mouvements (cinématique) et les efforts qu'il subit (dynamique), ainsi que la planification de trajectoires dans l'espace articulaire, réalisée avec MATLAB. La seconde partie a porté sur la mise en œuvre de la commande du système de palettisation, en utilisant l'automate, les variateurs de vitesse et les servomoteurs, programmés via le logiciel TIA Portal.

Grâce à ce projet, nous avons renforcé nos compétences en automatisation industrielle, en programmation d'automates et en robotique.

Malgré les résultats satisfaisants obtenus, ce projet peut encore être amélioré, notamment par l'implémentation réelle du système, l'ajout de capteurs intelligents pour améliorer la supervision, ou encore l'utilisation d'un bras avec plus de trois degrés de liberté pour offrir d'avantage de flexibilité. Ces pistes ouvrent la voie à de futures recherches et améliorations.

## Liste des bibliographies :

- [1] : H. Hamdi, " *Introduction à la robotique* ", Les éditions de l'université Mentouri Constantine 2002-2000.
- [2]: Thi Van Anh Nguyen, " *Commande de robots manipulateurs basée sur le modèle de Takagi-Sugeno : nouvelle approche pour le suivi de trajectoire* ", Thèse de doctorat, université polytechnique HAUTS-DE-France, Valenciennes le 04/10/2019.
- [3] : L. Boulhraouat, " *Valorisation des déchets solides issus du complexe CEVITAL* ", mémoire master, université de Constantine 1, 2023.
- [4] : M Farid, AZC Malika, " *Caractérisation de la formation des cadres aux TIC dans les grandes entreprises de la région de Béjaïa : Cas de Cevital Agro-industrie* ", Revue Algérienne d'Economie et gestion Vol, 2021 - asjp.cerist.dz.
- [5] : <https://www.cevital.com>
- [6] : B. Ikhlef, C.Issad, " *La planification de la production Cas de l'entreprise Cevital* ", mémoire, université de Bejaïa, 2019.
- [7]: J. Thibault, " *Palettiseurs et palettisation* ", Techniques de l'Ingénieur, Réf. A9280, 10 juillet 1998.
- [8] :E. Dombre et W. Khalil, " *Modélisation, identification et commande des robots* ", 15/01/1999.
- [9] : A .Lahbib, " *Généralités sur les bras de robots* ", support de cours, Université de Biskra, 2011.
- [10]: F. Morbidi, " *Initiation à la Robotique* ",support de cours, Université de Picardie france,2022.
- [11]:S. Bowling and L. Charpentier, " *Variable Frequency Drives: Installation and Troubleshooting* ", Pearson Education, 2007.
- [12]: N. Bellahsene, " *Représentation d'un automatisme industriel* ", support de cours, Université de Bejaïa, 2022.

- [13] : B. Ait salah, O. Medjahed, "*Commande intelligente d'un bras de robot manipulateur*", Mémoire d'ingénieur Université A.Mira Bejaia, 2010.
- [14] : B. Mendil " *Commande des bras de manipulation* ", Support de cours, Université Bejaia, 2025.
- [15] : R. Merzouki "*Cours et TP Robotique de Manipulation* ", Master SMART, 2010-2011.
- [16] : A. Sahi, M. Semara «*Planification du chemin et commande d'un bras de robot*» Mémoire master, Université A.Mira Bejaia, 2022.
- [17] :L.Bouakkache, K. Baazizi "*Commande d'un bras robotisé avec un automate siemens*", mémoire master, université de Bejaia, 2023.
- [18] : Manuel d'utilisateur SIMATIC S7-1200.
- [19] : Manuel d'utilisateur SINAMICS V90, SIMOTICS S-1FL.
- [20]: A.Ferrah et K.Hacene, «*Commande d'un moteur asynchrone triphasé autour d'un API S7-1200 de SIEMENS en utilisant les bus de communication industriels PROFIBUS et PROFINET*», Mémoire de Master, université de Médéa, 2019.

## A.1 Génération des trajectoires cubiques des trois articulations

### Trajectoire cubique de 1<sup>er</sup> mouvement articulaire

```
function [tetad1,tetapd1,tetappd1]= fcn(teta10,teta1f,tf,tt)

tf2=tf*tf; tf3=tf2*tf;

a0=teta10; a1=0; a2=3*(teta1f-teta10)/tf2; a3=-2*(teta1f-teta10)/tf3;
tetad1=(a0+a1*tt+a2*tt^2+a3*tt^3)*pi/180;
tetapd1=(a1+2*a2*tt+3*a3*tt^2)*pi/180;
tetappd1=(2*a2+6*a3*tt)*pi/180;

end
```

### Trajectoire cubique de 2<sup>ème</sup> mouvement articulaire

```
function [tetad2,tetapd2,tetappd2]= fcn(teta20,teta2f,tf,tt)

tf2=tf*tf; tf3=tf2*tf;

a0=teta20; a1=0; a2=3*(teta2f-teta20)/tf2; a3=-2*(teta2f-teta20)/tf3;
tetad2=(a0+a1*tt+a2*tt^2+a3*tt^3)*pi/180;
tetapd2=(a1+2*a2*tt+3*a3*tt^2)*pi/180;
tetappd2=(2*a2+6*a3*tt)*pi/180;

end
```

### Trajectoire cubique de 3<sup>ème</sup> mouvement articulaire

```
function [tetad3,tetapd3,tetappd3]= fcn(teta30,teta3f,tf,tt)

tf2=tf*tf; tf3=tf2*tf;

a0=teta30; a1=0; a2=3*(teta3f-teta30)/tf2; a3=-2*(teta3f-teta30)/tf3;
tetad3=(a0+a1*tt+a2*tt^2+a3*tt^3)*pi/180;
tetapd3=(a1+2*a2*tt+3*a3*tt^2)*pi/180;
tetappd3=(2*a2+6*a3*tt)*pi/180;

end
```

## A.2 Commande PD des mouvements articulaires

### Commande PD de 1<sup>er</sup> mouvement articulaire

```
function u1 = PD_1(tetad1,teta1,wd1,w1,wpd1)
u1=(650*(tetad1-teta1)+50*(wd1-w1)+wpd1);
```

### Commande PD de de 2<sup>ème</sup> mouvement articulaire

```
function u2 = PD_2(tetad2,teta2,wd2,w2,wpd2)
u2=(1600*(tetad2-teta2)+80*(wd2-w2)+wpd2);
% kp=wn*wn    kv=2*wn
```

### Commande PD de 3<sup>ème</sup> mouvement articulaire

```
function u3 = PD_3(tetad3,teta3,wd3,w3,wpd3)
u3=(650*(tetad3-teta3)+50*(wd3-w3)+wpd3);
```

## A.3 modèle dynamique

### Les éléments de la matrice d'inertie $M(\Theta)$

$$m_{11}=2.57+1.38 c_{22}+0.30 s_2 s_3+0.744 c_2 s_{23}$$

$$m_{12}=m_{21}= 0.69 s_2-0.134 c_{23}+0.0238 c_2$$

$$m_{13}=m_{31}= -0.134 c_{23}-0.00397 s_{23}$$

$$m_{22}= 6.79+0.744 s_3$$

$$m_{23}=m_{32}=0.333+0.372 s_3-0.011 c_3$$

$$m_{33}=1.16$$

### Les éléments de la matrice des forces centrifuge et Coriolis $V(\Theta, \dot{\Theta})$

$$v_{11} \approx - 2.76 s_2 c_2 + 0.744 c_{223} + 0.6 s_2 c_3 - 0.02.13(1-2 s_2 s_3) \dot{\Theta}_2$$

$$v_{12} \approx (0.69 c_2 + 0.134 s_{23} - 0.0238 s_2) \dot{\Theta}_2 + (0.267 s_{23} - 0.00758 c_{23}) \dot{\Theta}_3$$

$$v_{13} \approx (0.744 c_2 c_{23} + 0.6 s_2 c_3 + 0.022 c_2 s_{23} - 0.0213 (1-2 s_2 s_3)) \dot{\Theta}_1 + 0.5 (0.267 s_{23} - 0.00758 c_{23}) \dot{\Theta}_3$$

$$v_{21} \approx - 0.5(-2.76 s_2 c_2 + 0.744 c_{223} + 0.6 s_2 c_3 - 0.0213 (1-2 s_2 s_3)) \dot{\Theta}_1$$

$$v_{22} \approx (0.022 s_3 + 0.744 c_3) \dot{\Theta}_3$$

$$v_{23} \approx 0.5*(0.022 s_3 + 0.744 c_3) \dot{\Theta}_3$$

$$v_{31} \approx - 0.5 (0.744 c_2 c_{23} + 0.6 s_2 c_3 + 0.022 c_2 s_{23} - 0.0213 (1- 2 s_2 s_3)) \dot{\Theta}_1$$

