

الجمهورية الجزائرية الديمقراطية الشعبية

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

وزارة التعليم العالي والبحث العلمي

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

جامعة أبي بكر بلقايد - تلمسان -

Université Aboubakr Belkaïd – Tlemcen –

Faculté de TECHNOLOGIE



MEMOIRE

Présenté pour l'obtention du **diplôme** de **MASTER**

En : Génie industriel

Spécialité : Ingénieur de production

Par : ZOUATI Fatima Hadjer

Intitulé

Pilotage et supervision d'un module d'emmagasinement à contrôle cartésien

(MOD.MCS-620/EV)

Soutenu publiquement le /06/ 2023 devant le jury composé de :

M. MELIANI Sidi Mohamed	Pr	Université de Tlemcen	Président
M. MEKAMCHA Khalid	MCB	Université de Tlemcen	Examineur
M ^{me} . KOULOUGHLI Siham	Pr	Université de Tlemcen	Examinatrice
M. GUEZZEN Amine Hakim	MCB	Université de Tlemcen	Encadrant

Année universitaire : 2022/

Dédicace

Tout d'abord, je voudrais remercier Dieu Tout-Puissant de m'avoir donné la santé, la patience et la volonté de terminer mes études. Ensuite, je voudrais saluer notre Prophète Muhammad, les prières et le salut d'Allah soient sur lui.

Je dédie ce travail insignifiant à : Personne ne pourra les récompenser des sacrifices qu'ils ont consentis pour mon éducation et mon bonheur, mes chers parents qui se sont sacrifiés pour prendre soin de moi tout au long de ma formation, ils sont mes Une source de réussite peut Dieu les bénisse.

Je tiens particulièrement à remercier mes parents, qui ont toujours cru en moi et m'ont accordé un soutien et une confiance inconditionnels.

Un merci spécial également à mes trois frères farés islam, Abde elhak et Mohamed Ayoub, qui m'ont encouragé dans mes moments les plus difficiles.

Je tiens également à remercier tous mes amis et mes proches qui ont été à mes côtés tout au long de ce travail.

Remerciement

Je veux dire merci à tous ceux qui ont aidé dans cette mémoire. Je souhaite exprimer mes remerciements en premier lieu à mon encadrant « GUEZZEN AMINE HAKIM » pour son soutien constant et ses conseils précieux ont été d'une grande aide tout au long du projet.

Votre expertise et vos commentaires m'ont beaucoup aidé.

Nous tenons à exprimer nos sincères remerciements aux membres du jury, à savoir Mme KOULOUGHLI, M. MELIANI et M. MEKAMCHA, pour avoir accepté d'examiner et de porter leur jugement sur notre humble travail.

Je veux aussi remercier mes professeurs et collègues qui m'ont encouragé et soutenu dans mes études. Vos réflexions et commentaires sont précieux pour améliorer ce travail.

Je tiens également à exprimer ma gratitude à ma famille et mes amis pour leur soutien inconditionnel durant l'intégralité de ma formation universitaire. Votre amour et votre soutien ont toujours été une source de motivation pour moi.

Enfin, je tiens à remercier tous ceux qui ont contribué à la réalisation de cette mémoire et m'ont inspiré. Merci beaucoup pour votre aide et vos encouragements.

Je suis très reconnaissant à tous ceux qui ont contribué à la réalisation de ce travail, je ne vous remercierai jamais assez. Merci beaucoup.

Table des matières

Liste des abréviations.....	0
Introduction générale.....	1
I. Chapitre 1 Développement industriel et automatisation.....	3
I.1 Introduction :	4
I.2 Généralité sur l'industrie :	4
I.2.1 Révolution industrielle :	4
I.2.1.1 Première Révolution Industrielle : (industrie 1.0)	4
I.2.1.2 Deuxième révolution industrielle : (industrie 2.0)	5
I.2.1.3 Troisième Révolution Industrielle : (industrie 3.0).....	6
I.2.1.4 Quatrième Révolution Industrielle :(l'industrie 4.0)	6
I.3 Définition de l'automatisme industriel :	7
I.4 Objectif de l'automatisation d'un système de production:	7
I.5 Niveaux de l'automatisme industriel:.....	7
I.5.1 Niveau Superviseur:	8
I.5.2 Niveau de contrôle:.....	8
I.5.3 Niveau terrain:	8
I.6 Différents types de commande sur les SAP:.....	9
I.6.1 Système automatisé combinatoire:	9
I.6.2 Système automatisé séquentiel:.....	9
I.7 Communication industrielle pour l'automatisation:.....	9
I.7.1 PROFINET:	9
I.7.2 PROFIBUS:	10
I.7.2.1 Différence entre PROFIBUS et PROFINET:.....	10
I.8 Conclusion:	10
II. Chapitre 2 Présentation du MOD MCS-620/EV.....	11
II.1 Introduction:.....	12
II.2 Présentation:	12
II.3 Capteurs dans l'instrumentation industrielle:.....	13
II.3.1 Définition du capteur:	13
II.3.2 Différents classifications des capteurs:	13
II.3.2.1 Classification selon le fonctionnement:	13
II.3.2.2 Classification des capteurs selon les mesurandes:.....	14
II.3.3 Type de détection :	14
II.3.3.1 Détection avec contact:.....	14
II.3.3.2 Détection sans contact:.....	14
II.3.4 Différentes Informations envoyées:	14
II.3.4.1 Information logique de manière binaire (1 ou 0):.....	14
II.3.4.2 Information analogique:.....	18
II.4 Actionneurs dans instrumentation industrielle:.....	22

II.4.1	Définition de l'Actionneur:.....	22
II.5	API (Automates programmables industriels):	22
II.5.1	Définition d'un API:	22
II.5.2	Architecture d'un API:	22
II.5.2.1	Aspect extérieur:	22
II.5.2.2	Automate compact:.....	23
II.5.2.3	Automate modulaire:	23
II.5.2.4	Structure interne de l'API:.....	24
II.5.2.5	Rack:	24
II.5.2.6	Module d'alimentation:	24
II.5.2.7	Unité centrale (CPU):.....	24
II.6	API choisi:.....	31
II.6.1	Description de l'Automate S7-1200:	31
II.6.2	Avantages du S7-1200:.....	31
II.6.3	Choix du CPU:	32
II.6.3.1	CPU1214 C:.....	32
II.7	Description du module d'emmagasinage à contrôle cartésien (MOD.MCS-620/EV):	32
II.7.1	Composants du module :	33
II.7.1.1	Deux codeurs linéaires:	35
II.7.1.2	Moteurs:.....	35
II.7.1.3	Box d'interface:	36
II.8	Fonctionnement de notre système:	37
II.9	Conclusion:	38
III.	Chapitre 3 Programmation du MOD.MCS-620/EV.....	39
III.1	Introduction:	40
III.2	Principe de la logique programmée:.....	40
III.2.1	Logique :.....	40
III.2.2	Logique programmée:.....	40
III.3	Langages de programmation de l'automate programmable industriel:.....	40
III.3.1	Langages littéraux:	40
III.3.1.1	Langage IL:.....	40
III.3.1.2	Langage Littéral Structuré (ST):.....	41
III.3.2	Langages graphiques:.....	41
III.3.2.1	Langage LD:	41
III.3.2.2	Bloc fonctionnel (FBD):.....	42
III.3.2.3	Langage SFC:.....	42
III.3.3	Langage de Programmation choisi:	43
III.3.3.1	Définition du langage LADDER:	43
III.4	Tableau des entrées et sorties: Au cours de notre travail nous avons défini le	
	tableau des entrées/sorties suivant :.....	47
III.5	Programme Ladder développé:	49
III.6	Simulation du programme:	58
III.7	Conclusion:.....	58
IV.	Chapitre 4 <i>Pilotage et supervision du MOD.MCS-620/EV</i>.....	59

IV.1	Introduction:	60
IV.2	Définition de la supervision:	60
IV.3	Fonctionnement du supervision :	60
IV.4	Modules fonctionnels d'un système de supervision:	60
IV.5	Logiciel de supervision:	60
IV.6	Simatic Step 7 TIA Portal:	60
IV.6.1	Généralité sur TIA portal:	60
IV.7	Présentation du logiciel de supervision:	61
IV.8	IHM :	61
IV.8.1	Fonctions d'IHM intégrées:	61
IV.8.2	Tableau IHM :	61
IV.8.3	IHM crée:.....	62
IV.8.4	Pc système :.....	63
IV.9	Câblage de notre système:	64
IV.10	Conclusion:	65
	<i>Conclusion générale</i>	66

Liste des figures

Figure I-1:Machine à vapeur [5]	5
Figure I-2:Ligne d'assemblage automatique [8]	5
Figure I-3: Automate Unimate [9]	6
Figure I-4: industrie 4.0 [10]	6
Figure I-5: Schéma représentent les niveaux de l'automatisme	8
Figure II-1:Module MCS-620 [17]	13
Figure II-2: Schéma de principe du fonctionnement d'un capteur	13
Figure II-3:Interrupteur de position [22]	15
Figure II-4:Capteur de proximité [22]	15
Figure II-5:Capteur de proximité inductif [23]	16
Figure II-6:Capteur de proximité capacitif [24]	16
Figure II-7:Capteur de proximité magnétique [26]	17
Figure II-8:Capteur photoélectrique [28]	17
Figure II-9:Capteur optique [29]	18
Figure II-10:Signaux analogique [31]	18
Figure II-11:Capteur d'accélération [33]	19
Figure II-12:Capteur de lumière [34]	19
Figure II-13:Capteur de pression [35]	20
Figure II-14:Capteur ultrason [37]	20
Figure II-15:Capteur thermocouple [39]	21
Figure II-16:Signaux numérique [31]	21
Figure II-17:Actionneurs [40]	22
Figure II-18:Différents Types des automates [41]	23
Figure II-19: Composants d'un API [43]	24
Figure II-20: connexion des entrées à l'état de repo [47]	25
Figure II-21: connexion des entrées à l'état de repo [47]	26
Figure II-22: connexion des sorties à l'état de repos [47]	26
Figure II-23: commande des sorties avec état actionnée [47]	26
Figure II-24: PLC Hardware [48]	28
Figure II-25: Bus entre le CPU et l'interface I/O [50]	28
Figure II-26:Fonctionnement d'un API [51]	29
Figure II-27:Interfaces des entrées et des sorties [52]	30
Figure II-28:Composants d'un S7-1200 [56]	31
Figure II-29:MOD.MCS-620/EV [17]	33
Figure II-30: Composants du MOD.MCS-620/EV	33
Figure II-31: Capteurs de fins de course	34
Figure II-32: Capteur réflecteur	34
Figure II-33: Capteurs magnétiques	35
Figure II-34: Codeurs linéaires de l'axe X et de l'axe Y	35
Figure II-35: Moteur 5M1	36
Figure II-36: Box d'interface	36
Figure II-37: Organigramme représentent le cahier de charge du mod mcs-620	38
Figure III-1: Exemple de langage IL [40]	41
Figure III-2: Exemple d'un langage ST [40]	41

Figure III-3: Exemple d'un langage à contact [40] -----	42
Figure III-4: Exemple d'un langage FBD [40] -----	42
Figure III-5: Exemple d'un langage SFC [40] -----	43
Figure III-6: Tableau des variables-----	47
Figure III-7 : Tableau des entrées et des sorties-----	48
Figure III-8 : Programme associé au bouton de mise en marche -----	49
Figure III-9: Programme associé au bouton de mise en reset-----	49
Figure III-10: Suite du programme associé au bouton de mise en marche -----	50
Figure III-11: Programme associé au bouton de mise en arrêt -----	51
Figure III-12: position du bouton 2 (P2)-----	52
Figure III-13: Nombres d'impulsions sur l'axe X -----	53
Figure III-14: Nombres des impulsions sur l'axe Y -----	53
Figure III-15: Temps de déplacements sur l'axe X -----	55
Figure III-16: Temps de déplacements sur l'axe Y -----	56
Figure III-17: Programme associé au bouton p1 avec temporisateur -----	57
Figure III-18: Simulation du programme LD-----	58
Figure IV-1: Tableau des entrées et sortie-----	61
Figure IV-2: Interface IHM du MOD/MCS -620-----	62
Figure IV-3: Simulation du bouton marche-----	62
Figure IV-4: Simulation du bouton arrêt-----	63
Figure IV-5: Création de notre interface-----	63
Figure IV-6: Connexion du Pc système et notre API-----	64
Figure IV-7: Schéma représentant le câblage de notre système-----	64
Figure IV-8: Câblage de notre système -----	65

Liste des tableaux

Tableau 1: Composants de MOD.MCS-620/EV [17]	37
Tableau 2: Principaux éléments du programme LD. [55]	43

LISTE DES ABRÉVIATIONS

<u>API</u> :	Automate programmable industriel
<u>PLC</u> :	Programmable Logic Controller
<u>DI</u> :	Digital inputs
<u>DO</u> :	Digital outputs
<u>AI</u> :	Analogie inputs
<u>AO</u> :	Analogie outputs
<u>FM</u> :	Fonction Module
<u>CP</u> :	Communication processors
<u>IL</u> :	liste d'instructions
<u>ST</u> :	Texte structuré
<u>LD</u> :	Ladder Diagramme
<u>FBD</u> :	Diagramme de blocs fonctionnels
<u>SFC</u> :	Diagramme fonctionnel séquentiel
<u>IHM</u> :	L'interface homme-machine
<u>LD</u> :	langage Ladder
<u>SCADA</u> :	Supervisory Control and Data Acquisition

Introduction générale

Les avancées technologiques numériques ont profondément changé notre monde en bousculant les modèles économiques traditionnels et créant pour les entreprises et les particuliers des nouvelles occasions. Cette évolution technologique a créé de nouvelles opportunités ouvertes et entraîné des changements majeurs dans notre société.

Le domaine de l'ingénierie s'est énormément développé au cours des dernières décennies, en grande partie grâce aux avancées de l'informatique ainsi qu'à l'électronique et à l'automatisation.

L'automatisation consiste à concevoir des systèmes capables de fonctionner de manière autonome sans qu'il y ait besoin d'une intervention humaine directe et les composants qui composent un système de production automatisé (SPA) ils s'adaptent facilement à tous les environnements industriels et gèrent de manière autonome des cycles de travail décomposés en séquences ou en phases.

Les systèmes de supervision permettent d'obtenir une vue modulaire des équipements ou des ensembles d'équipements pour visualiser leur état physique et fonctionnel et ils sont utilisés dans les salles de contrôle ou à l'intérieur des machines pour centraliser la vision physique des organes tels que les capteurs ou les moteurs qui peuvent être dispersés sur une large surface.

Les protocoles de communication industriels standard ou privés sont utilisés pour connecter les interfaces de communication nécessaires aux systèmes de contrôle tels que les API ou les machines spéciales.

Ces systèmes de supervision sont essentiels pour une gestion efficace des équipements et des processus industriels. Ils permettent aux opérateurs de contrôler et de surveiller l'état des équipements en temps réel et de prendre des décisions éclairées pour optimiser la production et la sécurité.

Pour mener à bien notre projet intitulé « Pilotage et supervision d'un module d'emménagement à contrôle cartésien », nous avons structuré notre travail en quatre chapitres distincts et Chaque chapitre aborde un aspect particulier de notre projet, allant de la démonstration et de l'exploitation de notre système à la programmation et à la supervision.

Dans la première partie de ce document, nous aborderons l'histoire de l'industrie ainsi que les différentes révolutions industrielles qui ont modifié la production et les échanges économiques, puis nous avons abordé une étude théorique sur l'automatisation.

La deuxième partie est consacrée à la présentation et à la définition de notre système industriel et sa structure, cela nous aide à comprendre notre système et son fonctionnement.

Dans la partie trois, notre attention est portée sur plusieurs programmes API et nous fournissons une description approfondie du programme que nous avons développé ainsi que sa manière d'être représenté dans le logiciel (TIA PORTAL).

Finalement, la dernière partie est dédiée spécifiquement pour la supervision et la représentation de l'interface que nous avons développée sur le logiciel (TIA PORTAL).

I. CHAPITRE 1

Développement industriel et automatisation

I.1 Introduction :

Les innovations se sont multipliées et les méthodes de production ont radicalement changé ce qui a conduit à d'autres améliorations dont beaucoup ne pouvaient être réalisées sans des progrès dans d'autres domaines à savoir la conception de la machine à vapeur qui dépassait le domaine de l'aviation, C'est la découverte de la révolution industrielle .

Presque tout de nos jours est automatisé, Cette automatisation est si rapide et précis qu'il faire un travail qui dépasse les capacités humaines dans l'industrie, donc dans ce chapitre, nous verrons tout ce qui touche à l'automatisation industrielle, du contrôle automatique des systèmes industriels à l'automatisation de ces derniers. Nous représentons également l'approche et la hiérarchie et la structure du système automatisé.

I.2 Généralité sur l'industrie :

L'industrie est définie comme un ensemble d'activités économiques qui combinent des facteurs de production (services publics, fournitures, travail et connaissances). [1]

La fonction principale de l'industrie est de fabriquer des biens tels que les articles matériaux et produits pour le marché. Elle a également un impact significatif sur la création d'emplois pour les fournisseurs. [2]

I.2.1 Révolution industrielle :

La révolution industrielle Poussée par de nombreuses innovations techniques, elle a commencé en Angleterre avant d'être en France et aux États-Unis en Allemagne et Canada. [3]

La révolution industrielle a conduit à la formation Un nouveau style de production et de performance lieu de travail d'une organisation scientifique, secteur d'activité.

I.2.1.1 Première Révolution Industrielle : (industrie 1.0)

Le principal moteur du travail était l'homme, dans cette période la révolution industrielle a remplacé le travail humain par des machines alimentées à la vapeur et à l'eau. [4]



Figure I-1:Machine à vapeur [5]

I.2.1.2 Deuxième révolution industrielle : (industrie 2.0)

Dans cette période, les gens ne dépendent plus de l'eau et de la vapeur dans les usines, mais dépendent de l'électricité, ce qui fait gagner du temps aux usines. [6]

Henry Ford a su profiter à cette époque du développement de l'industrie pour développer l'industrie automobile. [7]



Figure I-2:Ligne d'assemblage automatique [8]

I.2.1.3 Troisième Révolution Industrielle : (industrie 3.0)

L'automatisation a évolué avec l'avènement de l'électronique et de l'informatique qui ont réduit l'intervention humaine. [9]

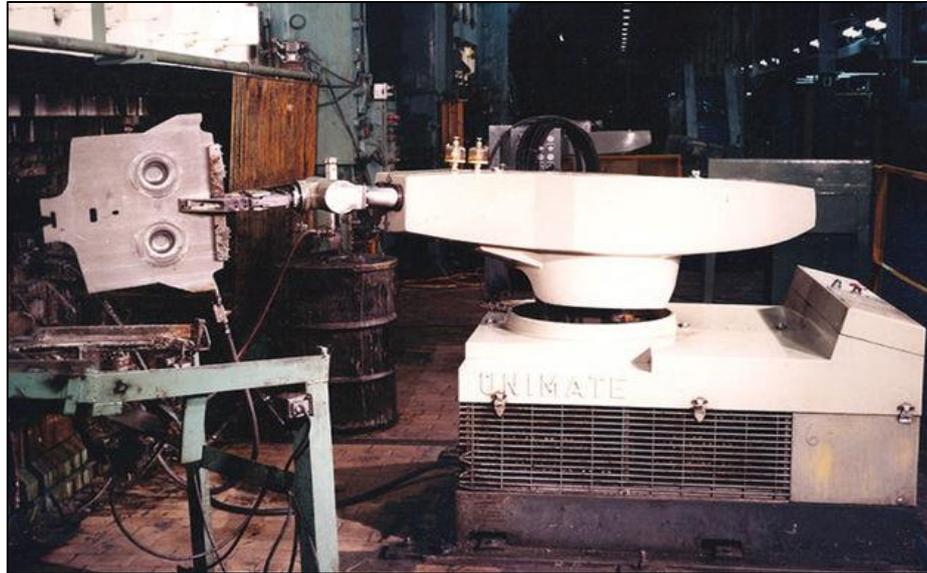


Figure I-3: Automate Unimate [9]

I.2.1.4 Quatrième Révolution Industrielle :(l'industrie 4.0)

C'est une continuation de la troisième révolution industrielle avec une foule de nouvelles technologies, Il comprend l'intelligence artificielle l'Internet des objets ou même l'impression 3D. [9]



Figure I-4: industrie 4.0 [10]

I.3 Définition de l'automatisme industriel :

C'est d'utiliser la technologie et des machines pour accomplir des tâches au sein d'un contexte industriel.[11]

Cette terminologie permet aux entreprises de réunir les machines d'elles et par ramasser aussitôt données entre celles ceci. [11]

Existe depuis des nombreuses années et son objectif principal est d'automatiser les opérations dans les secteurs industriels. [11]

Elle est utilisée dans le but d'automatiser les machines et installations pour qu'elles travaillent de manière autonome avec efficacité et un taux d'erreur faible. [11]

L'automatisation industrielle d'aujourd'hui rendue possible par une variété de technologies : le systèmes **SCADA** (utilisés pour automatiser quelques tâches), **PLC** est utilisé pour le contrôle des processus industriels et l'affectation de quelques taches, **IHM** (L'interface homme-machine facilite l'échange d'informations entre l'utilisateur et la machine peuvent être locales ou distantes), **les réseaux d'automatisation industrielle** (deux types de réseaux : de données pour envoyer les données et de commande pour transmettre des signaux de commande entre les machine). [11]

I.4 Objectif de l'automatisation d'un système de production:

Le but de l'automatisation est d'assurer l'objectif principal des différentes entreprises, visait aussi à des autres objectifs tel que :

- Réduire les temps de production globaux. [12]
- Réduire les risques professionnels liés aux opérations de fabrication dangereuses des employés. [12]
- Diminuer les coûts d'exploitation. [12]
- Renforce la disponibilité et performance. [12]