$$v_{32} \approx - 0.5 (0.022 s_3 + 0.744 c_3) \dot{\Theta}_2$$

$$v_{33} = 0$$

### Les éléments du vecteur des forces gravitationnelles $G(\Theta)$

$$g_1 = 0$$

$$g_2 = -37.2 c_2 - 8.4 s_{23} + 1.02 s_2$$

$$g_3 = -8.4 s_{23} + 0.25 c_{23}$$

#### A.4 Programme MATLAB du modèle dynamique

```
function [wp1,wp2,wp3]= DYNAMIC_MODEL(To1,To2,To3,teta1,teta2,teta3,w1,w2,w3)
to=[To1;To2;To3];
M=[2.57+1.38*cos(teta2)^2+0.30*sin(teta2)*sin(teta3)+7.44*10^-
1*cos(teta2)*sin(teta2+teta3),6.90*10^-1*sin(teta2)-1.34*10^-1*cos(teta2+teta3)+2.38*10^-
2*cos(teta2),-1.34*10^-1*cos(teta2+teta3)-3.97*10^-3*sin(teta2+teta3);
6.90*10^-1*sin(teta2)-1.34*10^-1*cos(teta2+teta3)+2.38*10^-2*cos(teta2),6.79+7.44*10^-
1*sin(teta3),0.333+3.72*10^-1*sin(teta3)-1.10*10^-2*cos(teta3);
-1.34*10^-1*cos(teta2+teta3)-3.97*10^-3*sin(teta2+teta3),0.333+3.72*10^-1*sin(teta3)-
1.10*10^-2*cos(teta3),1.16];
V=[((-2.76*sin(teta2)*cos(teta2)+7.44*10^-1*cos(teta2+teta2+teta3)+0.60*sin(teta2)*cos(teta3)-
2.13*10^-2*(1-2*sin(teta2)*sin(teta3)))*w2,((6.90*10^-1*cos(teta2)+1.34*10^-
1*sin(teta2+teta3)-2.38*10^-2*sin(teta2))*w2+(2.67*10^-1*sin(teta2+teta3)-7.58*10^-
3*cos(teta2+teta3))*w3),(7.44*10^-
1*cos(teta2)*cos(teta2+teta3)+0.60*sin(teta2)*cos(teta3)+2.20*10^-
2*cos(teta2)*sin(teta2+teta3)-2.13*10^-2*(1-2*sin(teta2)*sin(teta3)))*w1+0.5*(2.67*10^-
1*sin(teta2+teta3)-7.58*10^-3*cos(teta2+teta3))*w3;
(-0.5*(-2.76*sin(teta2)*cos(teta2)+7.44*10^-
1*cos(teta2+teta2+teta3)+0.60*sin(teta2)*cos(teta3)-2.13*10^-2*(1-
2*sin(teta2)*sin(teta3)))*w1,((2.20*10^-2*sin(teta3)+7.44*10^-
1*cos(teta3))*w3),(0.5*(2.20*10^-2*sin(teta3)+7.44*10^-1*cos(teta3))*w3);
(-0.5*(7.44*10^-1*cos(teta2)*cos(teta2+teta3)+0.60*sin(teta2)*cos(teta3)+2.20*10^-
2*cos(teta2)*sin(teta2+teta3)-2.13*10^-2*(1-2*sin(teta2)*sin(teta3)))*w1,(-0.5*(2.20*10^-
2*sin(teta3)+7.44*10^-1*cos(teta3))*w2),0];
V=V*[w1;w2;w3];
G=[0;-37.2*cos(teta2)-8.4*sin(teta2+teta3)+1.02*sin(teta2);-
8.4*sin(teta2+teta3)+0.25*cos(teta2+teta3)];
wp=inv(M)*(to-(V+G));
wp1=wp(1);wp2=wp(2);wp3=wp(3);
```

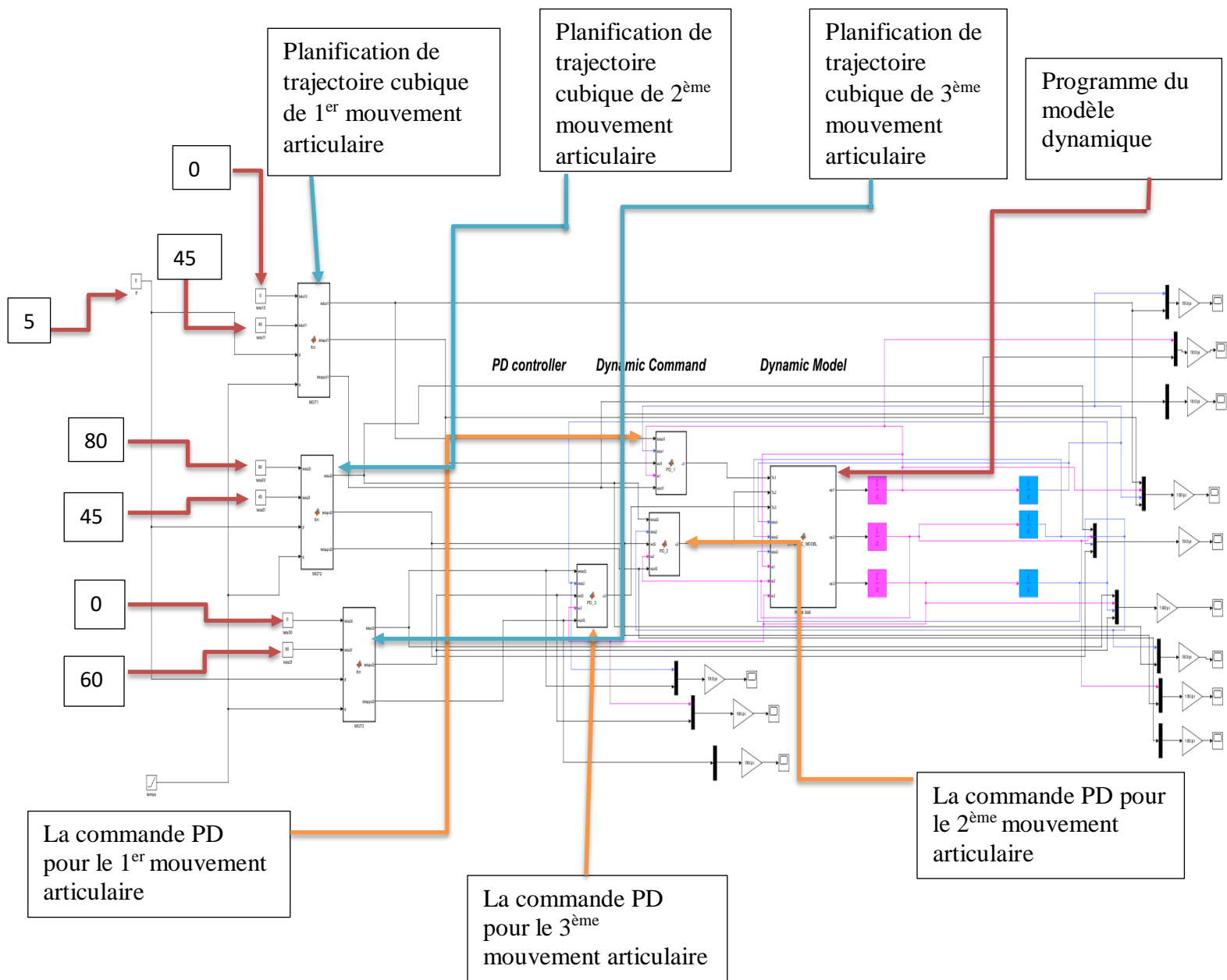


Figure A.1– Organigramme de la Commande Dynamique d'un bras Robotisé du modèle PUMA à 3 ddl avec Génération de Trajectoire et Régulation PD

## Annexe B

### B.1 Télégrammes du SINAMICS V90

Télégramme	Mot de réception	Mot d'émission	Description
Télégramme standard 1	2	2	p0922 = 1
Télégramme standard 2	4	4	p0922 = 2
Télégramme standard 3	5	9	p0922 = 3
Télégramme standard 5	9	9	p0922 = 5
Télégramme standard 7	2	2	p0922 = 7
Télégramme standard 9	10	5	p0922 = 9
Télégramme Siemens 102	6	10	p0922 = 102
Télégramme Siemens 105	10	10	p0922 = 105
Télégramme Siemens 110	12	7	p0922 = 110
Télégramme Siemens 111	12	12	p0922 = 111
Télégramme Siemens 750	3	1	p8864 = 750

### B.2 Télégrammes utilisés pour le mode régulation de vitesse

Télégramme	1		2		3		5		102		105	
Classe d'application	1		1		1, 4		4		1, 4		4	
PZD1	STW1	ZSW1	STW1	ZSW1	STW1	ZSW1	STW1	ZSW1	STW1	ZSW1	STW1	ZSW1
PZD2	NSOLL_A	NIST_A	NSOLL_B	NIST_B	NSOLL_B	NIST_B	NSOLL_B	NIST_B	NSOLL_B	NIST_B	NSOLL_B	NIST_B
PZD3	Recevoir un télégramme de PROFINET ↑	Envoyer un télégramme à PROFINET ↓										
PZD4			STW2	ZSW2	STW2	ZSW2	STW2	ZSW2	STW2	ZSW2	STW2	ZSW2
PZD5					G1_STW	G1_ZSW	G1_STW	G1_ZSW	MOMRED	MELDW	MOMRED	MELDW
PZD6									G1_STW	G1_ZSW	G1_STW	G1_ZSW
PZD7						G1_XIST1	XERR	G1_XIST1				
PZD8										G1_XIST1	XERR	G1_XIST1
PZD9						G1_XIST2	KPC	G1_XIS2				
PZD10										G1_XIST2	KPC	G1_XIST2



## Annexe B

---

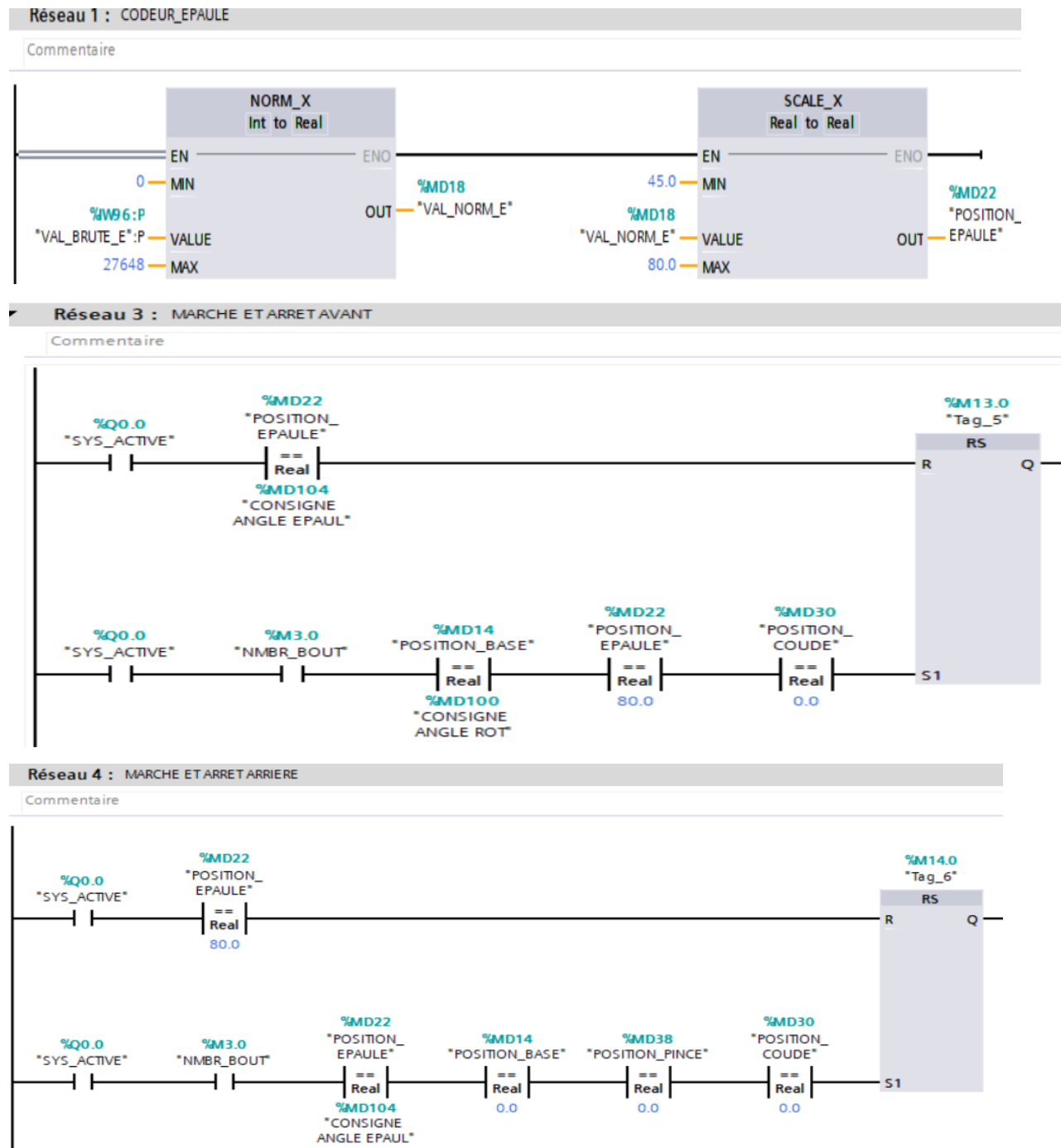
### B.3 Mot de contrôle du télégramme

Bit	Signification	Explication
0	0 = OFF1 1 = ON	Commande ON/OFF : OFF1 arrête selon la rampe p1121, ON autorise le démarrage.
1	0 = OFF2 1 = No OFF2	OFF2 : arrêt immédiat (coupure brutale, sécurité).
2	0 = OFF3 1 = No OFF3	OFF3 : arrêt rapide avec la rampe OFF3 (p1135).
3	0 = Inhibit 1 = Enable operation	Active ou inhibe le fonctionnement global du variateur.
4	0 = Disable RFG 1 = Enable RFG	Autorise ou non la sortie du générateur de rampe.
5	0 = Hold RFG 1 = Follow setpoint	Maintient ou autorise l'évolution de la consigne (rampe).
6	0 = Inhibit setpoint 1 = Enable setpoint	Bloque ou applique la consigne de vitesse.
7	0 = No fault acknowledge 1 = Fault acknowledge	Reset des défauts (ACK).
8	Réservé	Non utilisé.
9	Réservé	Non utilisé.
10	0 = Local control 1 = Control via PLC	Active la commande par automate (PLC).
11	0 = Normal rotation 1 = Reverse rotation	Inverse le sens de rotation du moteur.
12	Réservé	Non utilisé.
13	Réservé	Non utilisé.
14	Réservé	Non utilisé.
15	0 = No change 1 = Change operating mode	Permet de changer le mode de fonctionnement du variateur.

## Annexe B

### B.4 L'ensemble des réseaux Ladder utilisés dans le bras robotisé

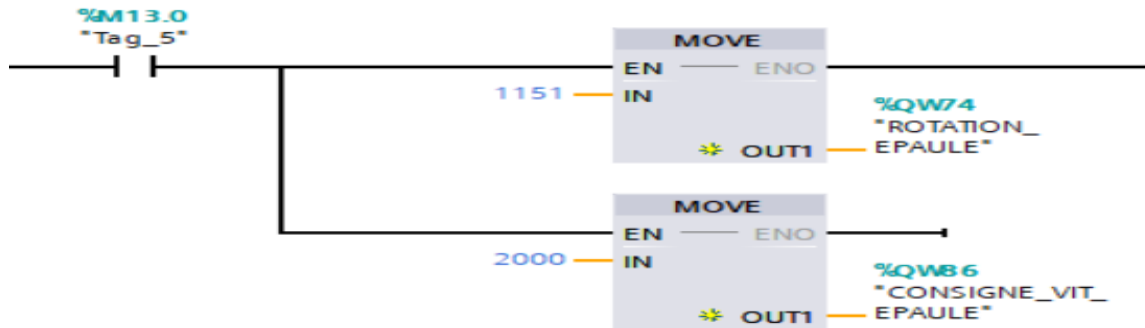
#### Moteur épaule :



## Annexe B

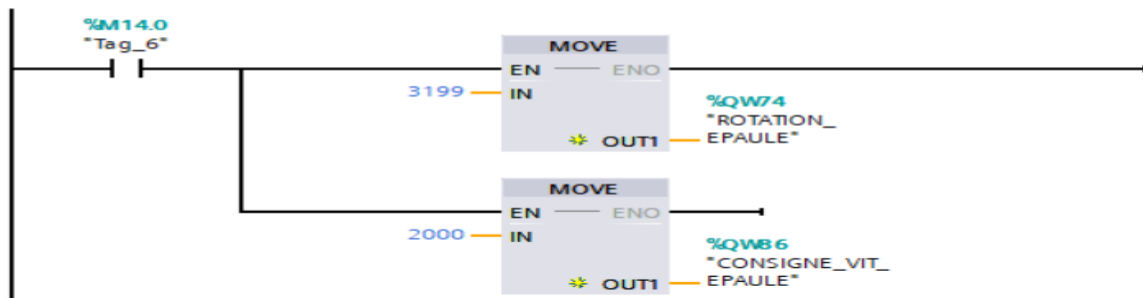
### Réseau 5 : CONSIGNE DE VITESSE POUR LA MARCHÉ A GAUCHE