I.5 Niveaux de l'automatisme industriel:

L'automatisation est divisée en plusieurs degrés différents. La forme la plus courante « la hiérarchie en triangle » elle est constituée de trois niveaux : [11]

I.5.1 Niveau Superviseur:

Il se compose généralement d'un ordinateur industriel (PC) contenant un logiciel spécial aux fins de visualisation et de paramétrage des processus, connu sous le nom de niveau superviseur est fourni par le fournisseur pour contrôler tous les processus industriels. [11]

I.5.2 Niveau de contrôle:

A ce niveau, il trouve la mise en œuvre de tous les programmes liés à l'automatisation. Les PLC sont généralement implémentés et réalisés sur un OS propriétaire. Les automates sont également capables d'interagir avec de nombreux dispositifs d'entrée/sortie. [11]

I.5.3 Niveau terrain:

A ce niveau se trouvent les équipements périphériques tels que les capteurs (température, optique, pression, etc.) et les actionneurs (actionneurs, vannes, interrupteurs, etc.). Il est rattaché aux PLC. L'automate peut également diagnostiquer et affecter divers composants grâce à la communication entre les réseaux filaires et sans fil. [11]

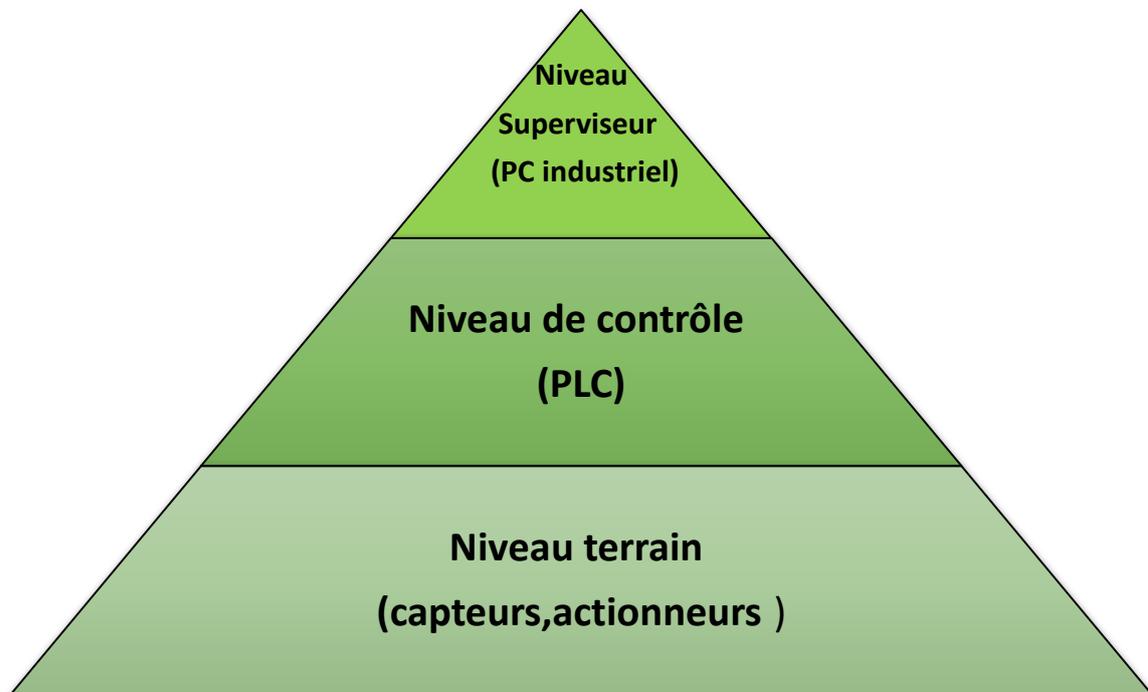


Figure I-5: Schéma représentant les niveaux de l'automatisation

I.6 Différents types de commande sur les SAP:

Dans les systèmes automatisés de production, ils existent différents types:

I.6.1 Système automatisé combinatoire:

Ces systèmes n'utilisent pas de mémoire et l'ensemble des données d'entrée font référence à une seule combinaison des sorties, logique combinatoire c'est la logique associée, l'algèbre de Boole, les tables de vérité, sont parmi les outils utiliser pour les concevoir. [11]

I.6.2 Système automatisé séquentiel:

Dans ce système, un état d'entrée peut correspondre à plusieurs états de sortie logique séquentielle, c'est la logique associée. Elle peut être : Avec la logique câblée, c'est à dire la commande pneumatique ou électrique ou avec logique programmée, c'est la commande électronique. [11]

I.7 Communication industrielle pour l'automatisation:

Les réseaux de communication dédiés aux applications industrielles nous permettent de transmettre des données du niveau terrain au niveau superviseur [13]

I.7.1 PROFINET:

C'est une norme de communication ouvert pour l'automatisation industrielle, c'est le bon choix pour automatiser toutes les installations grâce à sa connexion continue basée sur Ethernet, il répond aux exigences les plus diverses et il réduit le modèle OSI à 7 couches à un modèle TCP/IP à 4 couches. Les trois niveaux de performance, dans lesquels la communication avec PROFINET est évolutive sont: [14]

- Les données géométriques sont transmises via TCP/IP. [14]
- Le processus de transmission de données est effectué sur un canal en temps réel (RT). [14]
- On utilise la communication isochrone pour les échanges en temps réel. (IRT) Pour les applications synchronisées (Motion Control). [14]

I.7.2 PROFIBUS:

Le PROFIBUS est un système de transmission de données numériques ouvert et standardisé pour tous les domaines d'application de l'automatisation, de la fabrication et des processus. [15]

I.7.2.1 Différence entre PROFIBUS et PROFINET:

Le Profibus et le Profinet sont deux protocoles de communication introduits par les systèmes de contrôle industriels de Siemens., il y a une grande différence entre les deux protocoles. Ils utilisent des câbles et des connecteurs différents. [16]

❖ PROFINET:

- ✓ Les câbles Profinet sont verts. [16]
- ✓ Le Profinet est un protocole de communication industrielle basé sur Ethernet. [16]
- ✓ Il fonctionne à 100 Mbps. [16]
- ✓ Il est devenu le protocole de communication préféré des applications industrielles en raison de sa vitesse élevée et de sa plus grande flexibilité. [16]

❖ PROFIBUS:

- ✓ Les câbles Profibus sont violets. [16]
- ✓ Les réseaux Profibus fonctionnent à des vitesses allant de 9600 bps à 12 Mbps. [16]

I.8 Conclusion:

Ce chapitre aborde des généralités sur la révolution industrielle pour initier le lecteur au monde de l'industrie et de son évolution et ainsi qu'une étude théorique sur l'automatisation. Par la suite, nous avons défini le concept d'automatisation industrielle et ses niveaux.

Après l'introduction qui parle sur le domaine de l'automatisation, dans le chapitre suivant, nous parlerons du fonctionnement de notre système.

II. CHAPITRE 2

Présentation du MOD MCS- 620/EV

II.1 Introduction:

L'automate programmable industriel ou l'API appartient à la catégorie des machines série. Il présente des avancées du langage de programmation destinées à l'ingénieur en automatisation. Il peut être changé facilement et il dispose de capacités de simulation et de visualisation qui offrent à l'utilisateur une assistance efficace. L'automate nous donne la possibilité de connecter à l'extérieur avec un ordinateur ou d'autre API, il améliore les conditions de travail et il augmente la production pour améliorer la productivité et pour l'élimination du travail répétitif.

Parmi les composants d'un système industriel, nous trouvons les capteurs et les actionneurs. Dans ce chapitre, nous exposons dans un premier temps une présentation sur les actionneurs et les capteurs, ensuite nous présentons une étude approfondie sur les automates programmables industriels. A la fin de ce chapitre nous définissons notre système puis sa structure pour finir par son principe de fonctionnement.

II.2 Présentation:

Le MCS-620 /EV est un magasin de stockage vertical à 3 niveaux et 12 emplacements. Sa fonction principale est d'emmagasiner les pièces selon le type de matériau dont elles sont constituées, le système de contrôle cartésien est responsable du déplacement sur les trois axes (X, Y, et Z). IL permet la collecte ou l'emplacement des pièces sur trois plans différents. Le magasin comporte douze positions au total, il existe quatre positions dans chaque niveau. La détermination des positions libres et occupées se fait à travers un capteur situé sur la pince. Les déplacements le long des axes X et Y sont réalisés par deux petits moteurs. Le déplacement le long de l'axe Z est réalisé par un vérin couplé à une pince.

On peut utiliser le module à la fois comme magasin de pièces de rechange et comme magasin de pièces brutes et finies, Il est fixé à un cadre en aluminium qui peut être déplacé facilement à l'intérieur dans un espace de travail.

Le contrôle complet de notre système MCS-620/EV est mis en œuvre par un automate connecté au système via un boîtier d'interface Entrées et Sorties.

Grâce au logiciel de programmation dédié à l'automate industriel, il est possible de faire une grande variété d'expériences inhérente sous forme de module d'automatisation MCS-620/EV, qui peut être intégré dans un cours d'automatisation. [17]



Figure II-1: Module MCS-620 [17]

II.3 Capteurs dans l'instrumentation industrielle:

II.3.1 Définition du capteur:

Le rôle fondamental d'un capteur est de transformer l'état d'une grandeur physique appelée "mesurande" (m) en une grandeur exploitable, appelée "réponse" (s). [18]

Par exemple : tension électrique ou hauteur de mercure. [18]

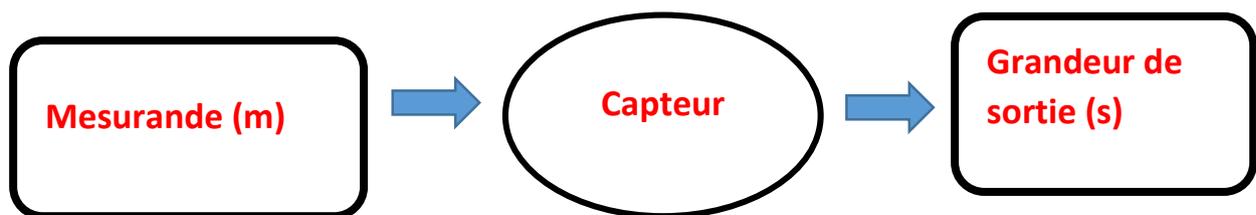


Figure II-2: Schéma de principe du fonctionnement d'un capteur

II.3.2 Différents classifications des capteurs:

Selon les propriétés électriques de la grandeur de sortie, on distingue deux grands groupes différents type de capteurs: le groupe des capteurs passifs et groupe des capteurs actifs. [18]

II.3.2.1 Classification selon le fonctionnement:

On distingue deux catégories de capteurs.

➤ **Capteurs actifs:**

Un capteur actif est généralement se base sur un effet qui permet de convertir l'énergie spécifique de la grandeur physique mesurée en une forme d'énergie utilisable : énergie thermique, mécanique ou radiative. [18]

Il s'agit de capteurs que l'on pourrait modéliser par des générateurs comme dans les systèmes électromagnétiques. [18]

➤ **Capteurs passifs:**

Se définissent comme des capteurs modélisables par une impédance qui peut être une résistance, une inductance ou une capacité. Dont l'un des paramètres clés est influencé par la grandeur mesurée. [18]

II.3.2.2 Classification des capteurs selon les mesurandes:

- **Bio ou chimique** : humidité, détection de gaz ... [20]
- **Mécanique** : force, déplacement, pression, vitesse, accélération, pression ... [20]
- **Électrique** : charge, impédance, courant ... [20]
- **Thermique** : flux thermique, température ... [20]
- **Magnétique** : perméabilité, champ magnétique ... [20]

II.3.3 Type de détection :

II.3.3.1 Détection avec contact:

Dans ce type de capteur, il est nécessaire d'avoir un contact physique avec le corps pour le détecter, exemple : la plupart des capteurs de fin de course... [21]

II.3.3.2 Détection sans contact:

En revanche, ce type de capteur est capable de détecter des phénomènes à proximité sans nécessiter de contact physique. [21]

II.3.4 Différentes Informations envoyées:

II.3.4.1 Information logique de manière binaire (1 ou 0):

Il s'agit du capteur TOR (tout ou rien), On l'appelle aussi "capteurs logiques", dans tout le domaine industriel des capteurs de type tout ou rien utilisé pour la détection de la présence ou l'absence d'un état ou d'un objet, il remonte donc 0 ou 1 selon le paramétrage défini. [19]

On peut classer ce type de capteur (TOR) en 4 familles :

➤ **Capteurs mécaniques (interrupteurs de position) :** interrupteurs de fin de course

Il est situé dans la partie opérative ils se chargent de transmettre au système de traitement les informations relatives à la présence et à la position ou fin de course.[21]

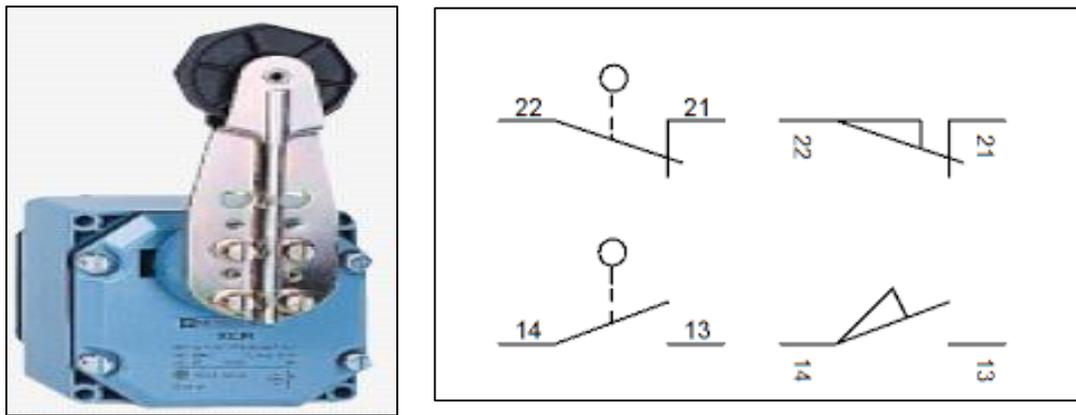


Figure II-3:Interrupteur de position [22]

➤ **Capteurs de proximité:**

Ils sont indispensables pour détecter les objets sans avoir besoin d'un contact physique direct et il détecte tout objet magnétique proche de la tête de détection, Ils sont généralement placés sur les vérins pour obtenir la position du piston. [21]



Figure II-4:Capteur de proximité [22]

➤ **Capteurs de proximité inductifs:**

Ce type de capteur est utilisés pour la détection sans contact physique à une distance proche du métal, Ils sont constitués d'un oscillateur dont les bobines forment la face sensible et fonctionnent à l'aide d'un champ magnétique . [21]



Figure II-5:Capteur de proximité inductif [23]

➤ **Capteurs de proximité capacitifs:**

Ils permettent de détecter des objets de toutes sortes. Ils comportent un oscillateur et ils dépendent de la modification du champ électrique lorsqu'un objet s'approche. [21]



Figure II-6:Capteur de proximité capacitif [24]

➤ **Capteurs de proximité magnétiques:**

Ce genre de capteur est utilisé pour la détection de position sans contact. Ils détectent les pièces et les objets en utilisant une propriété magnétique. Ils sont précisément conçus pour marcher et fonctionner avec des aimants. [25]



Figure II-7: Capteur de proximité magnétique [26]

➤ **Capteurs photoélectriques:**

Ils sont utilisés pour la détection des cibles. Cela peut être une chose ou une personne, la détection se fera par un faisceau de lumière. ILS se composent principalement d'un émetteur et un récepteur de lumière. [21]



Figure II-8: Capteur photoélectrique [28]

➤ **Capteurs optiques:**

Son rôle est de détecter l'intensité de la lumière et de la convertir en tension ou en courant, Ils sont efficaces même pour la détection de la température et des couleurs. [28]



Figure II-9:Capteur optique [29]

II.3.4.2 Information analogique:

Elle évolue constamment dans le temps dans une plage de variation. Elle a un nombre infini de valeurs, par exemple la température et la tension ou la pression sont analogiques. [30]

➤ Signal analogique:

Nous disons qu'un signal est analogique si le signal continu (DC) qu'il représente il peut prendre une multitude de valeurs dans une période de temps donnée, ce signal peut avoir différentes caractéristiques telles qu'un niveau et une forme, aussi une fréquence telle qu'illustré dans la figure ci-dessous. [31]

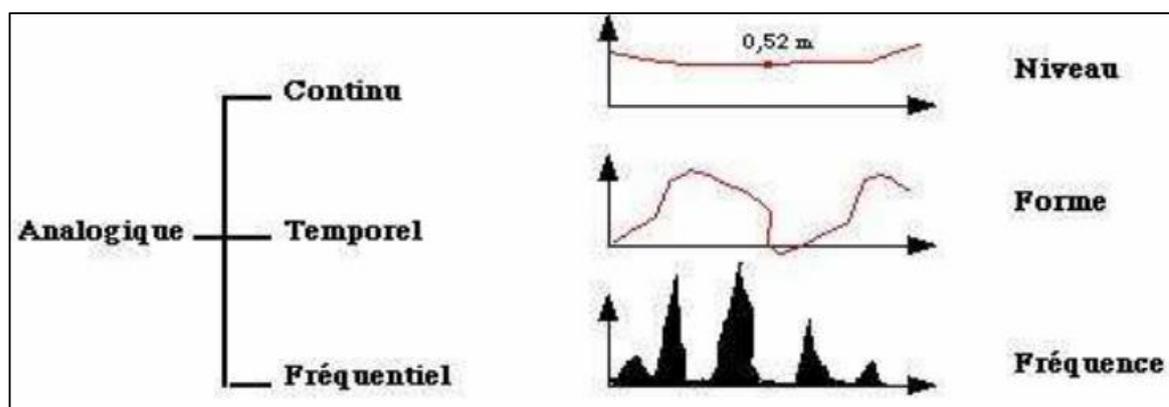


Figure II-10:Signaux analogique [31]

Donc le signal transmis peut être l'un de ces trois signaux :

- **Continu** : c'est un signal qui change petit à petit avec le temps. [31]
- **Fréquentiel** : Il s'agit du spectre de fréquence qui tient toutes informations requises. [31]
- **Temporel** : c'est la configuration que peut avoir un signal représenté au fil du temps. [31]

➤ **Capteurs analogiques:**

Ils ont pour fonction de transformer une grandeur physique en un signal qui peut être compris par le système de contrôle. [31]

➤ **Capteurs d'accélération:**

Appelé aussi accéléromètre, sa fonction principale est de détecter les variations de position et de vitesse. Il détectant aussi le mouvement. [32]



Figure II-11: Capteur d'accélération [33]

➤ **Capteurs de lumière:**

C'est un capteur électrique qui détecte l'intensité de la lumière. [34]

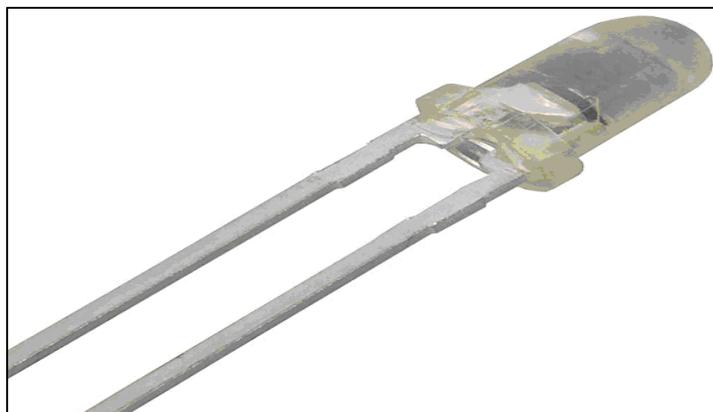


Figure II-12: Capteur de lumière [34]

➤ **Capteurs de pressions:**

Son rôle principal est de mesurer la pression effective exercée à certains composants et ils sont constitués d'un élément sensible à la pression. [35]



Figure II-13:Capteur de pression [35]

➤ **Capteurs ultrason:**

C'est dispositif électronique qui utilise les fréquences ultrasonique pour détecter la présence et mesurer les distances entre toutes sortes d'objets. Pour son processus de calcul, nous avons besoin du temps nécessaire pour la réception des ondes ultrasonique envoyées. Les caractéristiques de ces capteurs leur permettent d'être robustes et de rester fiables. [36]



Figure II-14:Capteur ultrason [37]

➤ **Capteurs thermocouple:**

Sa tâche principale est de déterminer la température, il existe plusieurs catégories de ces capteurs qui varient en fonction du matériau de fabrication de sous forme d'alliages ou de pureté et chacun ayant ses propres caractéristiques et applications. [38]



Figure II-15:Capteur thermocouple [39]

➤ **Information numérique:**

Il s'agit d'informations binaires (1 ou 0) utilisé pour coder les informations binaires est appelé un bit. [31]

La sortie du capteur qu'il mesurera donnera n'importe laquelle des informations de type :

« **TOR** » (**tout ou rien**) : Il indique l'état d'un système qui ne peut avoir que deux états fixes soit 0 ou 1. [31]

Train d'impulsions : Dans ce type nous constatons que chaque changement d'état est représenté par une impulsion. [31]

Échantillonnage : Le signal analogique est converti en une version numérique qui est représentée comme une forme d'escalier. [31]