Commentaire



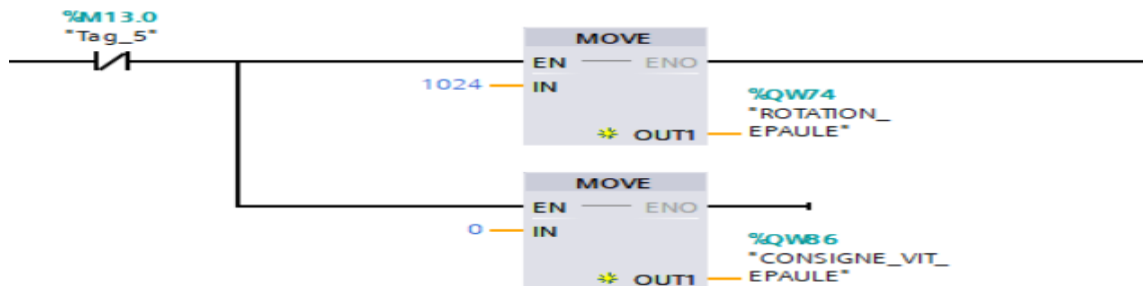
### Réseau 7 : CONSIGNE DE VITESSE POUR LA MARCHÉ A DROITE

Commentaire



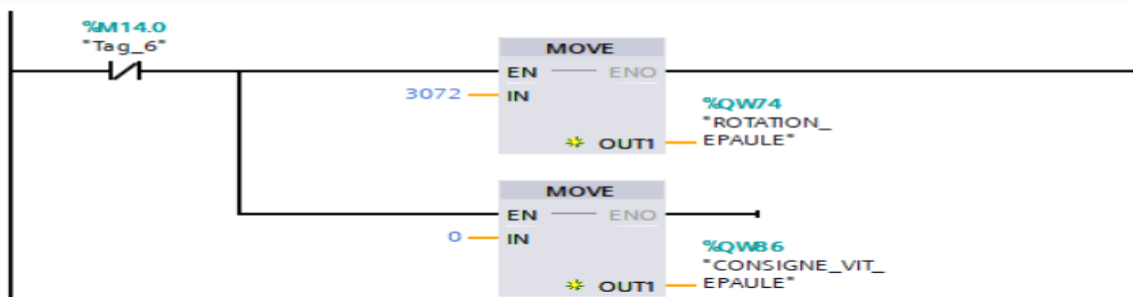
### Réseau 6 : CONSIGNE DE VITESSE POUR L'ARRET (A GAUCHE)