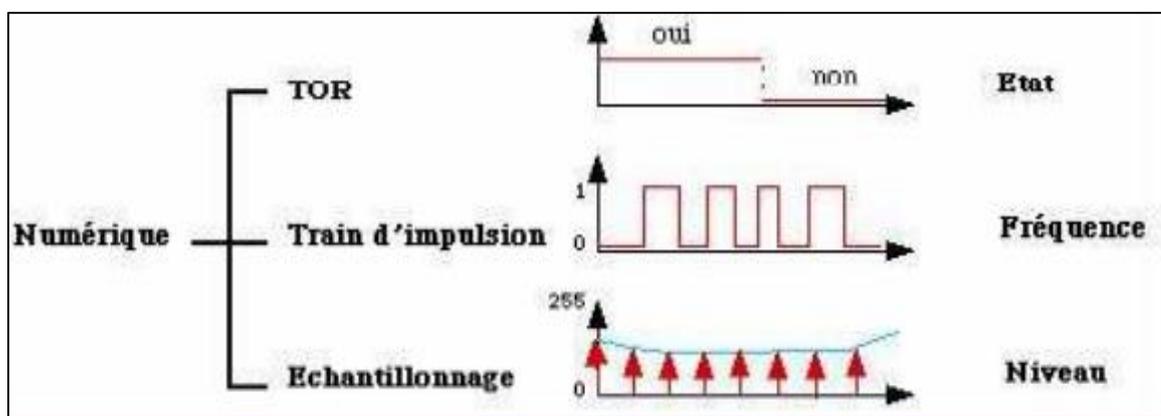


Figure II-16:Signaux numérique [31]

II.4 Actionneurs dans instrumentation industrielle:

II.4.1 Définition de l'Actionneur:

C'est un objet technique qui convertit une information numérique en un phénomène physique, Il peut modifier le comportement ou l'état du système, et peut également déclencher des alarmes ou des interrupteurs. [40]

Il peut être comme : (voir la figure ci-dessous)

-**Moteurs** : soit un moteur asynchrone ou pas à pas ou bien linéaires

-**Vérins** : soit pneumatique ou bien hydraulique ;

-**Pompes ou Vannes.**

-**Effecteur** : Il est défini comme un élément qui reçoit d'énergie de l'actionneur réalisé une tâche dans la partie opérative. [40]

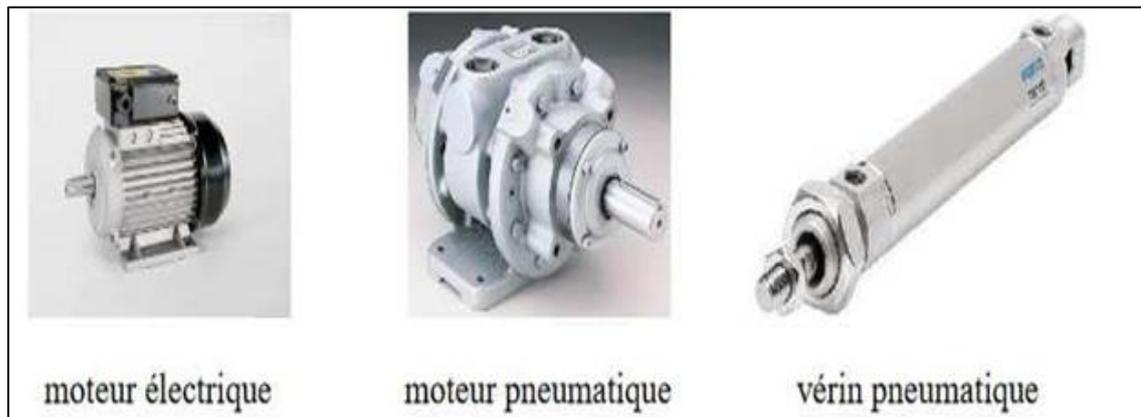


Figure II-17:Actionneurs [40]

II.5 API (Automates programmables industriels):

II.5.1 Définition d'un API:

L'API ou PLC en anglais (contrôleur logique programmable), Il se définit comme une interface logicielle qu'il facilite l'échange de données entre diverses applications et la communication mutuelle entre elles ou des services .C'est un type spécial de contrôleur à base de microprocesseur. [41]

II.5.2 Architecture d'un API:

II.5.2.1 Aspect extérieur:

Ces automates sont principalement disponibles sous plusieurs forme tels que : boîtier ou en version rack ou modulaire. [41]

L'API peut être trouvée en deux types, compact ou modulaire.

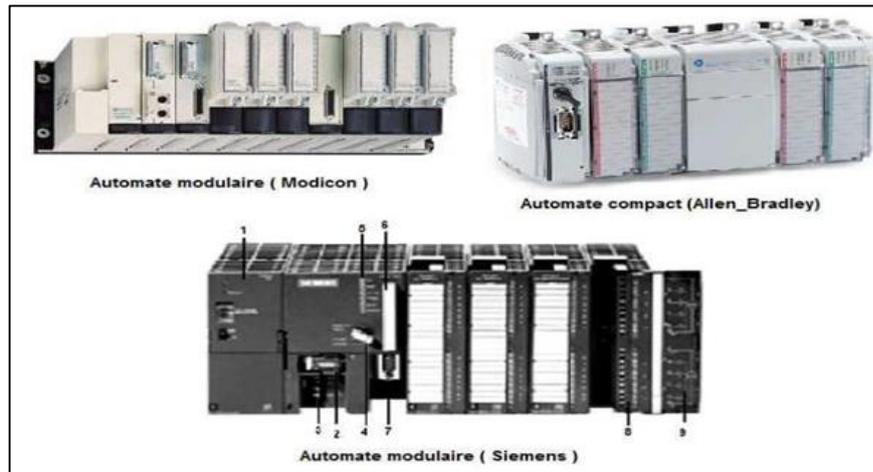


Figure II-18: Différents Types des automates [41]

II.5.2.2 Automate compact:

Ce type d'automate compact peut avoir six, huit et jusqu'à 24 entrées et quatre à seize sorties et une seule mémoire capable de stocker un nombre d'instructions entre 300 et 1000. Il est conçu pour l'affectation des certaines fonctions comme le comptage rapide, entrée et sortie analogique, etc... et recevoir des extensions limitées. [40]

II.5.2.3 Automate modulaire:

Un API modulaire intégré pour les automatisations complexes nécessitant la puissance et il peut être compatible avec toutes les tailles d'automates et des unités fonctionnelles variées. [40]

Il renferme ces éléments : [40]

- Modules de puissance. [40]
- batterie de rechange. [40]
- Connecter à 24Vdc. [40]
- Commutateur de mode (verrouillé avec clé). [40]
- Les LED indiquent l'état et les défauts. [40]
- Carte mémoire. [40]
- Interface multipoint (MPI). [40]
- Rabat avant. [40]

II.5.2.4 Structure interne de l'API:

La composition de l'API repose sur divers éléments fondamentaux, tels qu'illustré dans la figure ci-dessous :

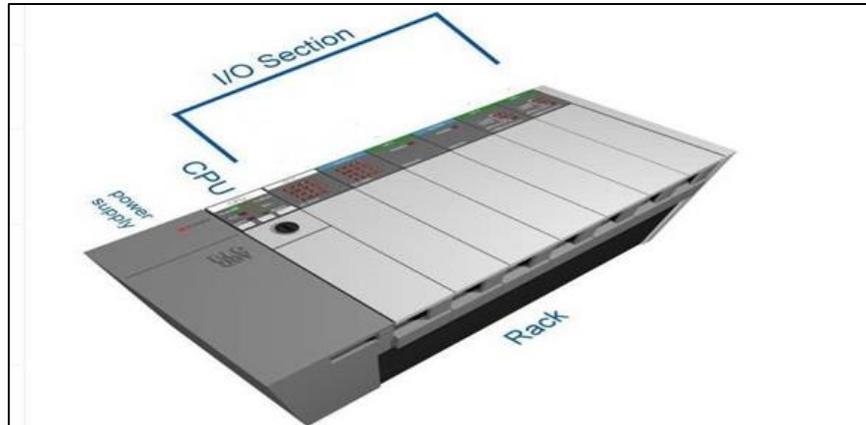


Figure II-19: Composants d'un API [43]

II.5.2.5 Rack:

C'est l'élément central du système de contrôle qui regroupe les différents composants de l'automate. [40]

II.5.2.6 Module d'alimentation:

Il permet de répartir l'énergie aux différentes unités. [40]

II.5.2.7 Unité centrale (CPU):

Il est doté des composants suivants : un microprocesseur et une puce mémoire, accompagnés d'autres circuits intégrés pour le contrôle logique, la surveillance et la communication. Il est chargé de décoder les signaux d'entrée et de l'exécution des actions de contrôle. Selon un programme stocké dans sa mémoire et les décisions sont communiquées sous forme de signaux d'action aux sorties. [44]

Il contient les composants suivants : voir la figure (II-20)

➤ **Unité arithmétique et logique (ALU):**

Il effectue toutes les opérations numériques et logiques et traite tous les données. [45]

➤ **Mémoire sous forme de registre (08 bits):**

Il s'agit d'un espace de stockage provisoire qui permet un transfert rapide des données et des instructions déjà utilisées. [45]

➤ **Unité de commande:**

Il spécifie la séquence d'instructions et de contrôles nécessaires pour récupérer les transactions appropriées doivent être mis en œuvre. [45]

➤ **Mémoire:**

L'automate contient de nombreux éléments de mémoire tels que la "RAM" et la "ROM" la mémoire vive (RAM) est utilisée pour stocker les données logicielles et les programmes développés par l'utilisateur et (ROM) est une mémoire morte. Sa fonction principale est de sauvegarder le système d'exploitation et les données nécessaires au fonctionnement du processeur. [45]

➤ **Interface entrée et sortie:**

Les informations sont acheminées à l'intérieur et à l'extérieur de l'API par le biais des modules d'Entrée et sortie et on trouve aussi plusieurs types de cartes d'E/S sont disponibles pour correspondre aux différents types d'entrées et de sorties que le processeur peut exploiter dans son fonctionnement logique. [45], qui sont :

➤ **Cartes d'entrées TOR (DI):**

En général une carte d'entrée TOR contient 8 ou 16 ou bien 32 entrées numériques logiques. L'automate est informé de l'état du procédé grâce à des capteurs associés aux entrées de l'automate pour qu'il convertit les informations. [46]

Carte d'entrée distincte gère des appareils distincts qui signalent l'activation ou la désactivation. [46]

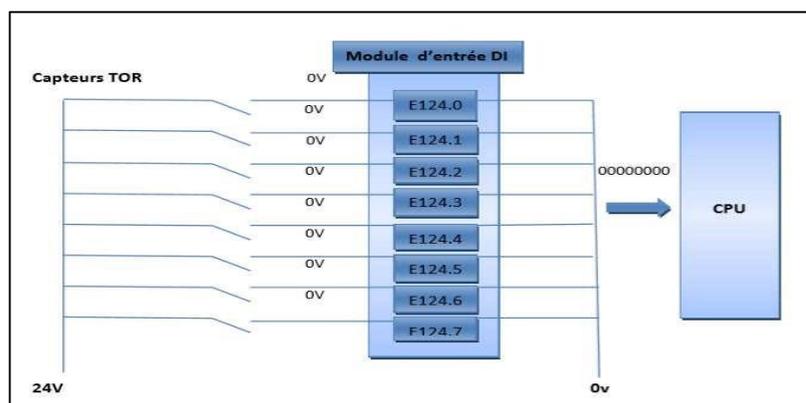


Figure II-20: connexion des entrées à l'état de repos [47]

Une distinction est également faite entre les contacts normalement fermés et les contacts normalement ouverts. [46]

Les capteurs TOR sont interconnectés et alimentés en 24V par l'automate, tandis que les entrées sont reliées aux : E124.0, E124.2 [46]

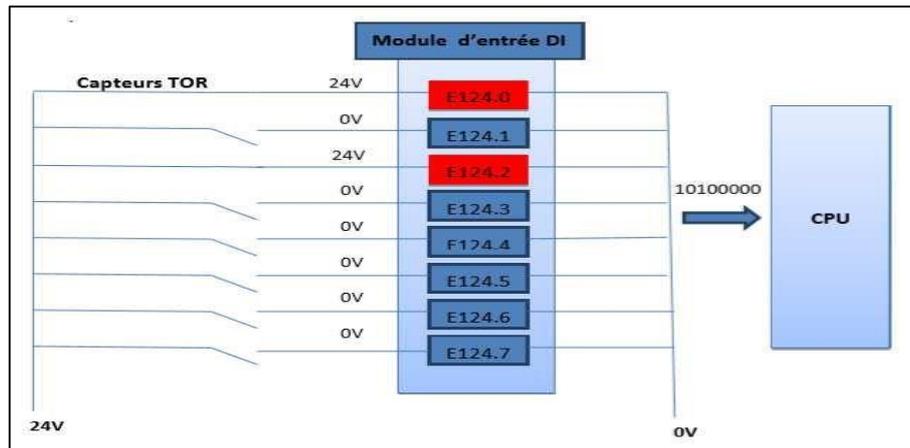


Figure II-21: connexion des entrées à l'état de repo [47]

➤ **Cartes de sorties TOR (DO):**

En règle générale, chaque carte comporte de 8 à 32 sorties logiques, il correspondre aux schémas fonctionnels suivants :

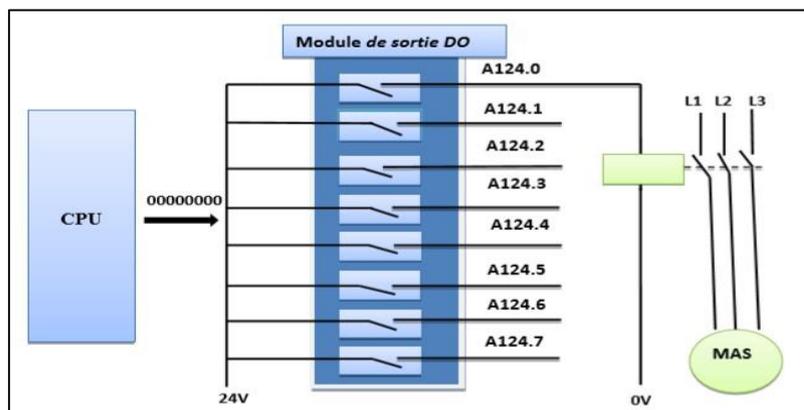


Figure II-22: connexion des sorties à l'état de repos [47]

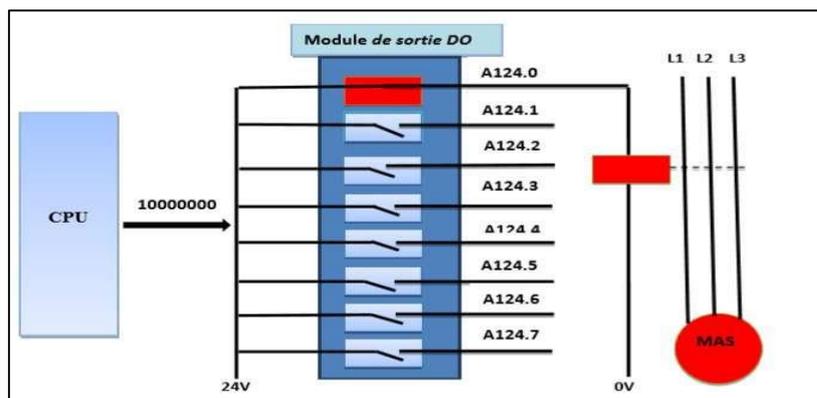


Figure II-23: commande des sorties avec état actionnée [47]

Un automate gère le processus en reliant des actionneurs à des sorties d'automate activées par une tension de commande de 24 volts permettant d'effectuer des actions telles que le

démarrage ou l'arrêt du moteur, ou encore l'activation ou la désactivation des vannes d'alimentation ou des lampes. [46]

Les pré-actionneurs sont reliés à l'automate à borne négative 24V et le trait de programme automatique est équipé de relais distribuant le 24V aux pré-actionneurs et si le relais A124.0 n'est pas fermé par le logiciel, le moteur ne peut pas démarrer. [46]

➤ **Carte d'entrée analogique (AI):**

Il sont utilisées pour transformer le courant ou la tension (un signal entre 0 et 20 mA) en un nombre équivalent que le CPU peut comprendre.[43]

Le rôle du composant de la carte d'entrée analogique de l'automate, qui convertit le signal analogique en signal numérique. [43]

➤ **Carte de sortie analogique (AO):**

La fonction principale de cette carte est de convertir en tension ou courant réel le nombre numérique envoyé par le CPU. [43]

Les plages de signal de sortie typiques sont de 0 à 10 V ou de 4 à 20 mA. Ils sont utilisés pour piloter des contrôleurs de débit massique, des contrôleurs de pression et des contrôleurs de position. [43]

Un module d'E/S analogique est responsable de la lecture (entrée) ou du contrôle (sortie) d'un élément variable d'état, tel qu'un élément chauffant contrôlé par un thermostat, et le module d'E/S est situé dans l'emplacement du rack après le CPU, comme illustré dans la figure II-24. [48]

➤ **Modules de fonctionnements (FM):**

Une unité spéciale est requise pour différents types de traitement de contrôle, ces contrôleurs spéciaux sont appelés unités fonctionnelles ou soi-disant FM, ces modules transmettent des signaux à l'extérieur de l'unité centrale, lors de l'utilisation de FM. [49]

➤ **Processeur de communication (CP):**

Il facilite l'échange d'informations entre plusieurs contrôleurs logiques (PLC) et peut être interfacé non seulement avec le contrôleur d'un système automatisé mais également avec les panneaux de commande. [50]

Chaque point d'E/S a une adresse unique que le CPU peut utiliser. [50]

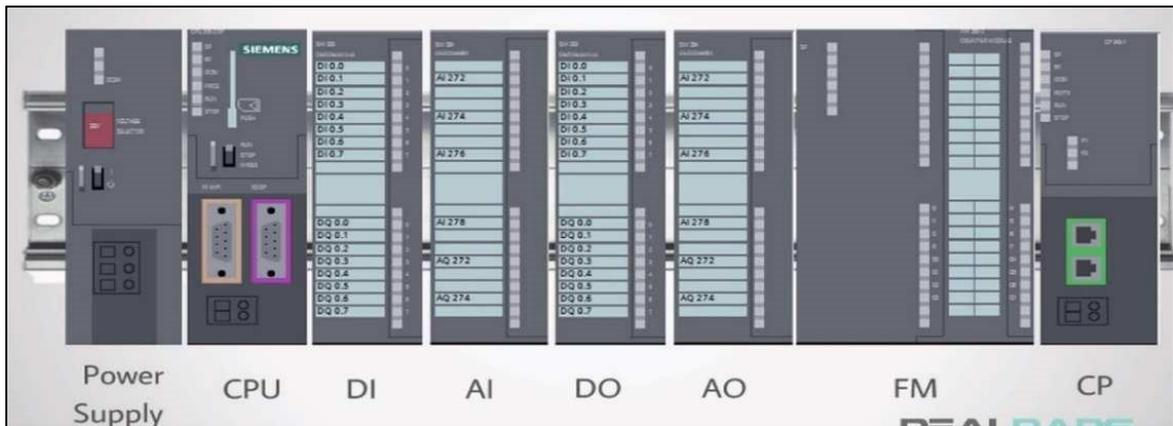


Figure II-24: PLC Hardware [48]

➤ **Bus :**

Ce sont les canaux utilisés pour assurer la communication interne de l'automate.

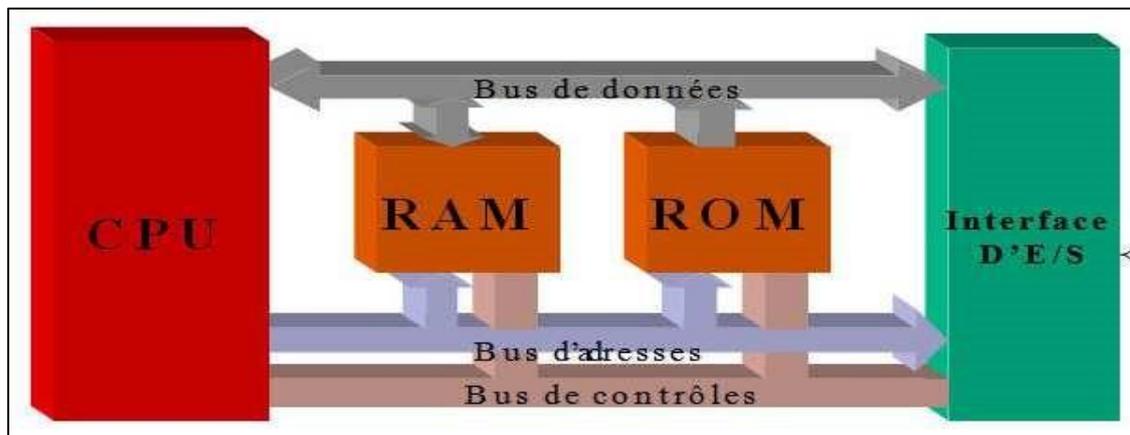


Figure II-25: Bus entre le CPU et l'interface I/O [50]

Ces bus établissent une connexion entre le microprocesseur, les périphériques, les mémoires et les interfaces d'entrée/sortie.[50]

➤ **Bus de données (Data bus):**

La mémoire contient les informations nécessaires au processeur pour effectuer les traitements, tandis que l'interface I/O assure la transmission des données entrant et sortant de l'ordinateur. [50]

➤ **Bus des adresses:**

Connu pour être unidirectionnel et il permet relier le microprocesseur avec la RAM, la ROM et l'interface entrée /sortie. [50]

➤ **Bus de contrôle:**

Il permet de synchroniser les flux d'informations sous forme binaire via les bus mentionnés précédemment. [50]

➤ **Principe générale du fonctionnement d'un API:**

Le principe général du fonctionnement d'un automate programmable industriel est présenté comme illustré sur la figure II-26. [51]

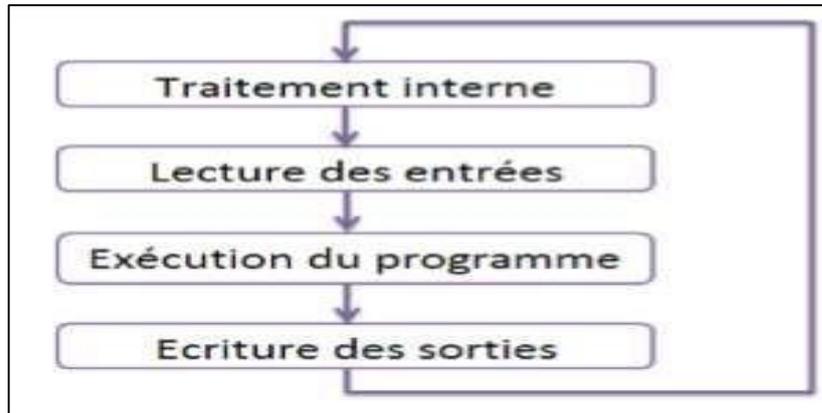


Figure II-26:Fonctionnement d'un API [51]

➤ **Traitement interne:**

L'automate exécute des fonctions qui vérifient et mettent à jour certains paramètres du système, comme la détection de voie en mode RUN ou STOP, etc. [51]

➤ **Lecture des entrées:**

L'API lit synchrone l'entrée, puis la copie dans l'image d'entrée. [51]

➤ **Exécution du programme:**

L'unité de contrôle exécute les instructions du programme de manière séquentielle et inscrit la sortie dans la mémoire d'image de sortie. [51]

➤ **Ecriture des sorties:**

L'API convertit les différentes sorties aux adresses spécifiées dans la mémoire des sorties. [51]

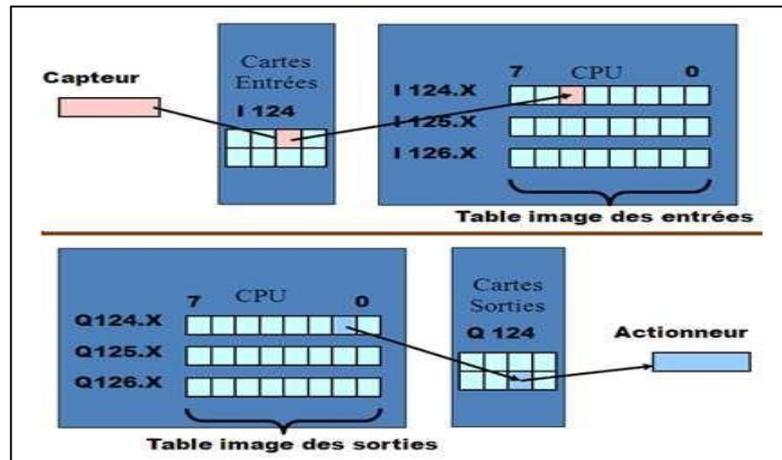


Figure II-27: Interfaces des entrées et des sorties [52]

➤ **Marques et logiciels d'automates :**

Il existe de nombreuses marques, chacune avec son propre logiciel associé. Parmi ces marques les plus utilisées dans l'industrie on trouve : [40]

- Les Automates Omron.
- Les Automates WAG.
- Les Automates Siemens.
- Les Automates Mitsubishi.