Commentaire



### Réseau 8 : CONSIGNE DE VITESSE POUR L'ARRET( A DROITE)

Commentaire

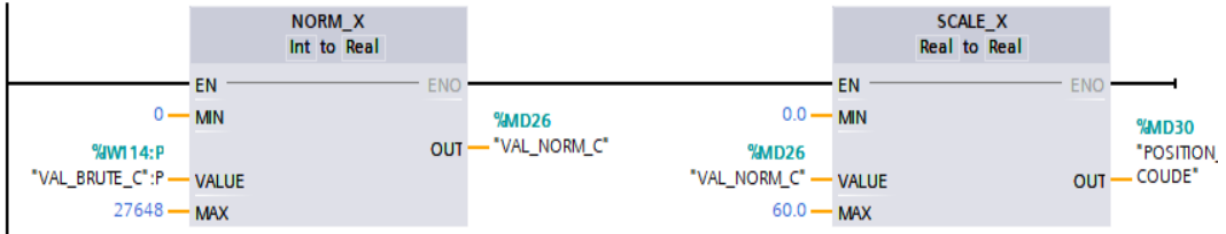


## Annexe B

### Moteur coude :

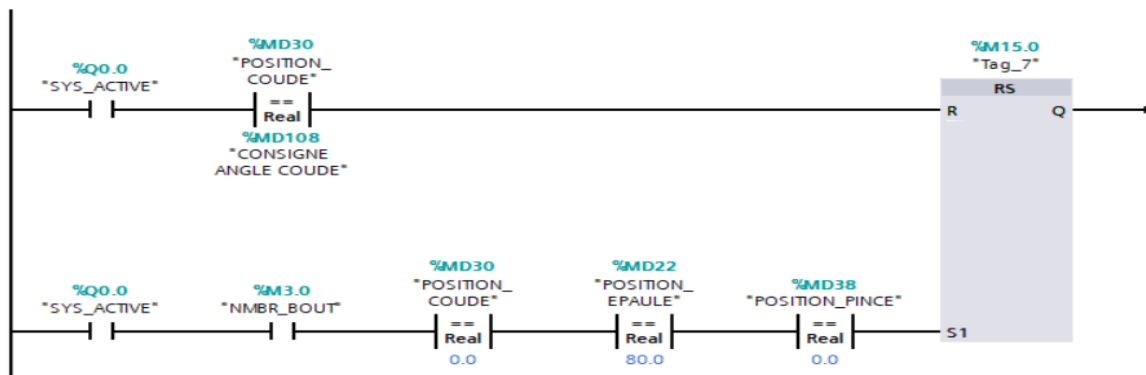
#### Réseau 1 : CODEUR\_COUDE

Commentaire



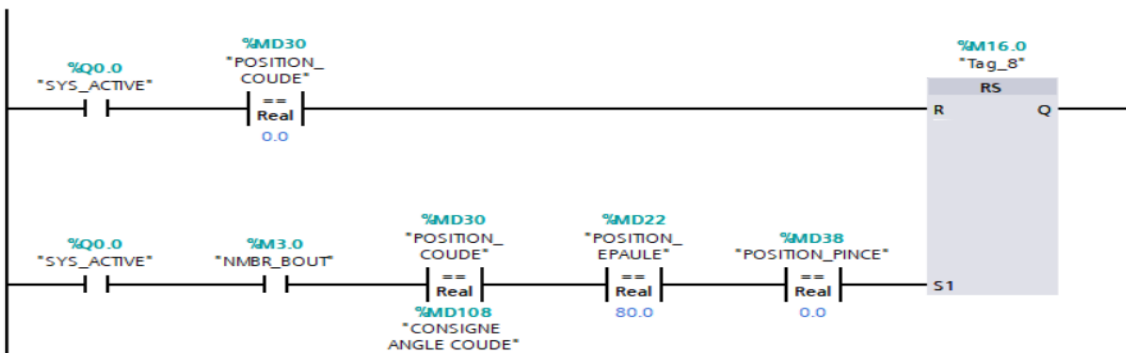
#### Réseau 3 : MARCHÉ ET ARRÊT AVANT

Commentaire



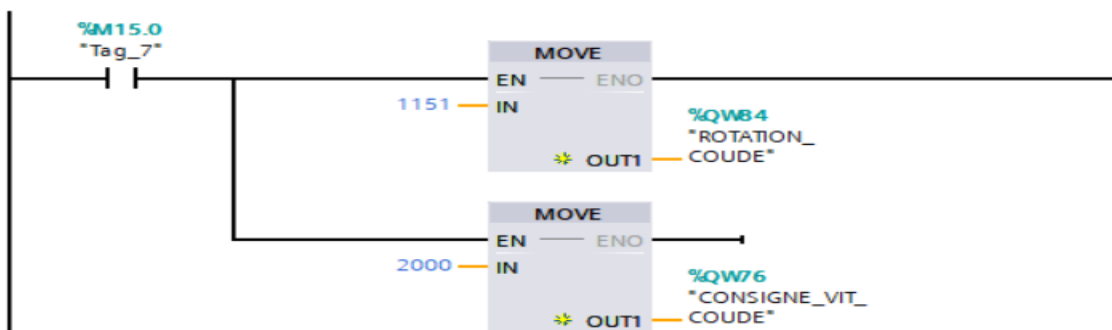
#### Réseau 4 : MARCHÉ ET ARRÊT ARRIERE

Commentaire



#### Réseau 5 : CONSIGNE DE VITESSE POUR LA MARCHÉ A GAUCHE

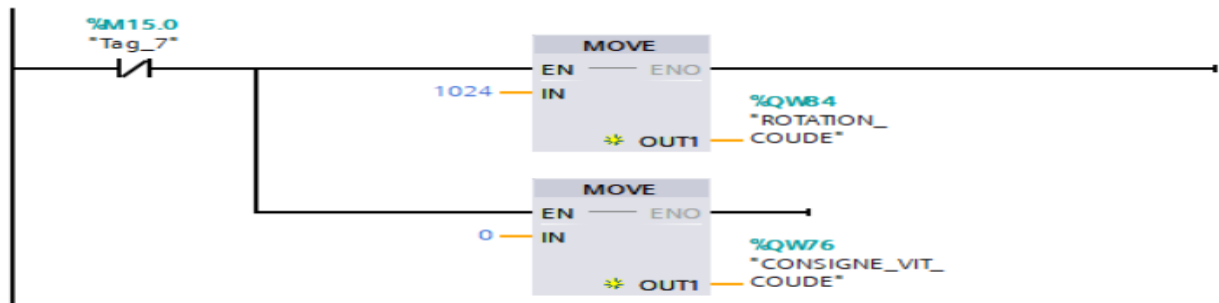
Commentaire



## Annexe B

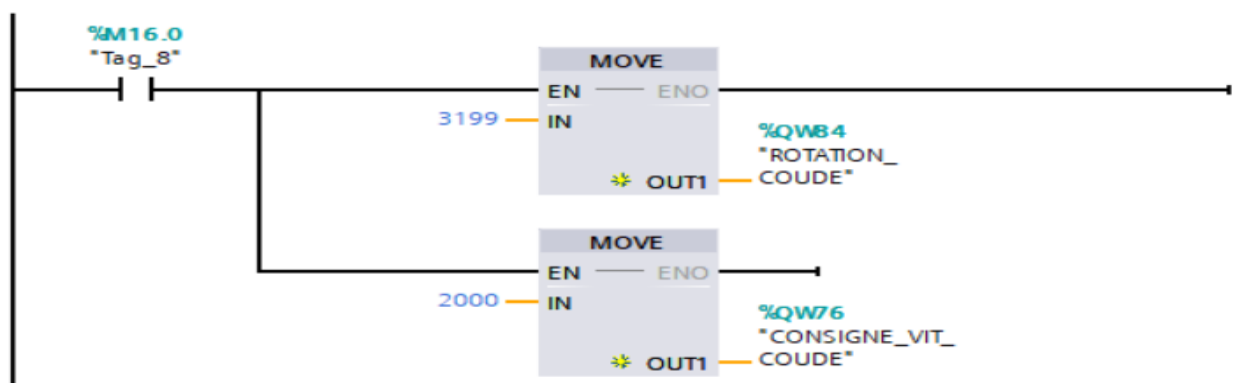
### Réseau 6 : CONSIGNE DE VITESSE POUR L'ARRET (A GAUCHE)

Commentaire



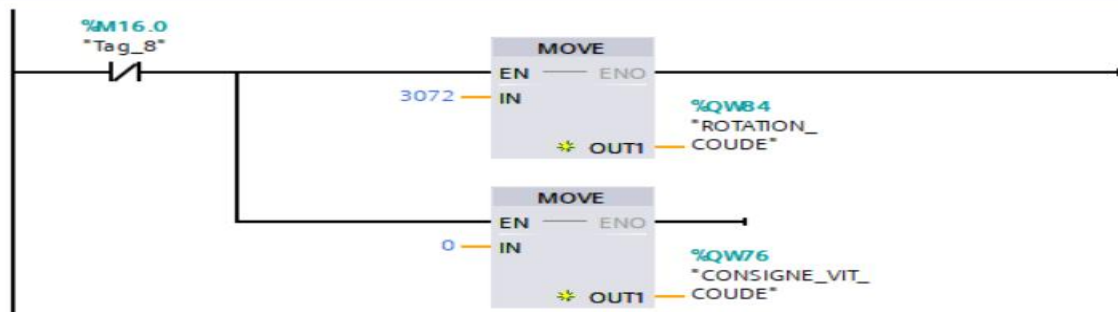
### Réseau 7 : CONSIGNE DE VITESSE POUR LA MARCHÉ A DROITE

Commentaire



### Réseau 8 : CONSIGNE DE VITESSE POUR L'ARRET(A DROITE)

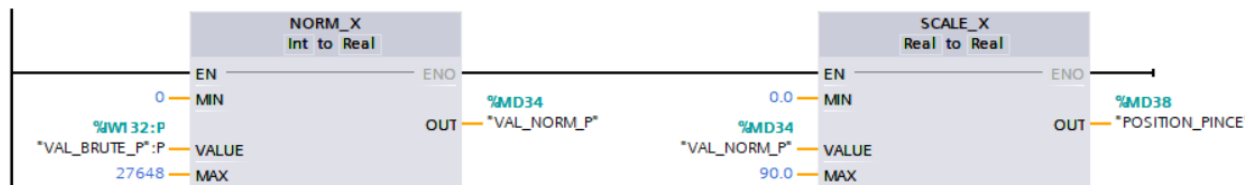
Commentaire



## Moteur pince :

### Réseau 1 : CODEUR\_PINCE

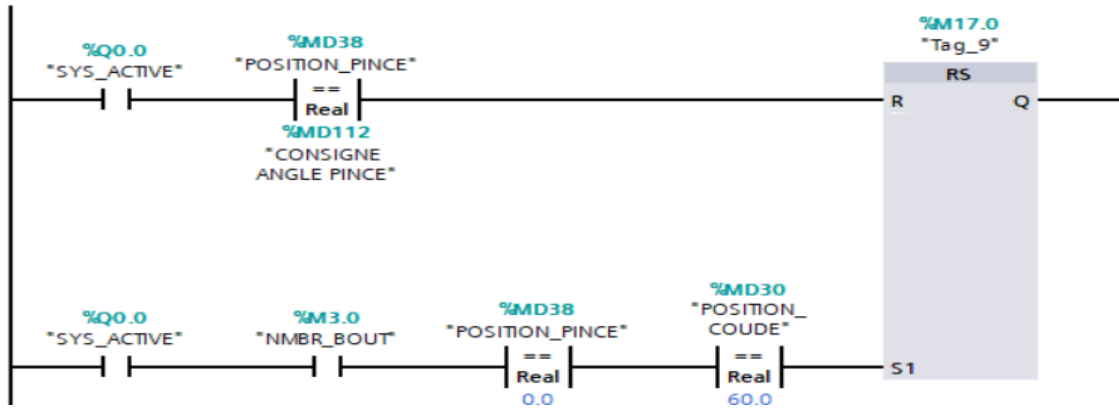
Commentaire



## Annexe B

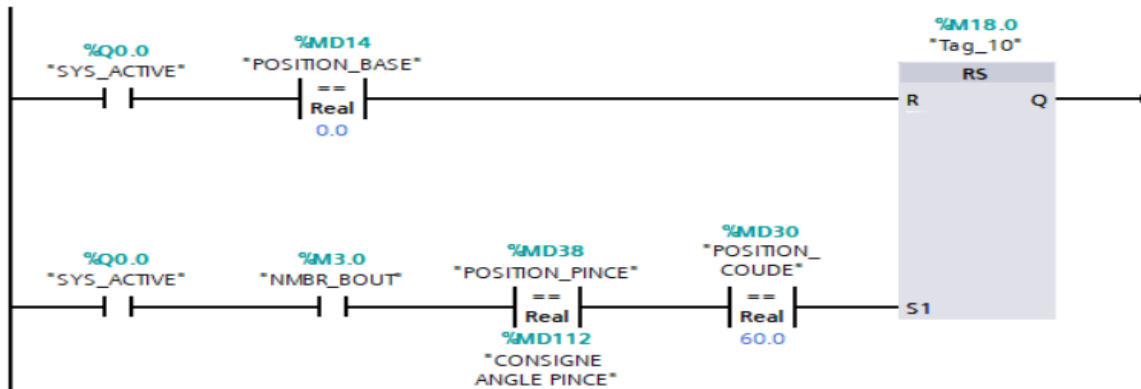
### Réseau 2 : MARCHÉ ET ARRÊT AVANT

Commentaire



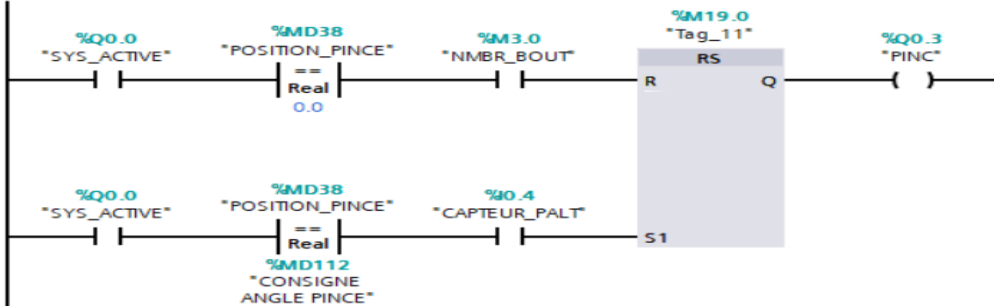
### Réseau 3 : MARCHÉ ET ARRÊT ARRIERE

Commentaire



### Réseau 4 : COMMANDE DE LA PINCE

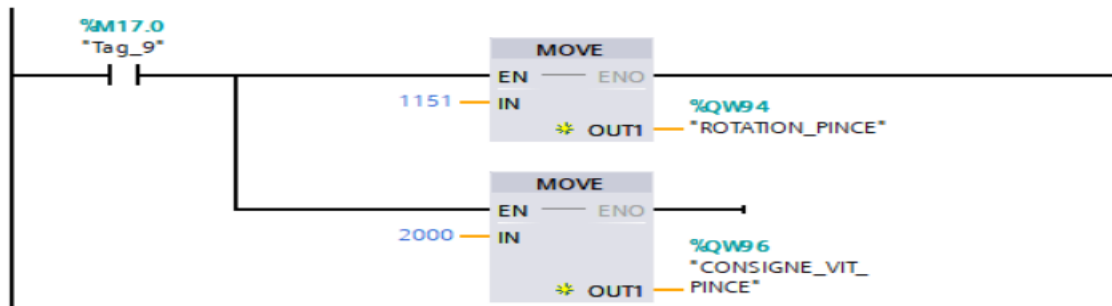
Commentaire



## Annexe B

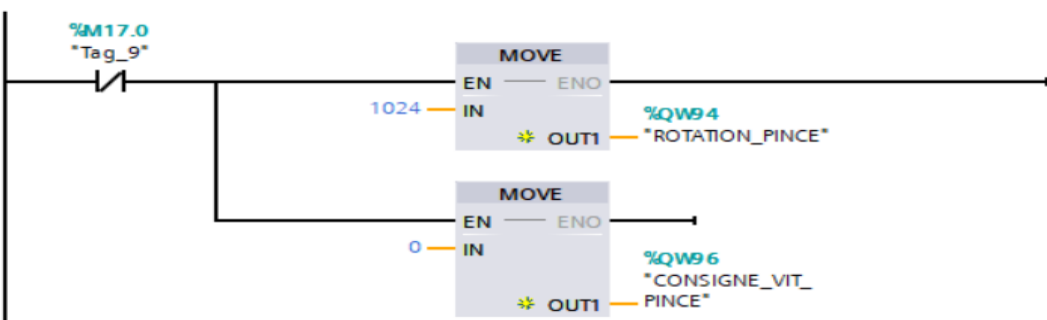
### Réseau 5 : CONSIGNE DE VITESSE POUR LA MARCHÉ A GAUCHE

Commentaire



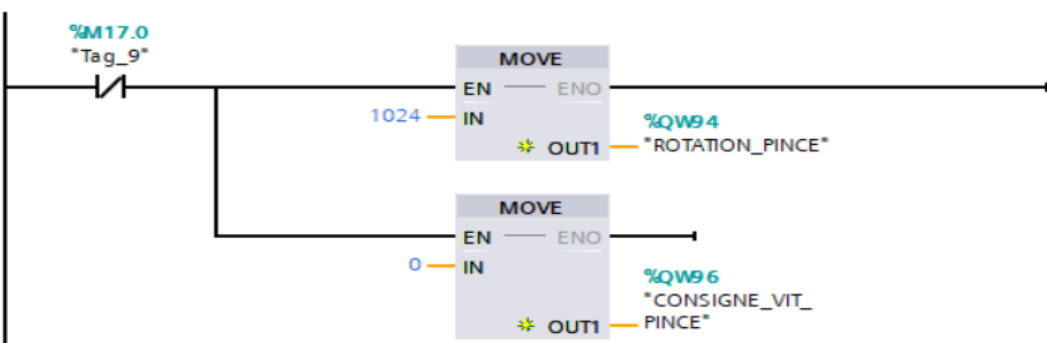
### Réseau 6 : CONSIGNE DE VITESSE POUR L'ARRET (A GAUCHE)

Commentaire



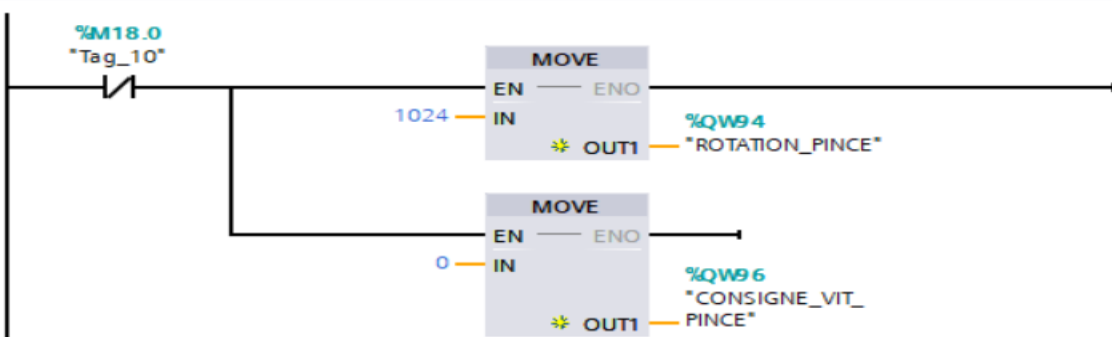
### Réseau 6 : CONSIGNE DE VITESSE POUR L'ARRET (A GAUCHE)

Commentaire



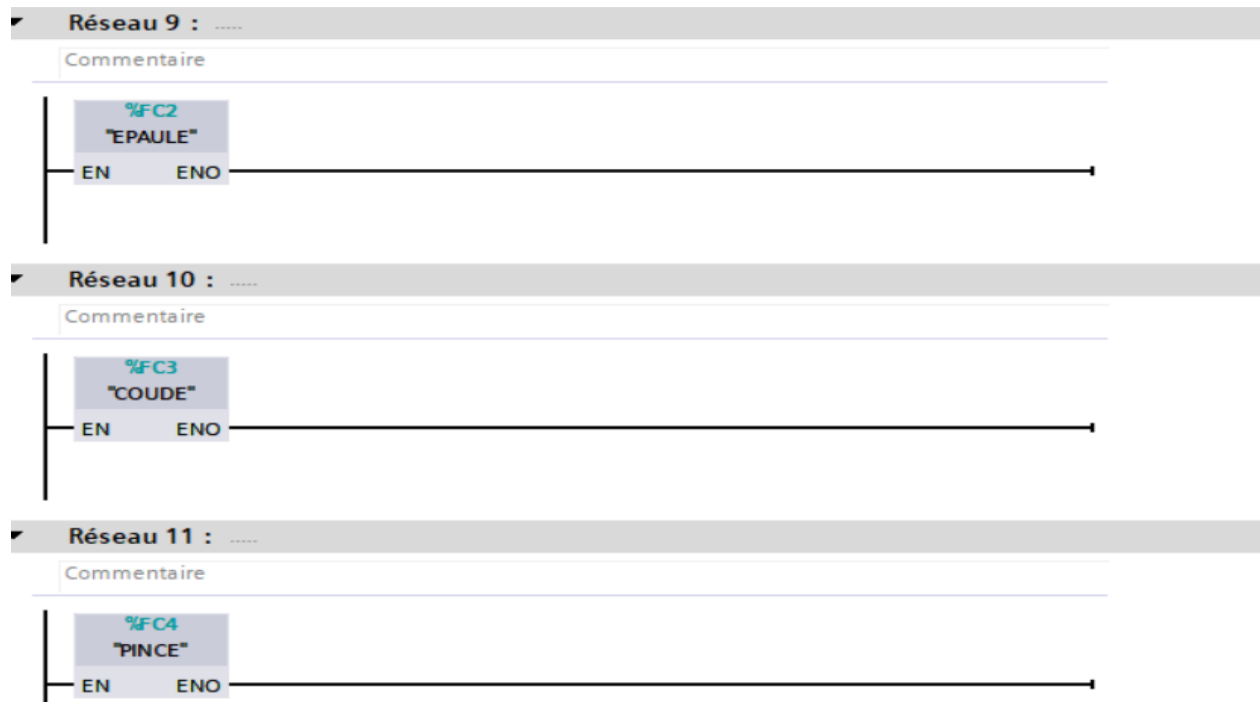
### Réseau 8 : CONSIGNE DE VITESSE POUR L'ARRET A DROITE