Les automates de type Siemens utilisent le logiciel "step7, Win CC ou Tia Portal". Ces logiciels sont utilisés pour la programmation et pour la création d'interface homme/machine (IHM). Afin de tester le fonctionnement nous pouvons utiliser un programme appelé CIROS et Il nous permet de créer, de programmer puis simuler des installations distribuées de complexité variable. Ces logiciels sont classés par gamme sous un nom comme SIMATIC S7-1200. [40]

Les processeurs de la série S7-1200 disposent d'une mémoire de travail suffisante et d'un temps d'exécution hautes performances pour répondre aux exigences de l'industrie. D'autres familles de processeurs, telles que les SIMATIC S7-300 et SIMATIC S7-400, sont également disponibles.[40]

La sélection de l'API se résume à quelques critères clés : [40]

- Type et nombre d'entrées/sorties.
- La vitesse de traitement.
- La taille de la mémoire.
- Le type du processeur.
- Délai de traitement.
- La communication avec les systèmes.

II.6 API choisi:

Pour notre travail nous avons opté pour le contrôleur S7-1200.

II.6.1 Description de l'Automate S7-1200:

Le contrôleur S7-1200 assure la commande vaste gamme d'équipements pour répondre à notre exigence d'automatisation. [56]

Le contrôleur S7-1200 combine un microprocesseur, une alimentation intégrée, un circuit d'entrée et sorties, des entrées analogiques intégrées dans un boîtier compact, après le chargement du programme, la CPU contient la logique nécessaire pour surveiller et contrôler les appareils de votre application. [56]

Le CPU surveille les entrées et modifie les sorties en fonction de la logique de notre programme. Dans le contrôleur S7-1200 on trouve :

- 1) Un emplacement pour la carte mémoire situé sous le volet supérieur. [56]
- 2) Les connecteurs destinés au câblage utilisateur sont amovibles et situés derrière les volets. [56]
- 3) Des diodes lumineuses sont présentes pour indiquer l'état des entrées/sorties intégrées. [56]
- 4) Le connecteur PROFINET se trouve sur la face inférieure de l'unité centrale de traitement (CPU). [56]

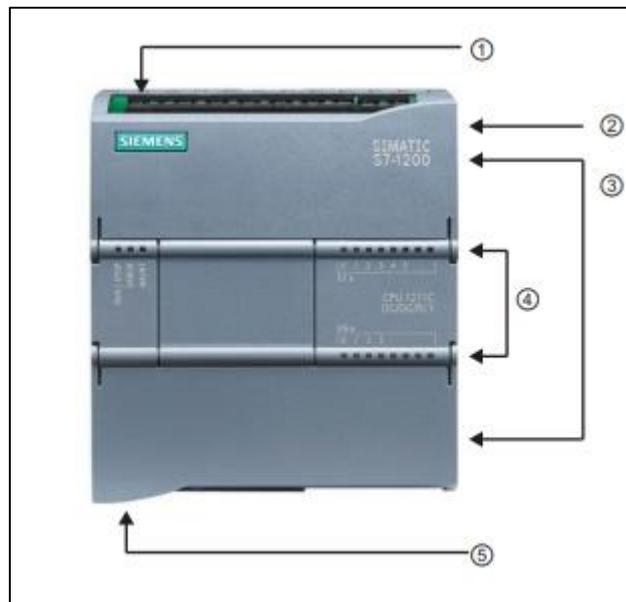


Figure II-28: Composants d'un S7-1200 [56]

II.6.2 Avantages du S7-1200:

Les automates S7-1200 présentent des avantages indéniables notamment leur format compact, Le CPU contient une alimentation intégrée, des entrées/sorties numériques avec un port Profinet intégré et des entrées analogiques. [57]

Avec sa conception modulaire et flexible et son interface de communication de pointe et sa gamme étendue de fonctions technologiques performantes et intégrées. Elle est aussi polyvalente qui répond à de multiples applications en milieu industriel.[57]

II.6.3 Choix du CPU:

Les CPU système **SIMATIC S7-1200** ont trois types de performances, en premier on a le **CPU1211 C**, nous avons également le **CPU 1212C** et sans oublier **CPU1214 C**. Dans tous ces CPU nous permettons d'ajouter une carte d'extension pour ajouter des E/S et sur le côté droit. [56]

Nous avons utilisé le CPU « **CPU1214 C** » pour notre projet.

II.6.3.1 CPU1214 C:

Le CPU1214C de Siemens est un CPU puissant et compact qui fait partie de la famille SIMATIC S7-1200. Il convient à une vaste gamme d'applications industrielles grâce à son processeur haute performance, son interface Ethernet intégrée et sa mémoire de programme de 75 Ko, extensible jusqu'à 1 Mo. [56]

Le CPU contient quatorze entrées et de dix sorties numériques, avec deux entrées analogiques et deux sorties analogiques. Il peut être programmé à l'aide du logiciel de programmation STEP 7 avec des fonctions de programmation avancées. [56]

Le CPU1214C est compatible avec de nombreux accessoires et modules d'extension pour répondre facilement à des besoins spécifiques. Avec ses hautes performances, sa grande flexibilité et ses fonctionnalités avancées, le CPU1214C est une solution d'automatisation idéale pour les applications industrielles les plus exigeantes. [56]

II.7 Description du module d'emmagasinage à contrôle cartésien (MOD.MCS-620/EV):

Comme mentionné au début du chapitre, notre système est un système de d'emmagasinage des pièces et il contient 12 positions, 4 pour chacun des trois niveaux comme indiqué sur la figure II-29 : [17]

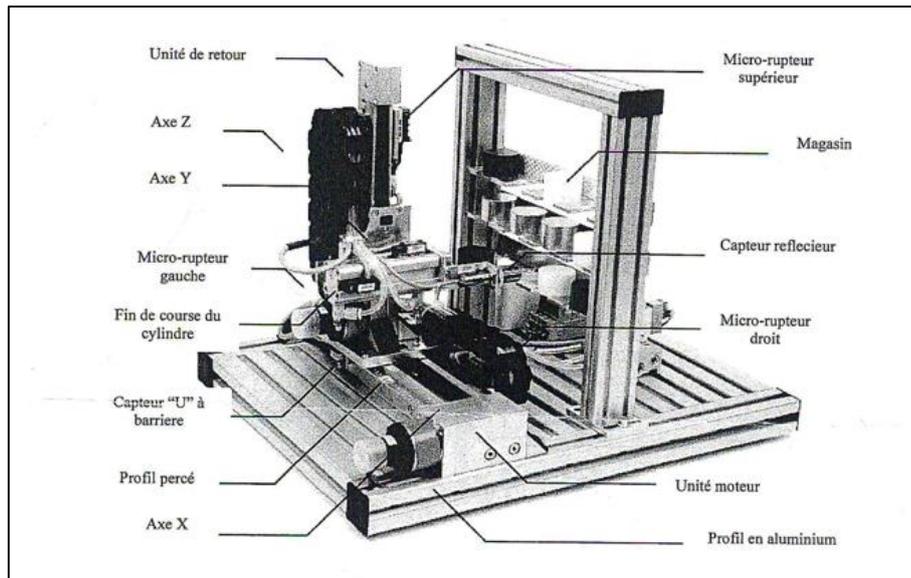


Figure II-29:MOD.MCS-620/EV [17]

II.7.1 Composants du module :

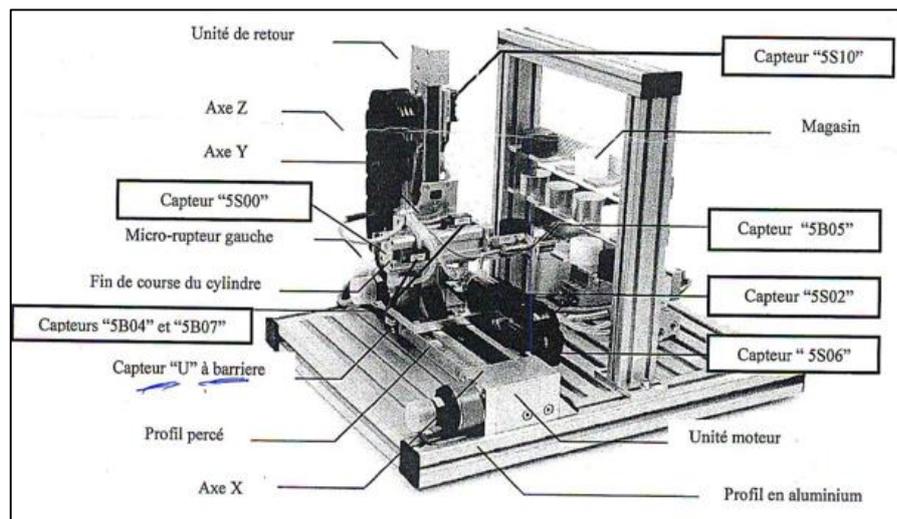


Figure II-30: Composants du MOD.MCS-620/EV

Le mouvement de la pince selon l'axe X et l'axe Z est géré par deux moteurs de 24 V. Les capteurs « 5S02 », « 5S10 », « 5S00 » et « 5S06 » représentent les fins de course des axes et ils permettent d'arrêter les moteurs une fois la position finale est atteinte, ils sont également utile pour trouver la position initiale au début du cycle de service. [17]

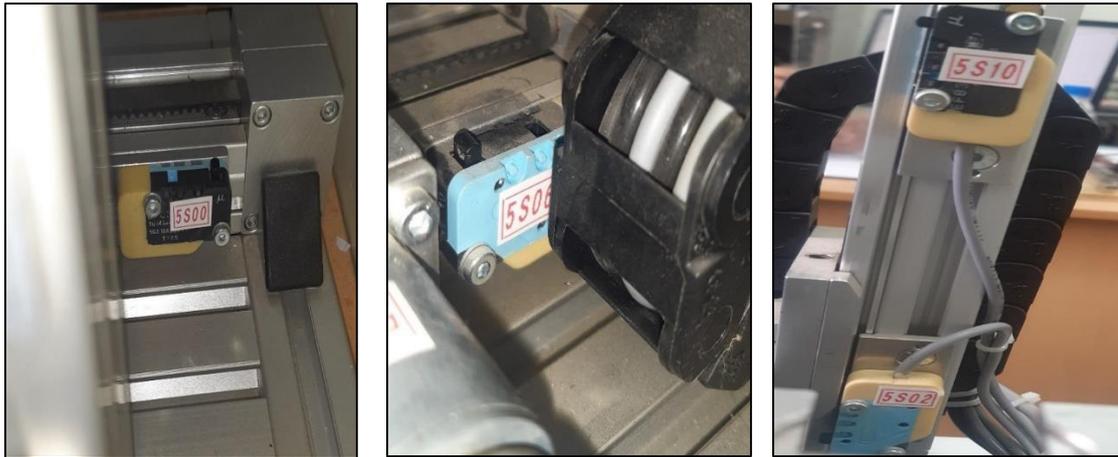


Figure II-31: Capteurs de fins de course

On a aussi le capteur réflecteur « **5B05** ». Il détecte la présence ou l'absence de la pièce ou d'un produit dans la cellule du magasin. [17]



Figure II-32: Capteur réflecteur

Les deux capteurs « **5B04** » et « **5B07** » déterminent l'emplacement du vérin sur l'axe Y. [17]



Figure II-33: Capteurs magnétiques

II.7.1.1 Deux codeurs linéaires:

Ils mesurent la position linéaire (distance) par calcul des impulsions et il est nécessaire de faire passer ces impulsions à travers un système de traitement de signal externe pour obtenir une mesure de la distance par rapport à l'emplacement de référence. Il existe un codeur linéaire pour l'axe X et un autre pour l'axe Y, Ils sont bidirectionnels avec deux sens. [58]

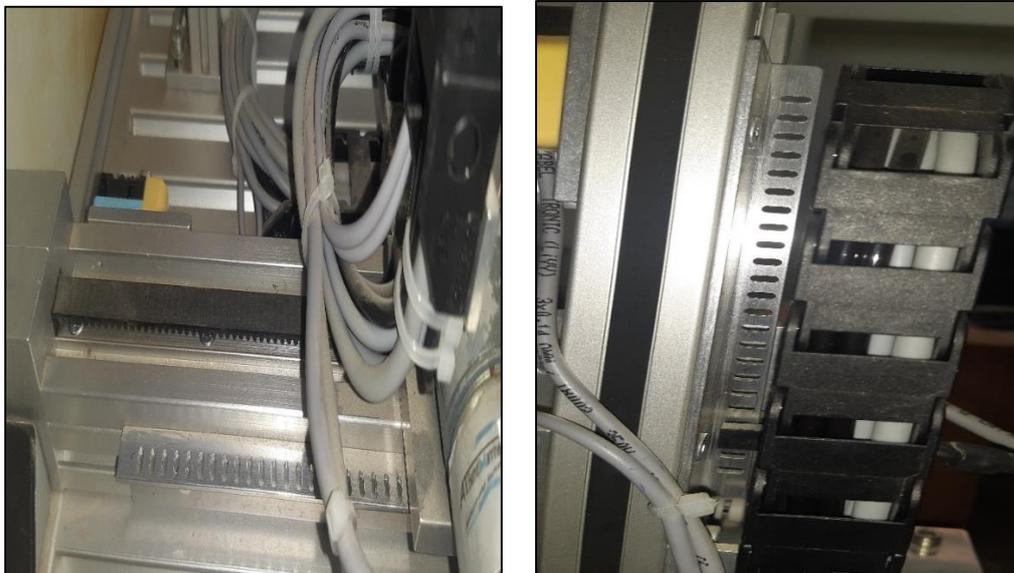


Figure II-34: Codeurs linéaires de l'axe X et de l'axe Y

II.7.1.2 Moteurs:

Moteur "5M1" entraîne le mouvement du bras sur l'axe X dans les deux sens : « 5M1 + » pour le sens gauche et « 5M1 - » pour le sens opposé (droite).

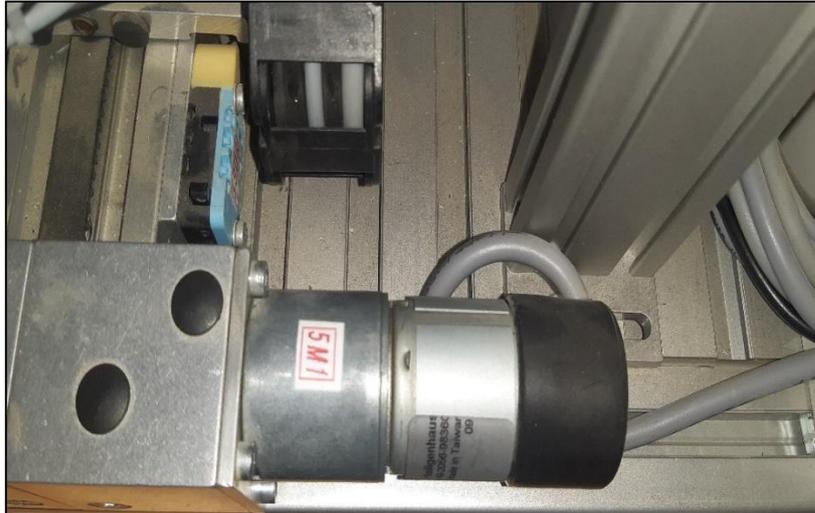


Figure II-35: Moteur 5M1

Le moteur "5M2" entraîne le mouvement du bras sur l'axe Y dans les deux sens: « 5M2 -> » pour déplacer le bras vers le bas sur l'axe Y et « 5M2+ » pour le sens opposé (vers le haut).

II.7.1.3 Box d'interface:

Le boîtier d'interface fournit les signaux d'entrée et de sortie au module. Tous les signaux sont câblés sur un connecteur de 25 broches (Tableau 1) ou sur des bornes de 2 mm situées sur le couvercle du boîtier. [17]

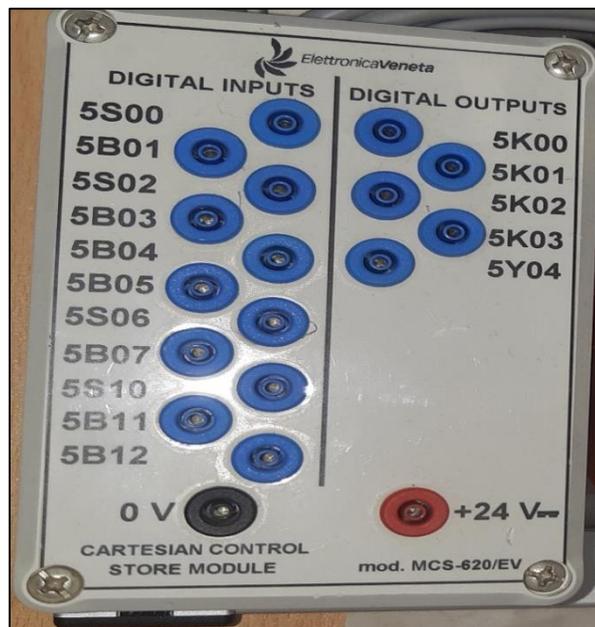


Figure II-36: Box d'interface

Tableau 1: Composants de MOD.MCS-620/EV [17]

Type	Indicatif	Description
Capteur	5S00	Micro rupteur position à gauche sur l'axe X
Capteur	5B01	Capteur de lumière U
Capteur	5S02	Micro rupteur position en bas sur l'axe Z
Capteur	5B03	Capteur optique U
Capteur	5B04	Capteur magnétique position retirée sur l'axe Y
Capteur	5B05	Capteur optique détecte la présence de la pièce
Capteur	5B06	Micro rupteur position à droite sur l'axe X
Capteur	5B07	Capteur magnétique
Capteur	5S10	Micro rupteur position en haut sur l'axe Z
Capteur	5B11	Capteur optique U
Capteur	5B12	Capteur optique U
Relais M1	5K00	Mouvement vers gauche sur l'axe X
Relais M1	5K01	Mouvement vers droite sur l'axe X
Relais M2	5K02	Mouvement vers le bas sur l'axe Z
Relais M2	5K03	Mouvement vers le haut sur l'axe Z

II.8 Fonctionnement de notre système:

Nous allons décrire le fonctionnement de notre système selon l'organigramme suivant:

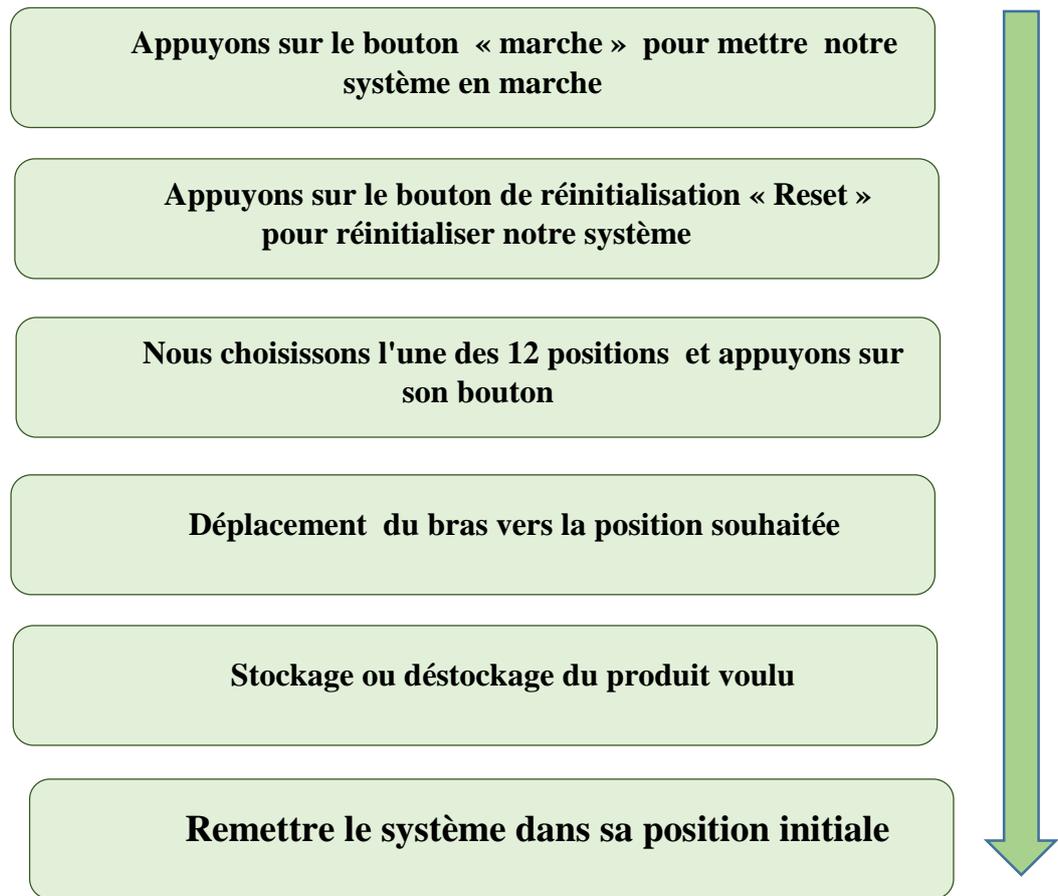


Figure II-37: Organigramme représentant le cahier de charge du mod mcs-620

II.9 Conclusion:

Au cours de ce chapitre, nous avons effectué une introduction à tout ce qui concerne les capteurs, les actionneurs et les API. Par la suite nous avons défini les composants de notre système et à la fin nous avons établi un cahier de charge pour notre système.