Commentaire

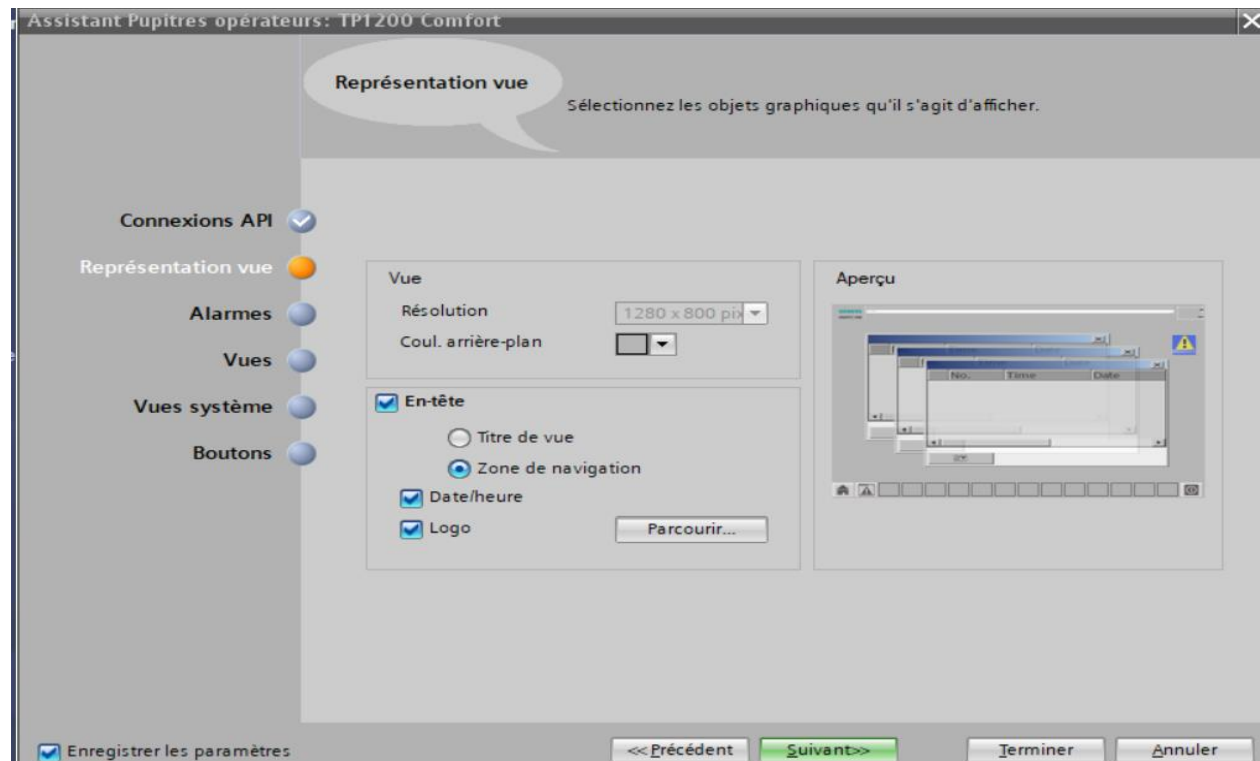


## Annexe B

### Appel aux fonctions :

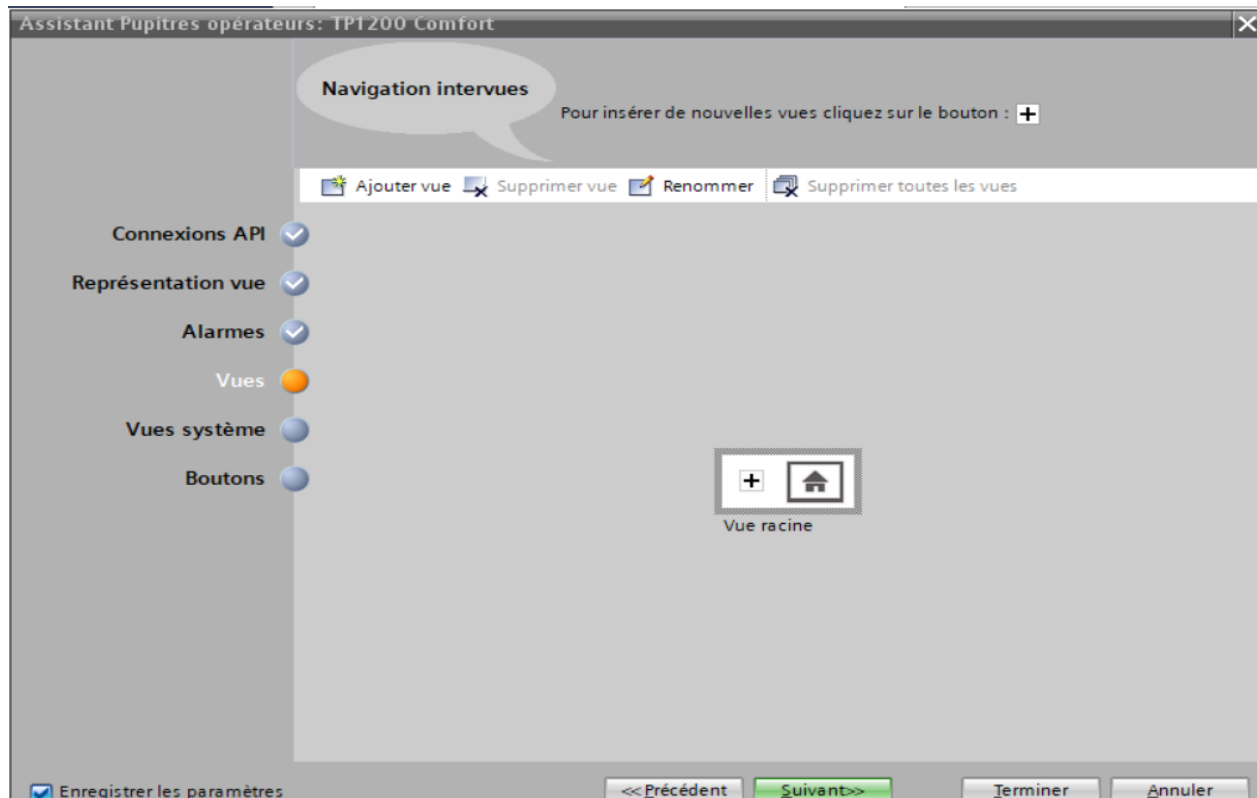
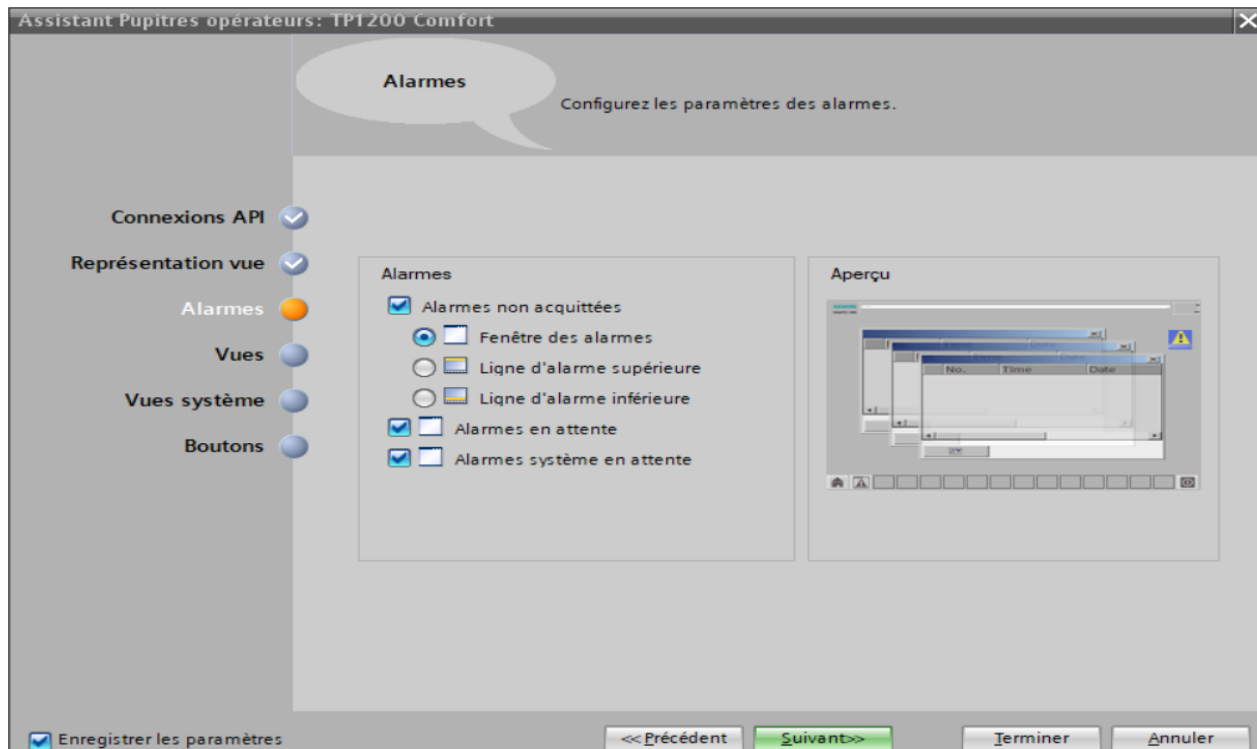