Dans le chapitre 3, nous explorerons en profondeur les phases de programmation et fournirons une explication détaillée de leur fonctionnement.

III. CHAPITRE 3

Programmation du MOD.MCS- 620/EV

III.1 Introduction:

Dans le troisième chapitre, nous appliquons les connaissances acquises dans les chapitres précédents. Ce chapitre est dédié à la programmation du langage à contacts et il est nécessaire pour comprendre les principes de base de la programmation des automates programmables industriels

Dans ce chapitre, nous explorerons les fondements de ce langage de programmation, couramment utilisé dans l'industrie pour concevoir des systèmes de contrôle et d'automatisation. Dans un premier temps, nous allons présenter de manière générale le langage Ladder ainsi que ses principales caractéristiques.

A la fin nous représentons le programme Ladder développé pour notre système avec une explication de chaque étape.

III.2 Principe de la logique programmée:

III.2.1 Logique :

Les lois de commande sont mises en œuvre dans des appareils connectés (avec câblage) sur la base de schémas théoriquement ou empiriquement construits pour exécuter des fonctions logiques de base, nos opérateurs logiques de base permettent des portes logiques AND/OR/NOT/NAND/NO et les relais normalement ouverts ou fermés effectuent toute fonction logique. [53]

La fonctionnalité est physiquement représentée dans le câblage, ce qui signifie que la modification est difficile, que ce soit au stade du développement ou lors des extensions post-processus et d'autre part, la complexité du problème n'affecte pas la vitesse de traitement car le processus est concurrent. [53]

III.2.2 Logique programmée:

Dans ce genre de logique la loi de commande est suffisante pour la modification et la simplification du programme et de la maintenance. Elle permet de se passer de fils encombrants et souvent inélastiques cela grâce à des automates et des microcontrôleurs. [53]

III.3 Langages de programmation de l'automate programmable industriel:

Afin de programmer un système, il est nécessaire de disposer d'un ensemble d'instructions pour effectuer les opérations nécessaires. Il existe différents types de langages de programmation, chacun avec sa propre syntaxe et sa propre façon de travailler. Nous avons cité ci-dessous une liste de ces différentes langues. : [40]

III.3.1 Langages littéraux:

III.3.1.1 Langage IL:

Méthode de programmation similaire à la saisie d'un diagramme à relais sous forme de texte et Les instructions de ligne sont utilisées dans ce langage de programmation et Ceux-ci correspondent en grande partie aux étapes que la CPU traite le programme

```

! %L0 :
      LD      %I1.0
      ANDN   %M12
      OR (    %TM4.Q
      AND    %M17
      )
      AND    %I1.7
      ST     %Q2.5
! %L5 :
      LD      %I1.10
      ANDN   %Q2.3
      ANDN   %M27
      IN     %TM0
      LD     %TM0.Q
      AND    %M25
      AND    %M000:XS
      [ %M0015 := %M0018+S00]

```

Figure III-1: Exemple de langage IL [40]

III.3.1.2 Langage Littéral Structuré (ST):

Il est peu couramment utilisé par les ingénieurs en automatique que les autres langages informatiques, aussi il utilise diverses fonctions comme conditionnelles « if...else ». [40]

```

IF %M0 THEN
  FOR %M0099 := 0 TO $1 DO
    IF %M00100 [%M0089] > 0 THEN
      %M0010 := %M00100 [%M0089]
      %M0011 := %M0099;
      %M1 := TRUE;
      EXIT;          (*Sortie de la boucle FOR*)
    ELSE
      %M1 := FALSE;
    END_IF;
  END_FOR;
ELSE
  %M1 := FALSE;
END_IF;

```

Figure III-2: Exemple d'un langage ST [40]

III.3.2 Langages graphiques:

III.3.2.1 Langage LD:

C'est un langage spécialement conçu pour les électriciens et c'est aussi le plus utilisé. Il permet de programmer des équations booléennes (vrai/faux) à l'aide de symboles tels que des relais et des blocs fonctionnels. Voici un exemple de ce langage, illustré dans la figure suivante: [40]

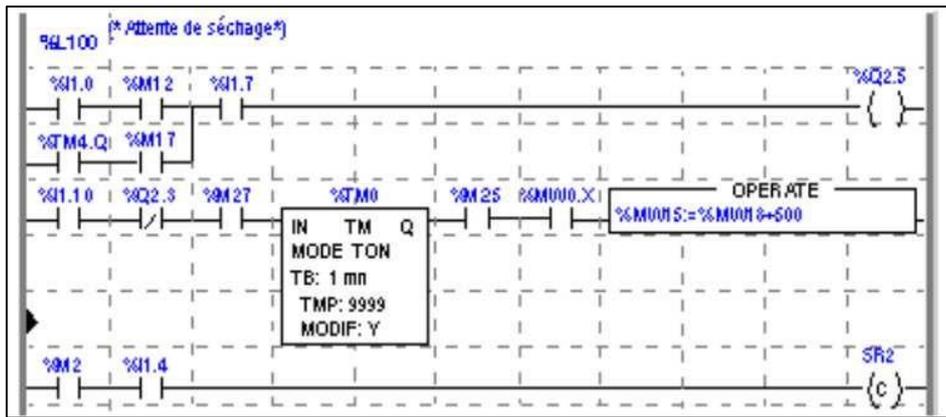


Figure III-3: Exemple d'un langage à contact [40]

III.3.2.2 Bloc fonctionnel (FBD):

Il se compose d'une série de blocs représentant des fonctions, désignées par des rectangles. Comme le montre la figure ci-dessous, les entrées sont à gauche et les sorties sont à droite. [40]

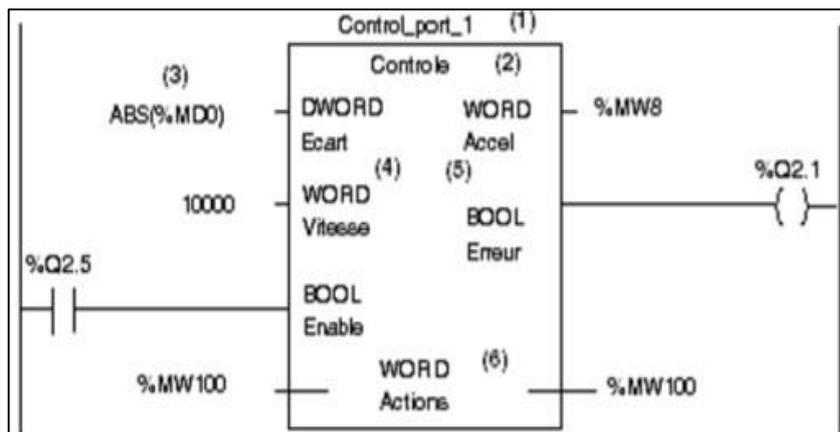


Figure III-4: Exemple d'un langage FBD [40]

Ce programme exécute toutes sortes de fonctions, des plus simples aux plus complexes. [40]

III.3.2.3 Langage SFC:

Il appelle aussi grafcet et il est défini comme un graphe de commande fonctionnel contenant des étapes et des transitions et les étapes sont reliées entre elles par des transitions, Nous avons toutes les transitions qui peuvent contenir une ou plusieurs conditions booléennes et il décrit également les opérations séquentielles. Les actions des étapes dans ce graphe sont décrites dans les langages ST où bien LD ou FBD. [40]

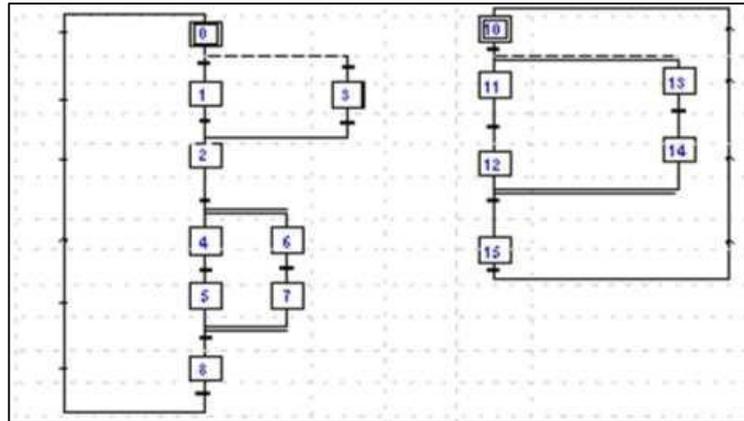


Figure III-5: Exemple d'un grafcet (langage SFC) [40]

III.3.3 Langage de Programmation choisi:

Nous avons opté pour le langage de programmation le langage CONTACT où il s'appelle langage LADDER.

III.3.3.1 Définition du langage LADDER:

Il est constitué de plusieurs réseaux les uns après les autres. D'autres par, ces grilles sont composées de différentes icônes représentant les entrées/sorties d'api et les opérateurs timers, compteurs, etc. Les opérations. [55]

➤ **Symboles du programme:**

On distingue trois types d'éléments dans ce langage : [55]

- **Les entrées ou les contacts :** Le principe de fonctionnement consiste à lire la valeur d'une variable booléenne. [55]
- **les bobines ou les sorties:** Le principe de fonctionnement consiste à définir ou affecter la valeur d'une variable booléenne. [55]
- **les blocs fonctionnels :** Ils nous offrent la possibilité de réaliser des fonctions avancées.[55]

Tableau 2: Principaux éléments du programme LD. [55]

Object graphique	signification
- / -	Un Contact normalement fermé
- -	Un Contact normalement ouvert
- P -	Un Contact fermé (front montant)
- N -	Un Contact fermé (front descendant)

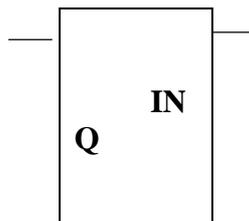
-(/)-	Une Bobine normalement fermée
-()-	Une Bobine normalement ouverte
-<return>	Le Retour incondtionnel (vers le sous-programme appelant)
-cond-<return>	Retour conditionnel
->>Label	Saut incondtionnel
-cond->>Label	Saut conditionnel
-(R)-ou -(U)-	Bobine Reset (remise à 0 de la bobine)
-(S)- ou -(L)-	Bobine (maintenu à 1 une fois actionné)
-(P)-	Bobine active au front montant de son entrée
-(N)-	Bobine active au front descendant de son entrée

Blocs fonctionnels ou les circuits séquentiels:

Blocs temporisations:

Des entrées (I) connectées à éléments graphiques précédents et lorsque le temps expire, la sortie est activée lorsque l'activation de la minuterie atteint la valeur prédéfinie. [55]

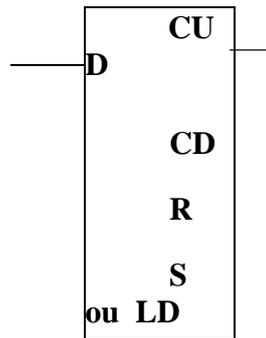
- **Q : la sortie**
- **IN : c'est l'entrée**



Fonctions de comptage/décomptage:

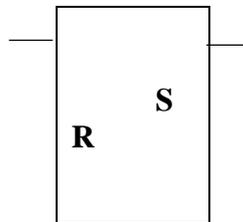
Selon les marques ils peuvent être séparés ou regroupés en bloc. [55]

- **CU** : Il indique que l'entrée du compteur est déclenchée par un front montant. [55]
- **CN** : Cette entrée représente le décompte déclenché par un front montant. [55]
- **R** : Il s'agit de l'entrée de réinitialisation de la valeur actuelle. [55]
- **S ou LD** : Cette entrée correspond à la charge de la valeur préétablie. [55]
- **D** : C'est la sortie qui est activée lorsque la valeur de présélection est atteinte, soit en comptage, soit en décomptage. [55]



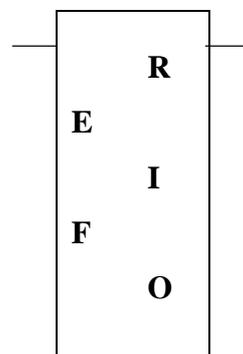
Monostables:

- **S** : L'entrée d'activation du flux unipolaire est déclenchée par un front montant.
- **R** : c'est la sortie logique.



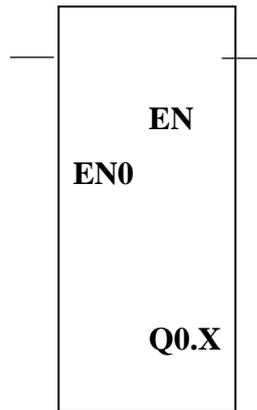
Registre TÉLÉMÉCANIQUE:

- **R** : C'est l'entrée de réinitialisation d'enregistrement. [55]
- **I** : Cela marque l'entrée du stockage sur un front montant. [55]
- **O** : Cela marque l'entrée du déstockage sur un front montant.[55]
- **E** : Il s'agit de la sortie qui indique le registre est vide.[55]
- **F** : Il s'agit de la sortie qui indique le registre est plein. [55]



□ **Générateur d'impulsions SIEMENS:**

- **EN :** C'est cette entrée qui déclenche le train d'impulsions. [55]
- **ENO :** C'est la sortie qui connecter nombreux générateurs pas en parallèle mais en série. nous avons $ENO = EN$. [55]
- **Q0.X :** Il s'agit d'une sortie. elle est réservée aux sorties uniquement Q0.0 et Q0.1...[55]



□ **Structure d'un réseau LD:**

Il est composé comme suit : le titre et le commentaire plus une grille graphique comprenant la zone de test et sa zone d'action respective [55]

La zone de test contient :

- les blocs de comparaison. [55]
- les blocs fonction comme les compteurs et temporisations [55]
- les contacts. [55]

La zone d'action :

- les blocs d'opérations.
- les bobines.

□ **règles de développement d'un réseau de contacts sont:**

La grille connectée au réseau est lue en se connectant au réseau de haut en bas en séquence de gauche à droite dans un réseau connecté. [55]

Une grille connectée est formée par des éléments graphiques reliés les uns aux autres, mais chaque élément est autonome. Si une corrélation verticale d'affinité est trouvée, le sous-

réseau est d'abord évalué qui y est associé avant de procéder à l'évaluation du sous-réseau qu'il inclut. [55]

III.4 **Tableau des entrées et sorties:** Au cours de notre travail nous avons défini le tableau des entrées/sorties suivant :

Default tag table								
	Name	Data type	Address	Retain	Acces...	Writa...	Visibl...	Comment
1	lampe vert	Bool	%Q0.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
2	5s06	Bool	%I0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
3	5b03	Bool	%I0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
4	5s02	Bool	%I0.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
5	5b01	Bool	%I0.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
6	5s00	Bool	%I0.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
7	5s10	Bool	%I0.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
8	reset	Bool	%I0.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
9	5m1+	Bool	%Q0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
10	5m2-	Bool	%Q0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
11	5m1-	Bool	%Q0.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
12	5m2+	Bool	%Q0.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
13	compteur	Int	%MW1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
14	lampe bleu	Bool	%Q0.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
15	p1	Bool	%I0.7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
16	p2	Bool	%I1.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
17	p3	Bool	%I1.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
18	p4	Bool	%I1.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
19	p5	Bool	%I1.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
20	p6	Bool	%I1.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
21	p7	Bool	%I1.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
22	p8	Bool	%I1.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
23	p9	Bool	%I1.7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Default tag table								
	Name	Data type	Address	Retain	Acces...	Writa...	Visibl...	Comment
23	p9	Bool	%I1.7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
24	p10	Bool	%I2.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
25	p11	Bool	%I2.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
26	p12	Bool	%I2.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
27	temps	Time	%MD10	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
28	marche	Bool	%I2.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
29	arret	Bool	%I2.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
30	lampe rouge	Bool	%Q0.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
31	lampe p1	Bool	%Q0.7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
32	lampe p2	Bool	%Q1.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
33	lampe p3	Bool	%Q1.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
34	lampe p4	Bool	%Q1.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
35	lampe p6	Bool	%Q1.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
36	lampe p7	Bool	%Q1.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
37	lampe p8	Bool	%Q1.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
38	lampe p9	Bool	%Q1.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
39	lampe p10	Bool	%Q1.7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
40	lampe p11	Bool	%Q2.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
41	lampe p12	Bool	%Q2.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
42	marche panel	Bool	%M0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
43	arret panel	Bool	%M0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
44	p1-ihm	Bool	%M0.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
45	p2-ihm	Bool	%M0.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Figure III-6: Tableau des variables

Default tag table								
	Name	Data type	Address	Retain	Acces...	Writa...	Visibl...	Comment
46	p3-ihm	Bool	%M0.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
47	p4-ihm	Bool	%M0.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
48	p5-ihm	Bool	%M0.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
49	p6-ihm	Bool	%M0.7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
50	p7-ihm	Bool	%M3.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
51	p8-ihm	Bool	%M3.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
52	p9-ihm	Bool	%M3.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
53	p10-ihm	Bool	%M3.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
54	p11-ihm	Bool	%M3.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
55	p12-ihm	Bool	%M3.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
56	lampe p1-ihm	Bool	%M3.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
57	lampe p2-ihm	Bool	%M3.7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
58	lampe p3-ihm	Bool	%M4.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
59	lampe p4-ihm	Bool	%M4.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
60	lampe p5-ihm	Bool	%M4.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
61	lampe p6-ihm	Bool	%M4.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
62	lampe p7-ihm	Bool	%M4.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
63	lampe p8-ihm	Bool	%M4.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
64	lampe p9-ihm	Bool	%M4.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
65	lampe p10-ihm	Bool	%M4.7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
66	lampe p11-ihm	Bool	%M5.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
67	lampe p12-ihm	Bool	%M5.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
68	lampe rouge -ihm	Bool	%M5.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Default tag table								
	Name	Data type	Address	Retain	Acces...	Writa...	Visibl...	Comment
0	lampe p5-ihm	Bool	%M4.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
1	lampe p6-ihm	Bool	%M4.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
2	lampe p7-ihm	Bool	%M4.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
3	lampe p8-ihm	Bool	%M4.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
4	lampe p9-ihm	Bool	%M4.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
5	lampe p10-ihm	Bool	%M4.7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
6	lampe p11-ihm	Bool	%M5.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
7	lampe p12-ihm	Bool	%M5.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
8	lampe rouge -ihm	Bool	%M5.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
9	lampe vert -ihm	Bool	%M5.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
0	lampe bleu -ihm	Bool	%M5.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
1	reset-ihm	Bool	%M5.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
2	reset bouton	Bool	%I3.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
3	lampe p5	Bool	%Q2.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
4	<Add new>			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Figure III-7 : Tableau des entrées et des sorties

III.5 Programme Ladder développé:

❖ Bouton «marche»:

Le bouton **Marche** permet le démarrage de notre système, il allume la lampe verte et il désactive le bouton arrêt.

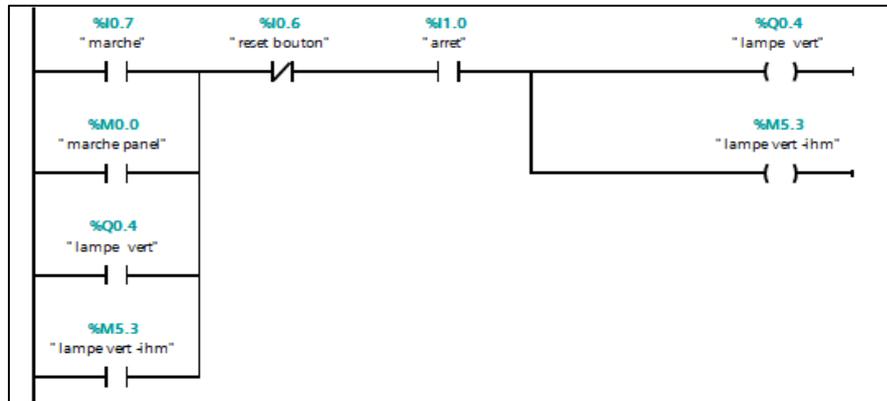


Figure III-8 : Programme associé au bouton de mise en marche

❖ Bouton d'initialisation « reset » :

Le moteur « 5M1- » déplace le bras vers la gauche sur l'axe X jusqu'à le capteur «5s06»

Le moteur « 5m2-» déplace le bras vers le bas sur l'axe Y jusqu'à le capteur « 5s02 ». A cette étape, la lampe bleue doit être allumée.

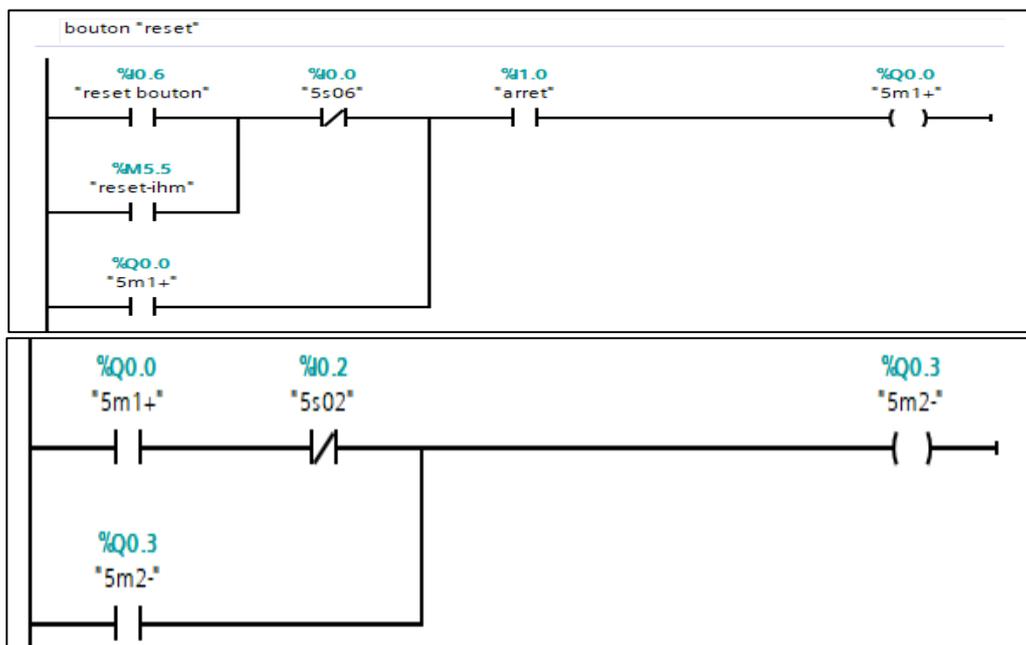


Figure III-9: Programme associé au bouton de mise en reset

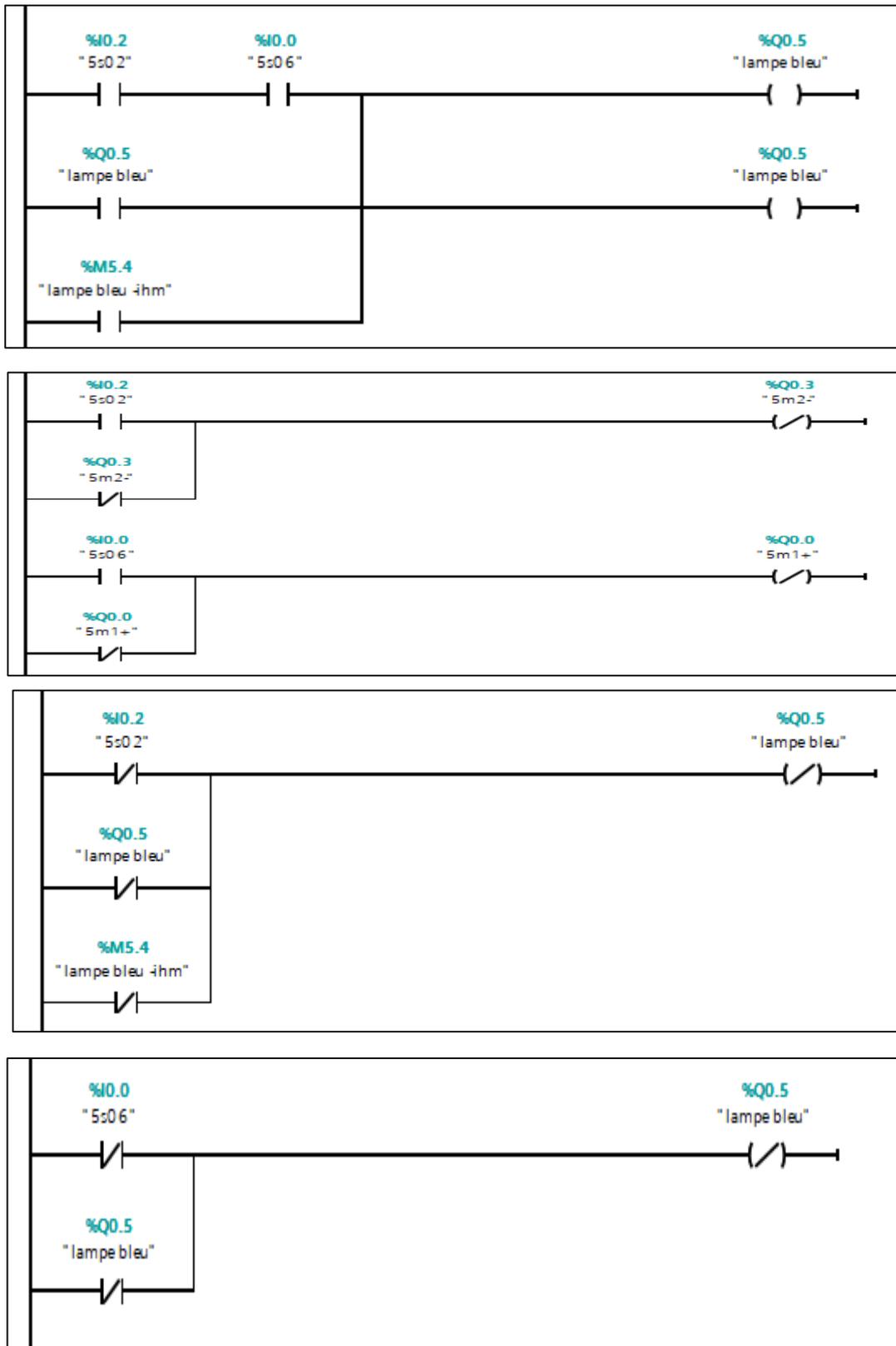


Figure III-10: Programme associé au bouton de mise en marche

❖ **Bouton « arrêt » :**

Ce bouton arrête le système et allumer la lampe rouge.

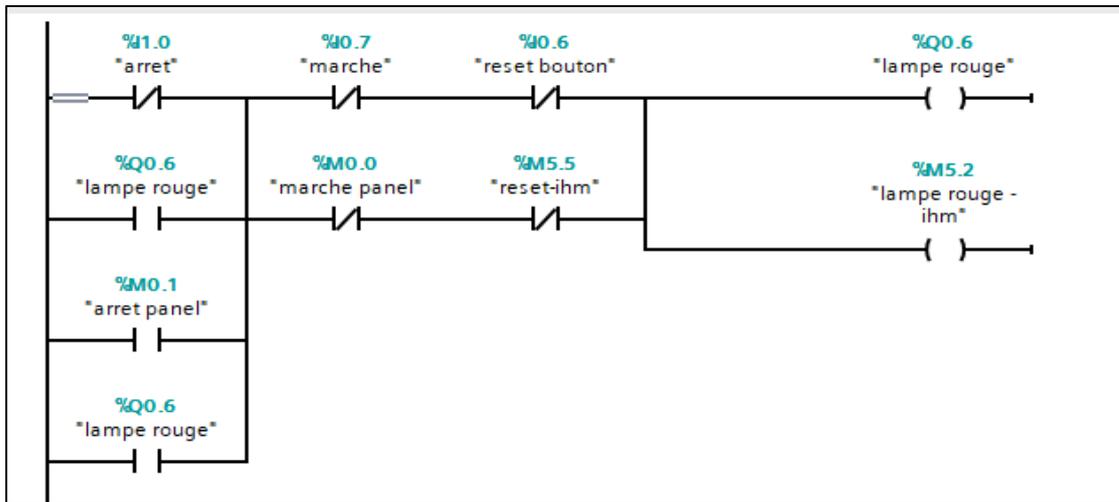


Figure III-11: Programme associé au bouton de mise en arrêt

❖ **Remarque:**

Les boutons de notre interface sont déclarés sur le tableau des variables avec des adresses mémoire (M). Ces boutons doit être en parallèle avec le programme comme illustré sur la figure (III, 11).

❖ **Appuyons sur le bouton « p1 » (position 1):**

Il faut que les deux capteurs « 5S02 » et « 5S06 » sont activés pour mettre le moteur « 5M1+ » en marche. Le capteur « 5B01 » active le comptage pour le calcul de 2 impulsions à fin de déplacer le bras vert la gauche sur l’axe X.

Après, le moteur « 5m2+ » démarre le déplacement du bras vers le haut sur l’axe Y et le capteur « 5B03 » active le comptage pour le calcul de 2 impulsions à fin d’atteindre la position P1.

La lampe « lampe p1 » est activée pendant 3 seconds, par la suite, notre système mettre le moteur « 5m1- » en marche, puis le moteur « 5m2- » en marche pour déplacer le bras sur l’axe X puis sur l’axe Y à fin de le remettre à sa position initiale.

Remarque :

Tous les boutons de notre interface sont déclarés sur le tableau des variables avec les adresses mémoire (M).

❖ **Appuyons sur le bouton « p2 » (position 2):**

Ce bouton a la même action que le bouton p1 sauf que le compteur d'impulsions doit être égal à 7 impulsions sur l'axe X.

Sur l'axe Y, le nombre d'impulsions doit être égal à 2 pour atteindre la position 2.

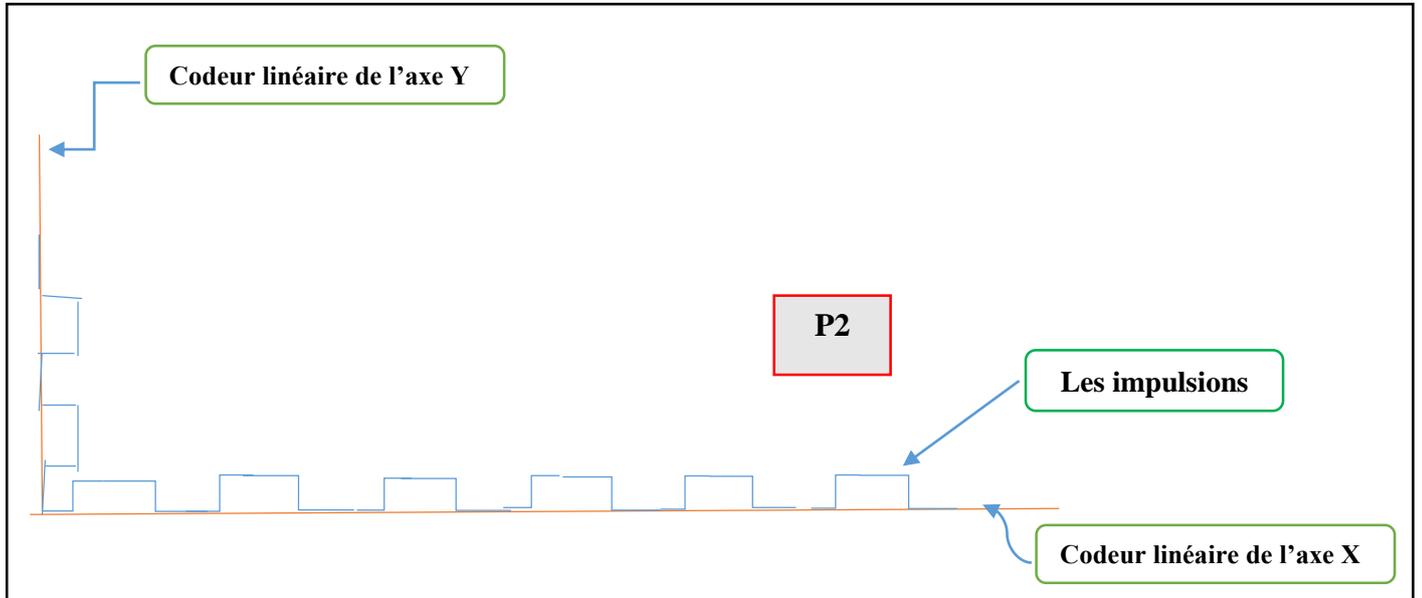


Figure III-12: position du bouton 2 (P2)

❖ **Appuyons sur le bouton « p3 » (position 3) ou « p4 » (position 4) :**

Ils exécutent les mêmes étapes comme dans les boutons p1 et p2. La seule différence est le nombre d'impulsions sur l'axe des X, Pour le bouton p2 il devrait être égal à 18 (camp (p2)= 18) et pour le bouton p3 le nombre d'impulsions égale à 27 (camp (p3)=27).

Compteurs des impulsions :

Il existe dans les deux axes X et Y un codeur linéaire bidirectionnel. Le déplacement sur ces deux axes est effectué après le calcul du nombre d'impulsions.

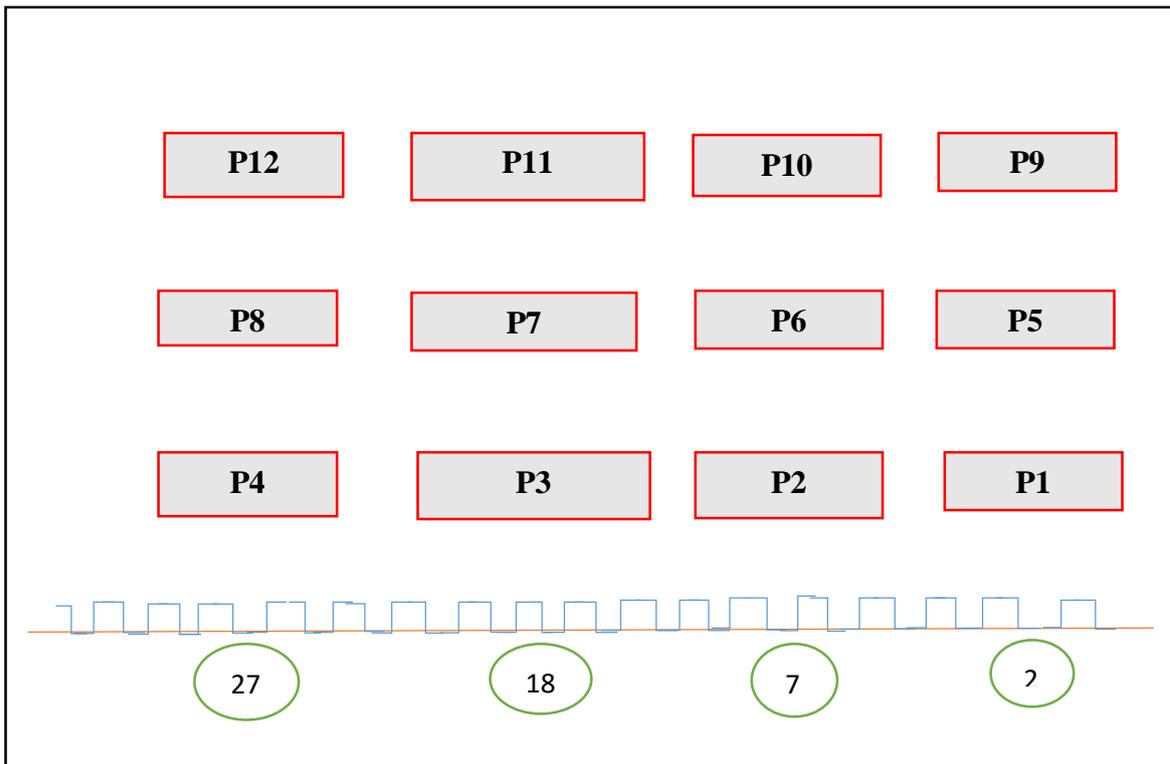


Figure III-13: Nombres d'impulsions sur l'axe X

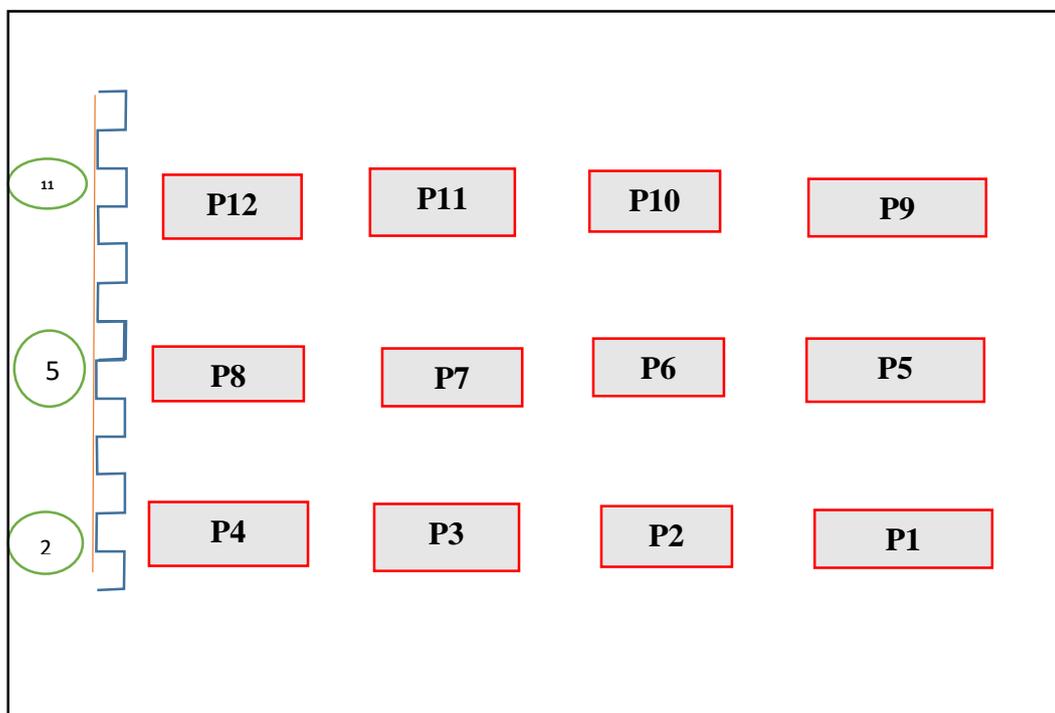


Figure III-14: Nombres des impulsions sur l'axe Y

L'avant dernier niveau de notre système contient 4 positions (Positions 9, 10, 11 et 12) si nous appuyons dans notre interface :

❖ **Appuyons sur le bouton « p5 » : (position 5)**

Il faut que les deux capteurs « 5S02 » et « 5S06 » sont active pour mettre le moteur « 5M1+ » en marche et le capteur « 5B01 » active le compteur pour calculer 2 impulsions pour déplacer le bras vert la gauche sur l'axe X

Après, le moteur « 5m2+ » démarre et le capteur « 5B03 » active le compteur pour calculer 5 impulsions pour déplacer le bras vers le haut sur l'axe Y

La lampe « lampe p5 » est activée pendant 3 seconds, puis on met le moteur « 5m1- » en marche, puis le moteur « 5m2- » pour déplacer le bras sur l'axe X et Y pour le remettre à sa position initiale.

❖ **Appuyons sur le bouton « p6 » (position 6) ou « p7 » (position 7) ou bien « p8 » (position 8):**

Lorsque nous appuyons sur l'un de ces trois boutons, les mêmes étapes du bouton 5 se répètent, il diffère que dans les nombres d'impulsion sur l'axe des X comme nous avons cité auparavant.

Concernant le troisième niveau, Il contient les positions 9 et 10 ,11 et 12.

❖ **Appuyons sur le bouton « p9 » : (position 9)**

Il faut que les deux capteurs « 5S02 » et « 5S06 » sont active pour mettre le moteur « 5M1+ » en marche, le capteur « 5B01 » active le compteur pour calculer 2 impulsions pour déplacer le bras vert la gauche sur l'axe des X.

Par la suite le moteur « 5m2+ » démarre et le capteur « 5B03 » active le compteur pour calcule 11 impulsions pour déplacer le bras vert la gauche sur l'axe des Y.

La lampe « lampe p2 » est activée pendant 3 seconds, après on met le moteur « 5m1- » en marche puis le moteur « 5m2- » pour déplacer le bras sur l'axe X puis l'axe Y pour le remettre à sa position initiale.

❖ **Appuyons sur le bouton « p10 » : (position 10) ou « p11 » (position 11) ou bien le bouton « p12 »:**

Les étapes de fonctionnement ne changent pas, seul le nombre d'impulsions pour chaque bouton.

Temporisateur:

Pour le déplacement de notre bras dans les deux axes X et Y on peut utiliser aussi un **timer** dans notre programme Ladder, ou chaque impulsion peut représenter un temps spécifique.

Par exemple, passer en position 1 (p1) oblige le moteur 5K00 (5M1+) à tourner sur l'axe X vers la droite pendant 850 ms et sur l'axe Y nécessite de déplacer le bras vers le haut avec le moteur 5M2 pendant 250Ms.

Nous pouvons alors remplacer les générateurs d'impulsions par des temporisateurs.

Comme il est montré sur figure ci-dessous :

La figure ci-dessous représente le temps nécessaire au bras pour se déplacer vers chaque position sur l'axe X, l'unité de temps est la milliseconde (Ms).