### B.5 Configuration de la vue IHM

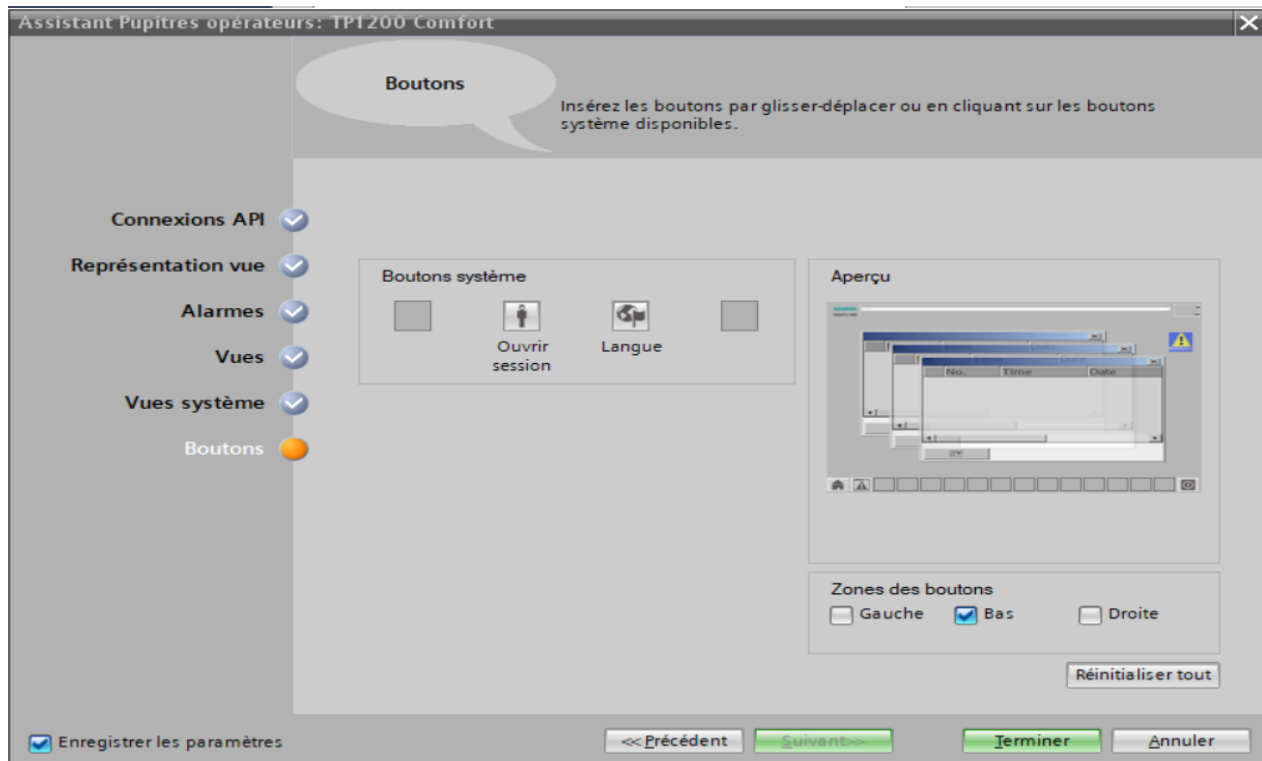
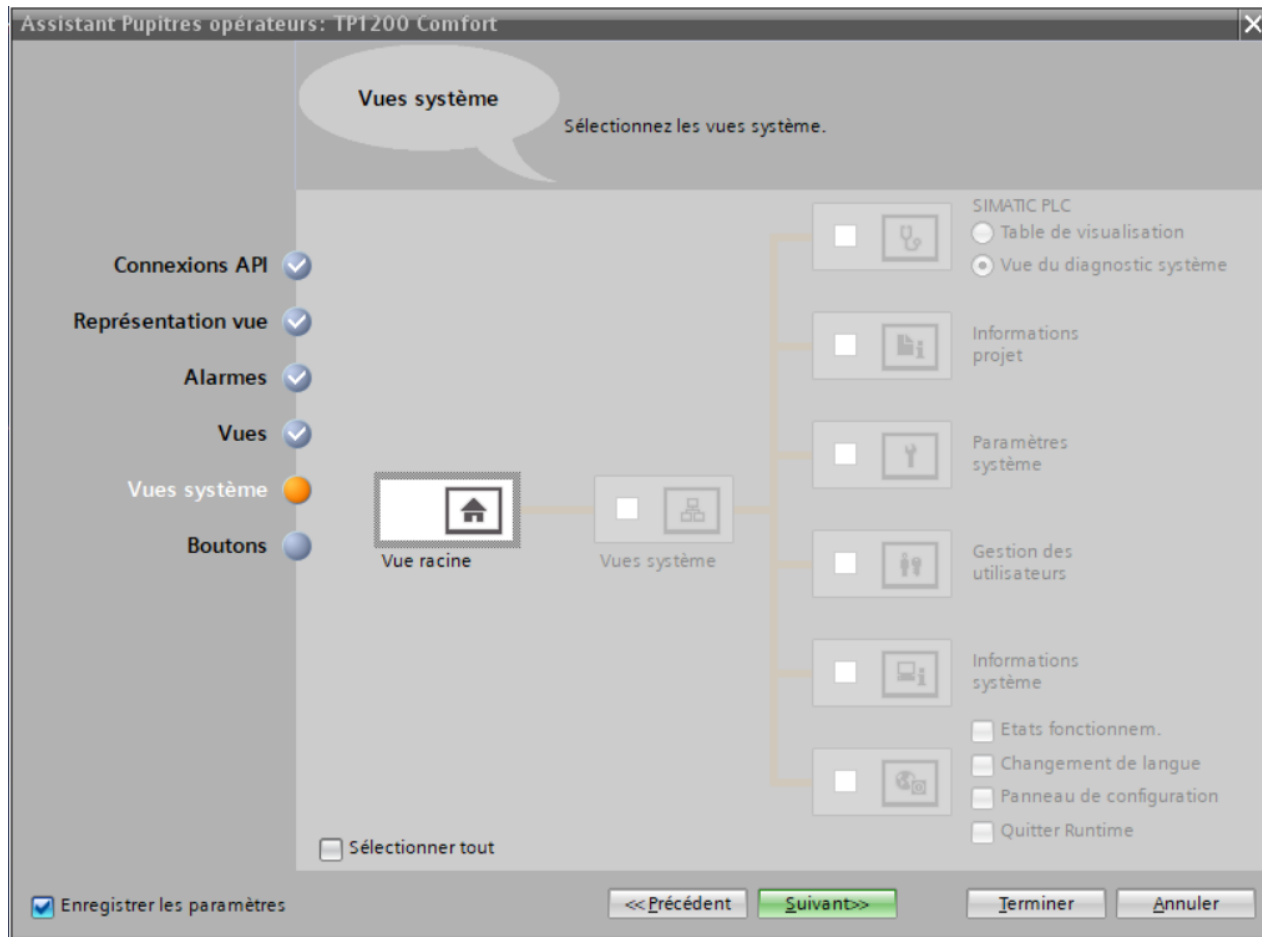




## Annexe B



## Annexe B



**Résumé :** Ce travail porte sur l'automatisation d'une opération de palettisation dans la chaîne de production d'huile de l'entreprise CEVITAL, à l'aide d'un bras robotisé à trois degrés de liberté. Après un aperçu sur les systèmes automatisés, une présentation générale des bras manipulateurs a été donnée. Nous avons étudié leur modélisation géométrique, cinématique et dynamique, ainsi que la planification de trajectoires. Le modèle a été validé par simulation sous MATLAB. La commande a été implémentée sur un automate S7-1200 via TIA Portal V15, testée sur PLCSim, et le bras robotisé est piloté par des variateurs communiquant en PROFINET. Une IHM a été développée pour superviser et contrôler le processus.

**Mots clés :** bras robotisé, palettisation, planification de trajectoires, MATLAB, commande, TIA PORTAL, automate S7-1200.

**Abstract:** This work deals with the automation of a palletising stage in the oil production line at CEVITAL, using a robot arm with three degrees of freedom. After an overview of automated systems, a general presentation of manipulator arms was given. The geometric, kinematic and dynamic models, as well as trajectory planning, are studied. A validation by simulation, using MATLAB, is performed. The control was implemented on an S7-1200 PLC via TIA Portal V15, and tested on PLCSim. The robot arm is controlled by PROFINET-compatible drives. An HMI has been developed to supervise and control the process.

**Key words:** robotic arm, palletising, trajectory planning, MATLAB, control, TIA PORTAL, S7-1200 PLC.

**ملخص-**تناول هذا العمل أتمتة عملية الترصيص في خط إنتاج الزيت بشركة سيفيتال، باستخدام ذراع آلي بثلاث درجات حرية. بعد تقديم لمحة عامة عن الأنظمة المؤتمتة، تم عرض تقديمي عام للأذرع الآلية. تم التطرق إلى نمذجتها الهندسية والحركية والديناميكية، بالإضافة إلى تخطيط المسارات. تم التحقق من صحة النموذج من خلال المحاكاة باستخدام برنامج خاص بالمحاكاة الرياضية  
نُفذ نظام التحكم باستخدام متحكم منطقي قابل للبرمجة وتم اختباره بواسطة برنامج محاكاة، ويُتحكم في الذراع الآلي بواسطة محركات كهربائية قابلة لتغيير السرعة، وتتواصل هذه المكونات عبر شبكة صناعية. كما تم تطوير واجهة رسومية تتيح للمستخدم الإشراف على العملية والتحكم فيها.

**الكلمات المفتاحية-** الذراع الآلي، الترصيص، تخطيط المسارات، المحاكاة، التحكم، الأتمتة، الواجهة الرسومي