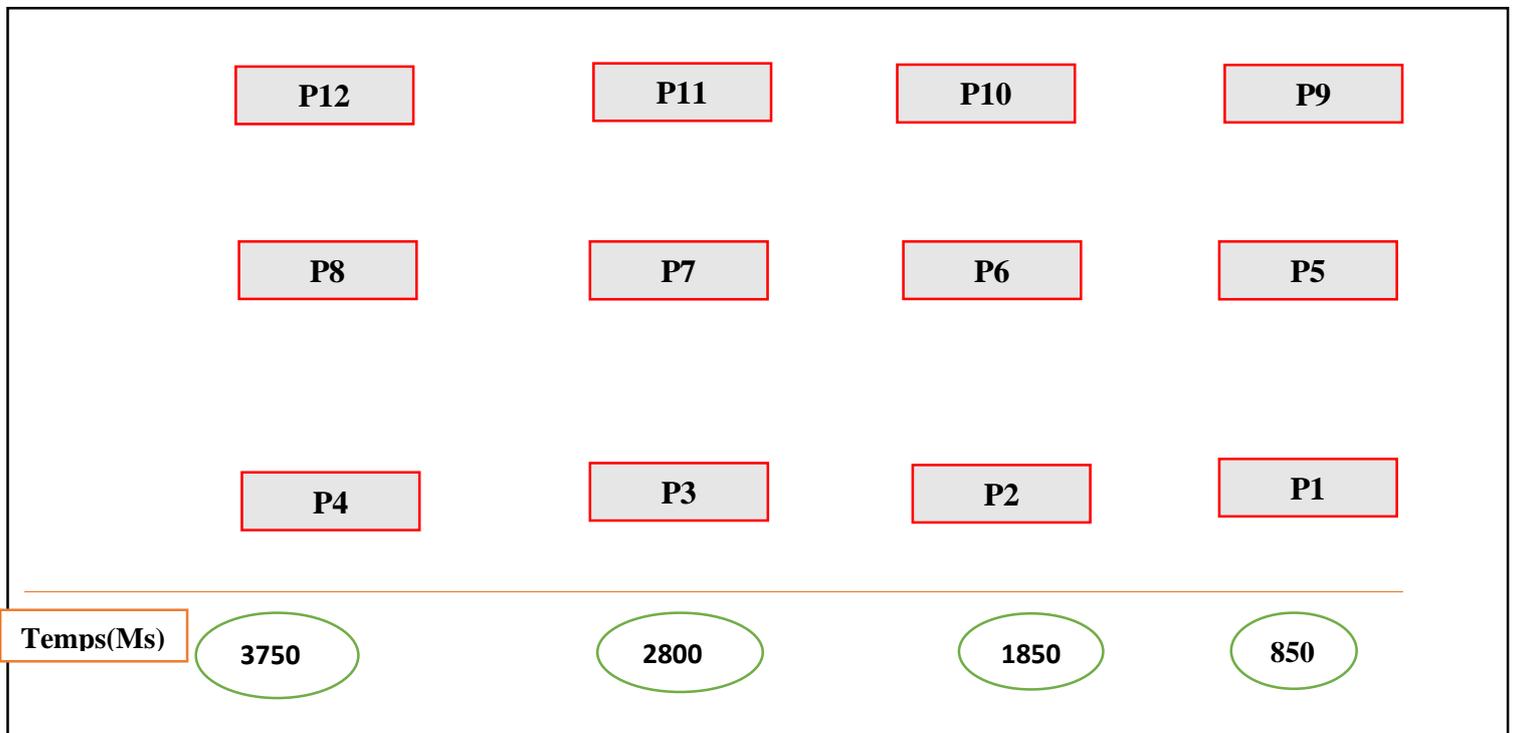


Figure III-15: Temps de déplacements sur l'axe X

Le temps requis pour lever le bras pour chaque position est illustrée sur la figure ci-dessous :

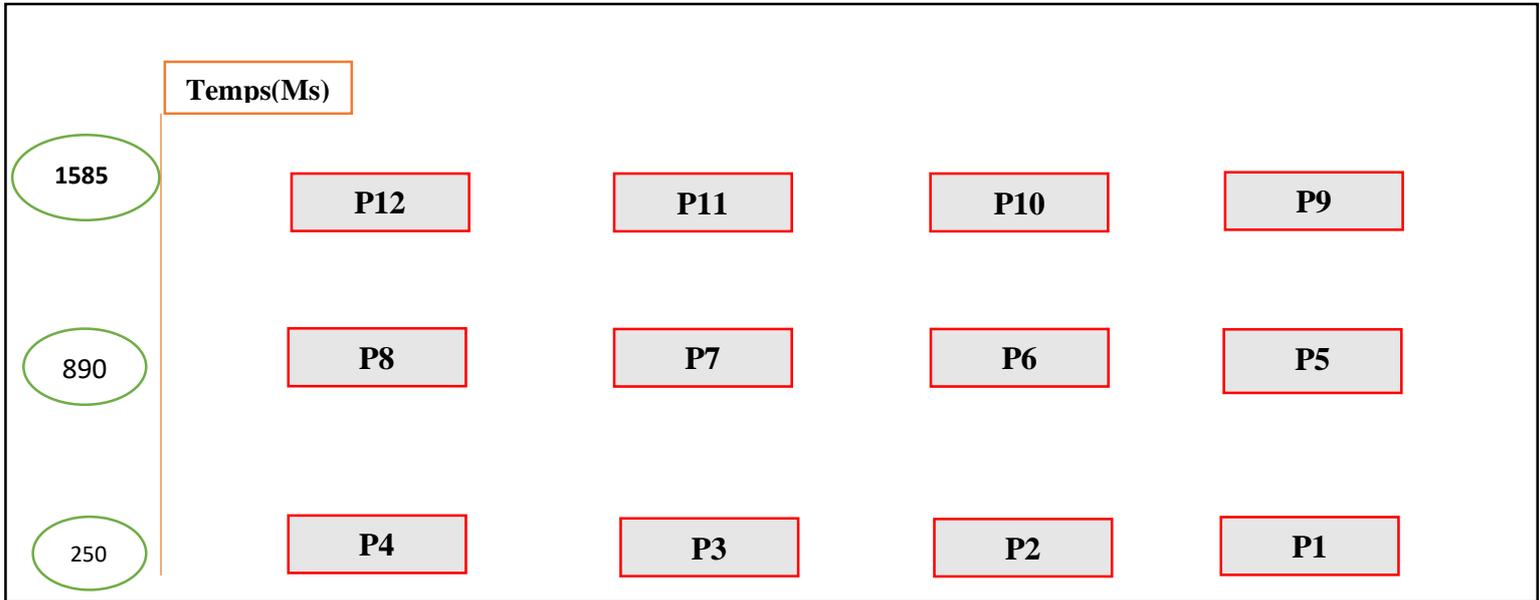


Figure III-16: Temps de déplacements sur l'axe Y

❖ **Programmation du bouton p1 (position 1) avec temporisateur:**

Pour la programmation du bouton P1 nous avons utilisé le temporisateur pour le déplacement du bras est illustrée sur la figure ci-dessous :

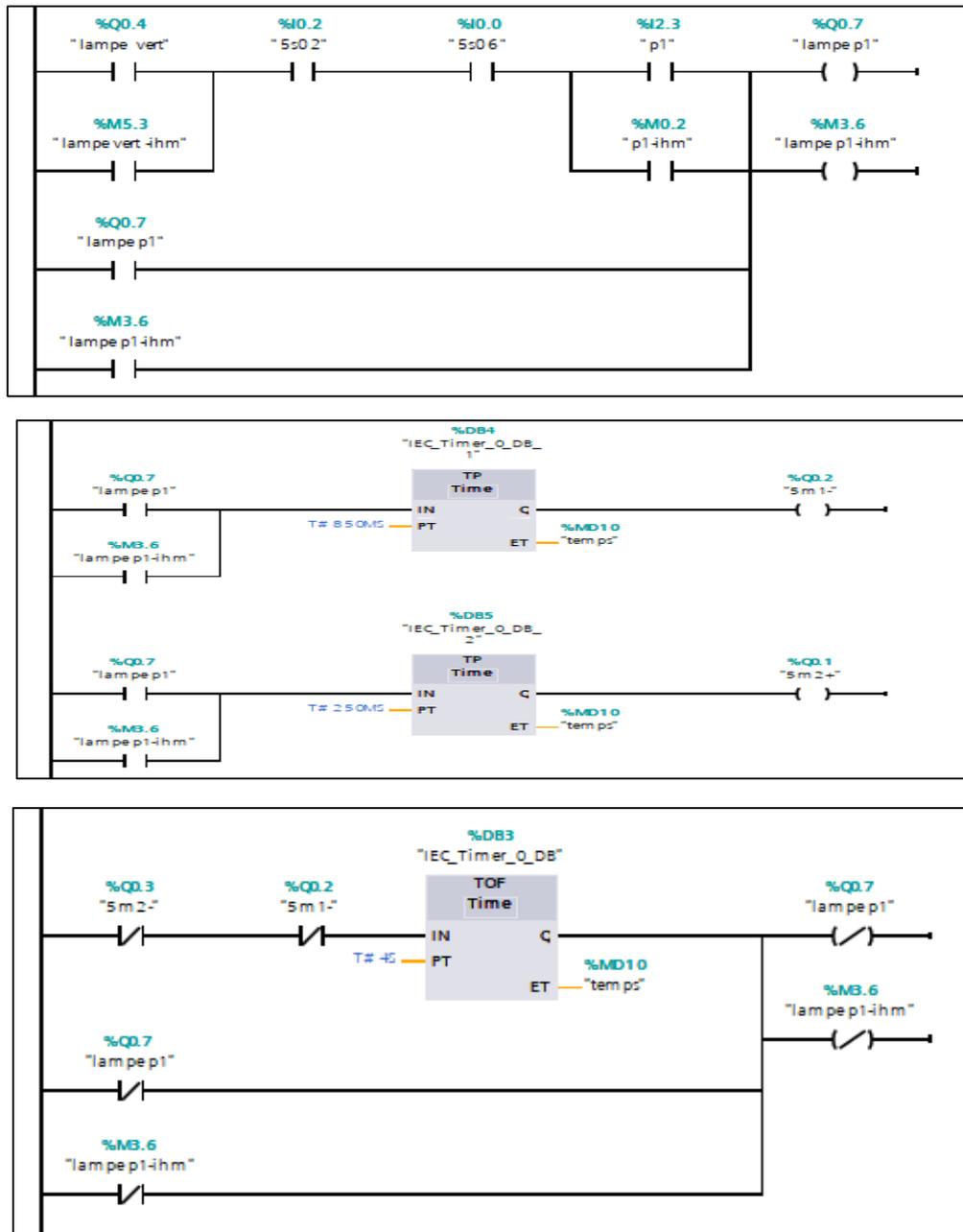


Figure III-17: Programme associé au bouton p1 avec temporisateur

III.6 Simulation du programme:

Après avoir lancé la visualisation nous avons vérifié notre programme et effectué un test. Cette opération est effectuée à l'aide de PLCSim, un contrôleur virtuel qui émule les automates.

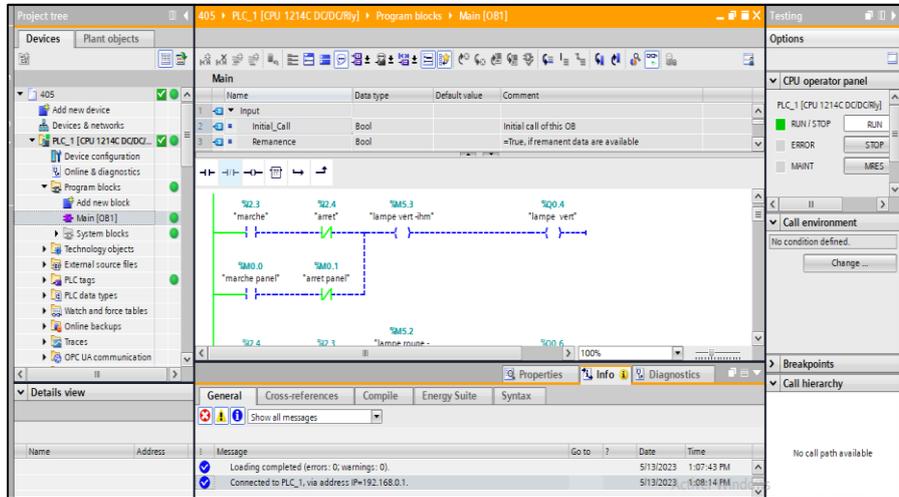


Figure III-18: Simulation du programme LD

III.7 Conclusion:

Au cours de ce chapitre, nous avons introduit les langages de programmation dans les PLC et nous avons développé un programme Ladder (LD) pour notre module d'emmagasinage à contrôle cartésien. Comparé à d'autres langages, celui-ci est le plus approprié pour le contrôle de systèmes complexes car il se distingue par sa grande précision et une facilité de contrôle.

Le dernier chapitre à venir, nous présentons ce que nous avons réalisé, notre interface et le câblage de notre système.

IV. CHAPITRE 4

Pilotage et supervision du MOD.MCS-620/EV

IV.1 Introduction:

Le monde industriel d'aujourd'hui est devenu plus complexe et les travaux de contrôle et de surveillance ainsi que de diagnostic et de maintenance rencontrent souvent d'énormes difficultés dans ce domaine, mais actuellement avec le développement de l'informatique industrielle, le recours à la supervision peut être une solution adaptée.

IV.2 Définition de la supervision:

Il s'agit d'une forme avancée de communication homme-machine comportant de nombreux avantages pour le processus de production industrielle. Elle assiste l'opérateur dans la surveillance de l'état de fonctionnement du processus et permet également son contrôle. [54]

Elle permet également d'obtenir une visualisation en temps réel de toutes les étapes du processus. à contrôler et elle révèle tous les problèmes attendus lors du travail dans une unité industrielle.[54]

IV.3 Fonctionnement du supervision :

Il s'agit de surveiller l'état d'exécution du processus pour atteindre son point de fonctionnement optimal. L'objectif est d'obtenir une visualisation en temps réel de l'état d'avancement des paramètres du procédé et cela offre à l'opérateur la possibilité de prendre des décisions rapides adaptées à leurs objectifs tels que notre rythme de production et la qualité de nos produits ainsi que la sécurité des biens et des personnes. [54]

IV.4 Modules fonctionnels d'un système de supervision:

Il est constitué d'un logiciel (moteur central) à laquelle sont rattachées des données d'équipements l'automatisés... etc. [54]

Parmi les modules fonctionnels principaux d'une supervision on a :

- L'historique de toutes les données.
- La gestion de toutes les alarmes et les événements.
- l'obtention des données du processus via l'unité de contrôle (l'automate programme).
- l'archivage et restauration des données à des fins d'analyse et de maintenance.

IV.5 Logiciel de supervision:

SIMATIC sont cœur de l'écosystème « Integrated Automation », il comprend une variété de produits et de systèmes flexibles et Comme il supervise un processus avec SIMATIC WinCC. [54]

IV.6 Simatic Step 7 TIA Portal:

IV.6.1 Généralité sur TIA portal:

Il a été conçu pour mettre en œuvre des solutions d'automatisation à l'aide d'un système d'ingénierie intégré incluant les logiciels SIMATIC STEP 7, SIMATIC WinCC, ainsi que STARDRIVE. [54]

IV.7 Présentation du logiciel de supervision:

Le logiciel WinCC dans TIA Portal est basé sur une architecture logicielle de portail d'automatisation entièrement intégrée, il fournit une solution cohérente et efficace aussi facile à utiliser pour toutes nos tâches d'automatisation. [54]

IV.8 IHM :

C'est un ensemble matériel et logiciel qui permet à l'opérateur la communication avec le système informatique. En quelques décennies, l'IHM et machine a connu un très grand développement, démarrage avec de simples boutons poussoirs et afficheur 7 segments jusqu'à des écrans LCD avec différentes bandes. [54]

IV.8.1 Fonctions d'IHM intégrées:

Parmi les différentes fonctions IHM intégrées à usage industriel, on retrouve :

- ❖ Visualisation graphique complète des processus et des états de processus. [54]
- ❖ Signalisation et reconnaissance des événements. [54]
- ❖ Placé des bases de données de processus: les valeurs mesurées et les messages Archivés. [54]
- ❖ Enregistrer les données de processus et les données déjà archivées. [54]
- ❖ Gérer les droits d'accès des utilisateurs. [54]

IV.8.2 Tableau IHM :

Name	Data type	Connection	PLC name	PLC tag
ind 2	Bool	HMI_Connectio...	PLC_1	"lampe rouge -ihm"
ind 3	Bool	HMI_Connectio...	PLC_1	"lampe bleu -ihm"
arretpanel	Bool	HMI_Connectio...	PLC_1	"arret panel"
b1	Bool	HMI_Connectio...	PLC_1	"p1-ihm"
ind 1	Bool	HMI_Connectio...	PLC_1	"lampe vert -ihm"
lampe 1	Bool	HMI_Connectio...	PLC_1	"lampe p1-ihm"
marchepanel	Bool	HMI_Connectio...	PLC_1	"marche panel"
reset b	Bool	HMI_Connectio...	PLC_1	"reset-ihm"
b2	Bool	HMI_Connectio...	PLC_1	"p2-ihm"
b3	Bool	HMI_Connectio...	PLC_1	"p3-ihm"
b4	Bool	HMI_Connectio...	PLC_1	"p4-ihm"
b5	Bool	HMI_Connectio...	PLC_1	"p5-ihm"
b6	Bool	HMI_Connectio...	PLC_1	"p6-ihm"

Name	Data type	Connection	PLC name	PLC tag
b12	Bool	HMI_Connectio...	PLC_1	"p12-ihm"
lampe 2	Bool	HMI_Connectio...	PLC_1	"lampe p2-ihm"
lampe 3	Bool	HMI_Connectio...	PLC_1	"lampe p3-ihm"
lampe 4	Bool	HMI_Connectio...	PLC_1	"lampe p4-ihm"
lampe 5	Bool	HMI_Connectio...	PLC_1	"lampe p5-ihm"
lampe 6	Bool	HMI_Connectio...	PLC_1	"lampe p6-ihm"
lampe 7	Bool	HMI_Connectio...	PLC_1	"lampe p7-ihm"
lampe 8	Bool	HMI_Connectio...	PLC_1	"lampe p8-ihm"
lampe 9	Bool	HMI_Connectio...	PLC_1	"lampe p9-ihm"
lampe 10	Bool	HMI_Connectio...	PLC_1	"lampe p10-ihm"
lampe 11	Bool	HMI_Connectio...	PLC_1	"lampe p11-ihm"
lampe 12	Bool	HMI_Conne...	PLC_1	"lampe p12-ihm"

Figure IV-1: Tableau des entrées et sortie

IV.8.3 IHM créé:

Les boutons marche et arrêt et le bouton de réinitialisation contrôlent notre systèmes à partir de l'interface graphique, Avec nos 12 positions (P1, P2,..., P12) dans l'interface, chacune avec un indicateur LED, nous pouvons contrôler le déplacement du bras dans le système.

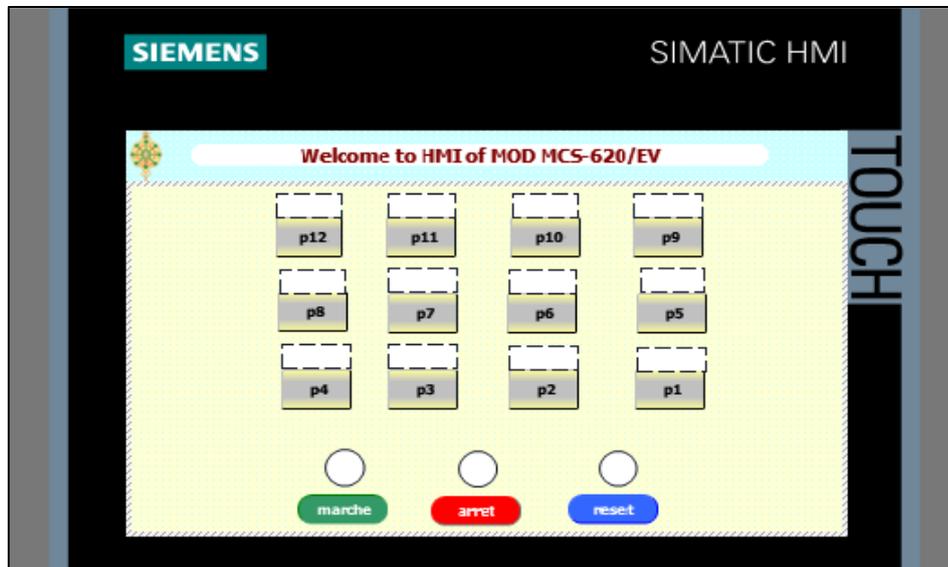


Figure IV-2: Interface IHM du MOD/MCS -620

Une fois que vous appuyez sur le bouton d'alimentation « Marche » une LED verte s'allume indiquant que le système est allumé.

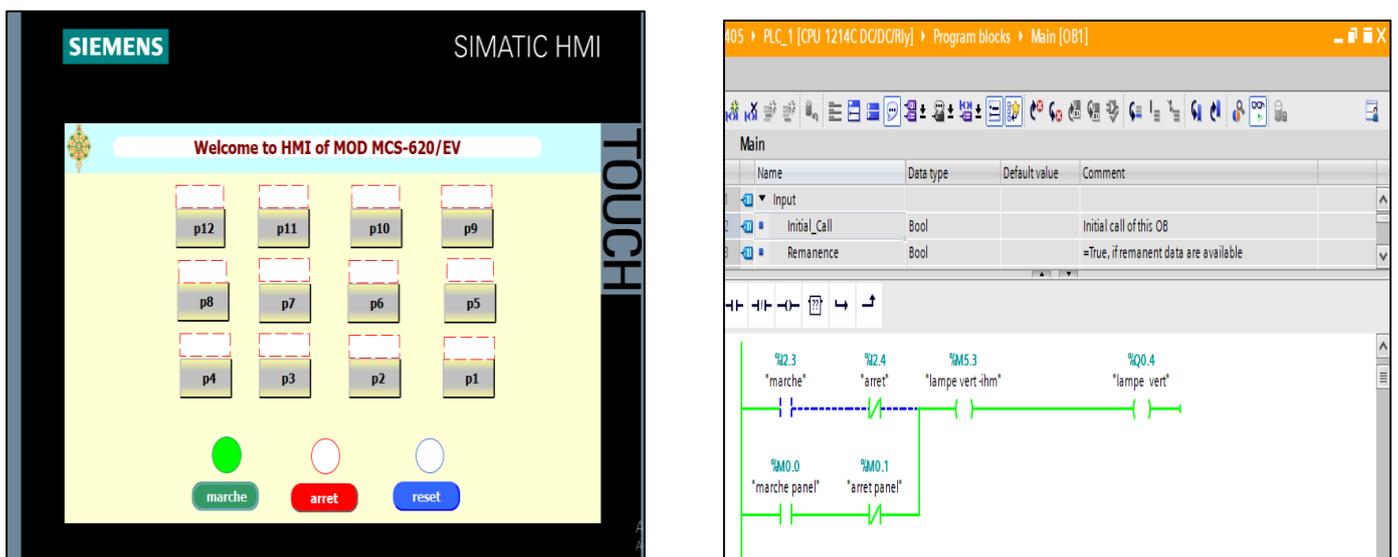


Figure IV-3: Simulation du bouton marche

Pour arrêter le système, nous appuyons sur le bouton « arrêt », nous voyons une LED rouge, Comme l'indiquent les deux figures ci-dessous :

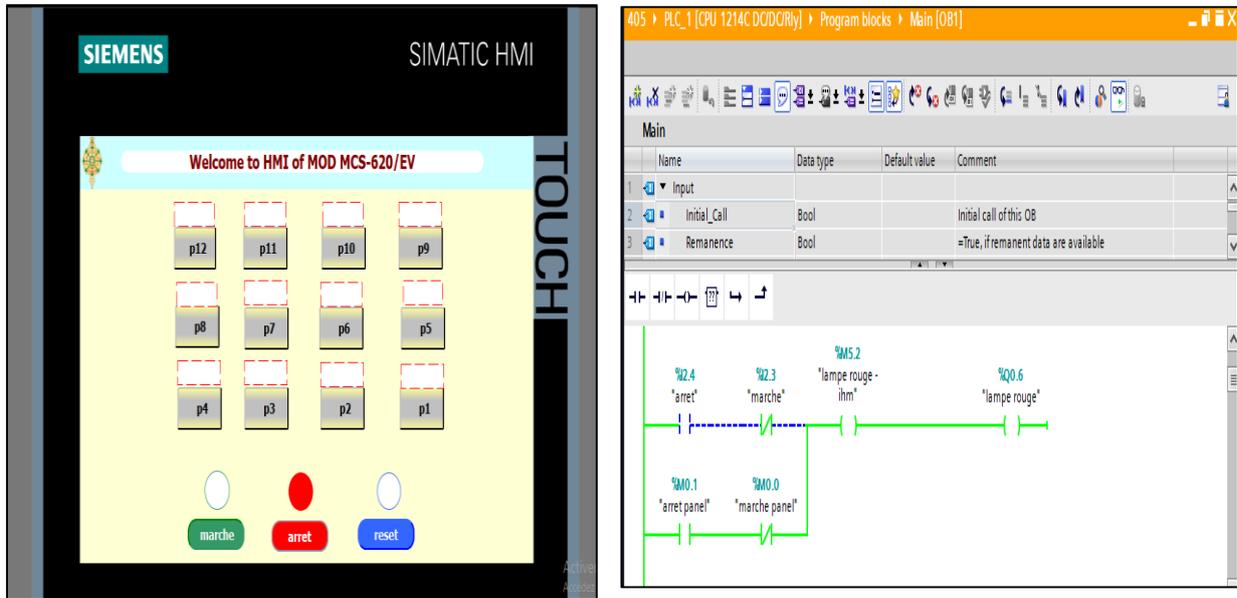


Figure IV-4: Simulation du bouton arrêt

IV.8.4 Pc système :

Après avoir créé IHM virtuelle et pour contrôler réellement notre système, nous avons d'abord créé PC système pour commander notre système dans la réalité.

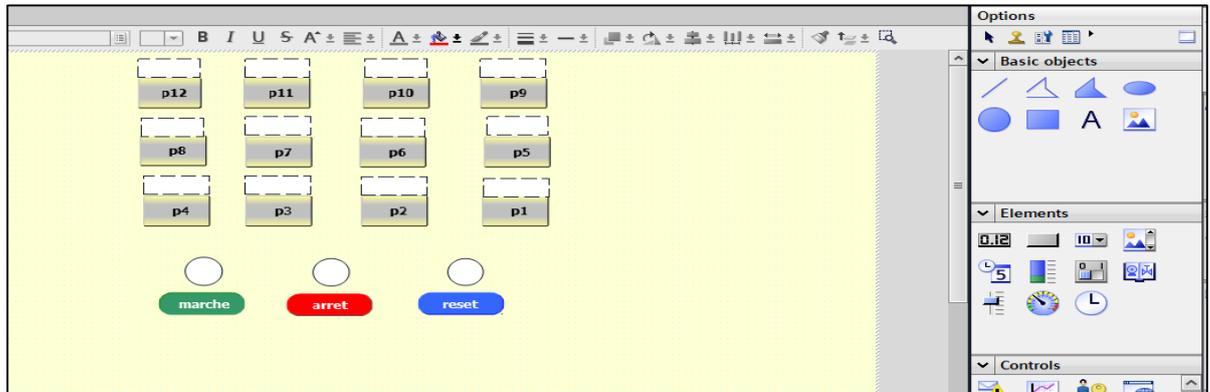


Figure IV-5: Création de notre interface

Connexion du Pc système et l'API:

Après avoir créé notre interface, nous devons d'abord faire la connexion entre PC système et l'API afin que nous puissions passer à l'étape de commande.

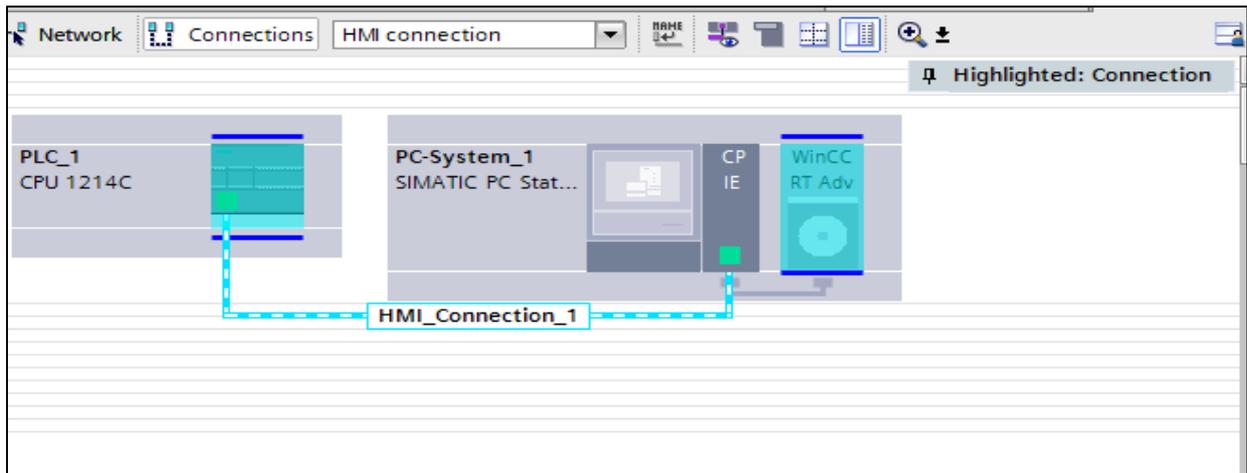


Figure IV-6: Connexion du Pc système et notre API

IV.9 Câblage de notre système:

Après la création de notre pc système nous avons passé au câblage pour piloter notre système selon l'interface, nos différents composants du système sont liés à des entrées et sorties de l'API selon le tableau des variables de programme.

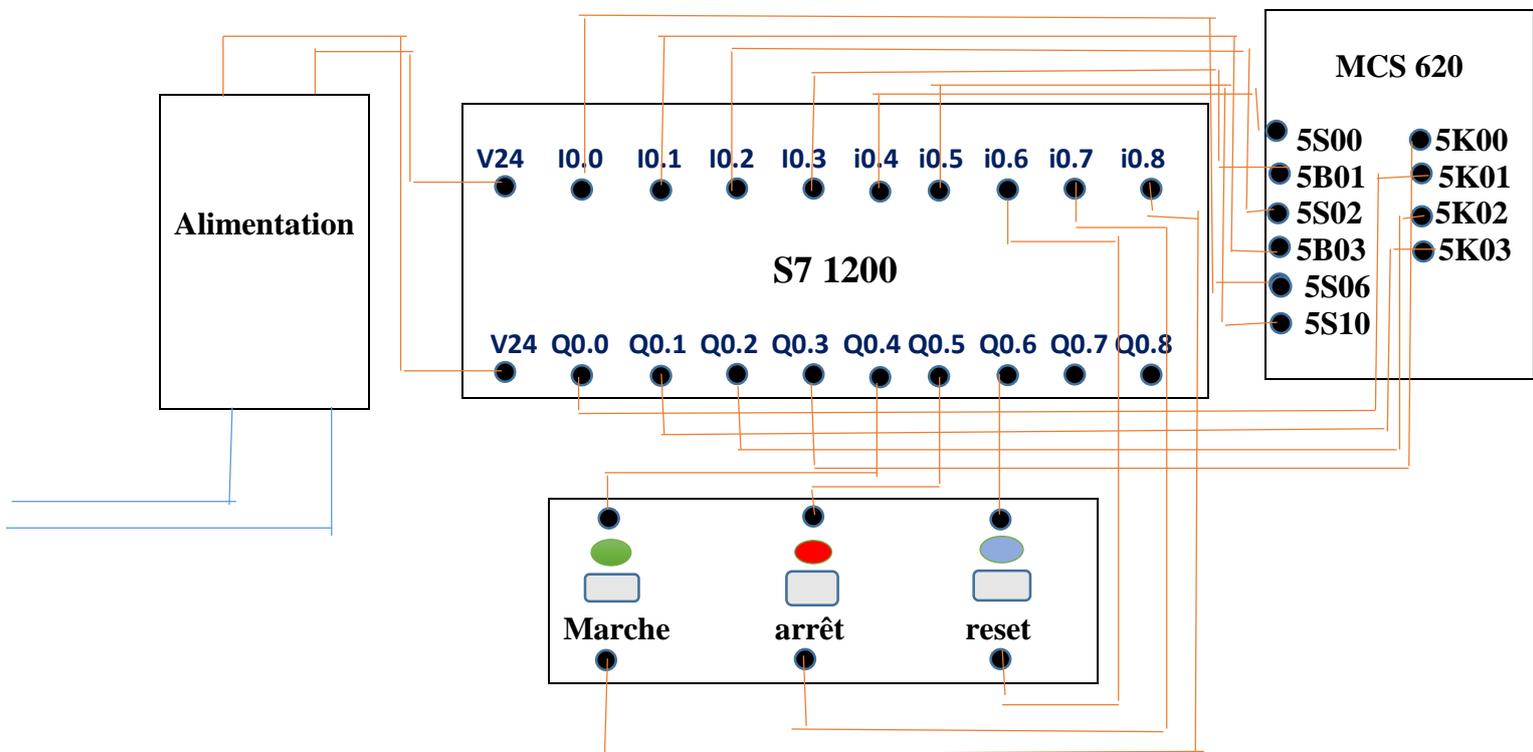


Figure IV-7: Schéma représentant le câblage de notre système



Figure IV-8: Câblage de notre système

IV.10 Conclusion:

Nous avons débuté tout d'abord notre chapitre en donnant quelques concepts sur la supervision et son rôle dans le domaine industriel. Ensuite, nous avons donné des informations générales sur le logiciel TIA PORTAL

Le chapitre actuel inclut une représentation visuelle du système MOD MCS-620/Ev qui a été effectuée en suivant les étapes précises décrites dans le fonctionnement

À la fin, nous avons présenté l'IHM d'interface homme-machine pour système.

Conclusion générale

Suite à l'étude de la supervision d'un système industriel dans ce mémoire, il est évident que les systèmes de supervision jouent un rôle vital dans la surveillance et le contrôle des équipements industriels.

L'utilisation de ces systèmes comporte plusieurs avantages, notamment la visualisation de l'état physique et fonctionnel des équipements ainsi que la centralisation de la surveillance sur une large zone et l'usage de protocoles de communication normalisés ou propriétaires pour connecter les interfaces requis.

Cependant, il convient de noter que la mise en place d'un système de suivi ne pas considérer cela comme une solution unique, Il est essentiel de prendre en compte plusieurs facteurs clés tels que les caractéristiques spécifiques de l'industrie et les objectifs et les besoins de l'entreprise ainsi que les coûts associés à la configuration et à la maintenance du système.

Cependant, La surveillance des systèmes industriels est une tâche fastidieuse qui nécessite une attention constante et une planification minutieuse, les systèmes de supervision peuvent être très bénéfiques pour les entreprises s'ils sont installés de manière judicieuse et utilisés efficacement et Ils peuvent permettre de favoriser l'optimisation de l'efficacité opérationnelle de réduire les temps d'arrêt et d'optimiser l'utilisation des équipements industriels.

Nous avons pu automatiser notre système MCS620 .Nous avons commencer par l'étape de définir le fonctionnement . Par la suite nous avons utiliser langage Ladder pour programmer notre système. Après nous avons crée une interface du PC système.Ala fin nous avons passer au cablage du système pour atteindre notre objectif de commander notre système depuis l'interface.

Enfin, Ce projet nous a été bénéfique à bien des égards car cela nous a donné l'opportunité d'explorer l'univers de l'industrie et d'appliquer les connaissances théoriques acquises lors de notre cursus universitaire.

De plus, cela nous a permis d'apprendre les différentes étapes indispensables à la conception de projets d'automatisation.

Références

- [1] : “l’Industrie”,(Enligne). Disponible : www.insee.fr .
- [2] : “ la révolution industrielle” ,,(Enligne). Disponible : www.geo.fr
- [3] : “ Industrie 1.0 ” ,,(Enligne). Disponible: www.forcam.com
- [4] : “ Industrie 2.0 ” ,,(Enligne). Disponible: www.forcam.com
- [5] : “ Industrie 1.0 ” ,,(Enligne). Disponible: www.lobry-automatisme.fr
- [6] : “ Industrie 2.0 ” ,,(Enligne). Disponible: www.forcam.com
- [7] : “ Industrie 4.0 ” ,,(Enligne). Disponible: www.lobry-automatisme.fr
- [8] : “ l’avantage d’industrie (4.0) ” , (Enligne). Disponible: www.integral-system.fr
- [9] : “ l’Automatisme industriel” , (Enligne). Disponible: www.iotindustriel.com .
- [10] : “ -Objectif de l’automatisation _d’un système de production” , (Enligne). Disponible: www.sites.google.com
- [11] : “ Communication industrielle” , (Enligne). Disponible: www.phoenixcontact.com
- [12] : “ PROFINET” , (Enligne). . Disponible: www.burkert.fr
- [13] : “ PROFIBUS ” , (Enligne).. Disponible: <https://www.profibus.com/>
- [14] : “ LA DIFFÉRENCE ENTRE PROFIBUS _ PROFINET” , (Enligne). Disponible: ww.us.profinet.com
- [15] : “MODULE D’EMMAGASINAGE A CONTRÔLE CARTÉSIEN Mod. MCS-620/EV” (Enligne). Disponible:www.electronicaaveneta.com
- [16] : “ Capteur” (Enligne). Disponible: www.techno-science.net
- [17] : “ Structure générale d’un système automatisé et éléments d’automatisme” (Enligne). Disponible : www.integral-system.fr , p.66
- [18] : “Les Capteurs” , (Enligne). Disponible: www.projet.eu.org/pedago/sin/1ere/2-capteurs.pdf p.12
- [19] : “Les capteurs mécaniques ou interrupteurs de position” , (Enligne). Disponible: https://sti2d.ecolelamache.org/les_capteurs_mcaniques_ou_interrupteurs_de_position.html
- [20] : “Capteur De Proximité Inductif” , (Enligne). Disponible: www.omch.com
- [22] : “ Capteur capacitif ” , (Enligne). Disponible: <https://www.eeca.eu/capteur-capacitif-quest-ce-que-cest%E2%80%89/>
- [23]: “DétecteurDeProximitéMagnétique” , (Enligne). Disponible: www.omch.com
- [24] : “Capteur De Proximité Magnétique” , (Enligne). Disponible: www.omch.com

- [25] : “ Capteurs photoélectriques”, (Enligne). Disponible: www.tecnoland.fr
- [26] : “ capteurs optiques”, (Enligne). Available :www.asrc.fr
- [27] : “capteurs optiques”, (Enligne). Disponible: www.dimelco.com
- [28] : “ Analyse de la chaîne d’information, Information analogique ou numérique”, (Enligne).Disponible:<file:///C:/Users/Hadjer/Downloads/Cours%20-%20Caract%C3%A9risation%20de%20l'information%20analogique%20et%20num%C3%A9rique.pdf>
- [29] : “ les capteurs”,(Enligne). Disponible: https://sti.discip.ac-caen.fr/IMG/pdf/Generalites_sur_les_capteurs.pdf,
- [30] : “ Accéléromètre “, (Enligne). Disponible: www.journaldunet.fr
- [31] : “Capteur d'accélération”, (Enligne). Disponible: www.directindustry.fr
- [32] : “ Capteur de lumière (LDR)”, (Enligne). Disponible: www.carglass.be.fr
- [33] : “ Capteurs de pression “, (Enligne). Disponible: www.hbm.com
- [34] : “ Capteurs à ultrasons”, (Enligne). Disponible: www.digikey.fr
- [35] : “ Capteur ultrasons ”, (Enligne). Disponible: www.automatismes-solutions.com
- [36] : “ Thermocouple” , (Enligne). Disponible: www.omega.fr
- [37] : “ Les thermocouples : Définition et fonctionnement “, (Enligne). Disponible: www.blog.beamex.com
- [38] : H. Belgacem, A Rais, “ *Système de contrôle distribué (DCS) _avec l’exploitation de l’automate programmable AC800F(ABB)* “ mémoire de master en Automatique Avancée, Université Mohamed Khi der-Biskra , p.128, (Enligne). Disponible: www.archives.univbiskra.dz
- [39] : W. Bolton, traduction de Hervé Soulard “ Automates Programmables_ Industriels” (Enligne). Available: www.excerpts.numilog.com , p.30
- [40] : A. Gonzaga ,“*les automates _programmables industriels* “, (Enligne). Disponible: www.geea.org/ES_AUT_TR.com , p.17
- [41] : “ How PLCs _Work “, (Enligne). Disponible: www.plcdev.com
- [42] : W. Bolton, “ Programmable_ Logic Controllers “, Fourth Edition, Published by Elsevier Newnes, (Enligne). Disponible: www.etf.ues.rs.ba.com , p.303
- [43] : “Von Neumann_ Architecture”, (Enligne). Disponible: www2.cs.siu.edu.com , p.66
- [44] : “ Les automates programmables_ industriels “, [Online]. Disponible: www.technologuepro.com ,
- [45] : A. Daghmous, H. Loucif, “ *Commande _et Supervision d’un système industriel par Automate programmable* “, Mémoire de Master_ en Automatique et Systèmes, Université Larbi Tebessi– Tébessa, p.99, 2017/2018.

- [46] : “ *All About PLC _Analog Input and Output Signals and Programming*”, (Enligne). Disponible: www.plcacademy.com .
- [47] : ‘ *PLC _Hardware*’, (Enligne). Disponible: <https://learn.realpars.com>,
- [48] : “ Introduction_ sur les systèmes à base de microprocesseur “, (Enligne).. Disponible: www.technologuepro.com
- [49] : H. Bensidhoum, L. Ayadi, “*Automatisation et_ supervision d’une station de purification des eaux usées* “, Mémoire_ de fin de cycle pour l’obtention du diplôme de MASTER en AUTOMATIQUE, Université A/MIRA de Bejaia_ Faculté de technologie, Département Automatique, Télécommunication et Electronique
- [50] : (Enligne). _Available: www.technologuepro.com ,
- [51] : Ir. H. Lecocq, “*Les automates programmables, Tome I, Caractéristiques et méthodologie de programmation* “, Université de Liège, Faculté des Sciences Appliquées, Dernières mises à jour 2005
- [52] : BERKAINE Ismail “ *Etude, amélioration et supervision d’un système de nettoyage CIP pour un cuiseur stérilisateur_ du fromage*“, Master professionnel _en automatique et informatique industrielles, Université Mouloud MAMMARI de Tizi-Ouzou, p.137 ,2016
- [53] : M. ELHACHEMI, T, IBRAHIM, “ *Les langages _de programmation de l’automate programmable industriel* “, Mémoire de MASTER en_ Commande des Machines Electrique, UNIVERSITE d’ADRAR, p.73, 2017
- [54] : “ *Automate programmable S7-1200*“, (Enligne). Disponible: media.automation24.com ,p1074
- [55] : Z .Imad Eddine, Y.Youcef “ *Supervision_ d’un niveau d’eau via un automate programmable : Etude et Conception* “, université IBN-KHALDOUN DE TIARET faculté DES SCIENCES APPLIQUEES.
- [56] : (Enligne). _ Disponible: www.si.blaisepascal.fr

Résumé:

Ce projet est concentré sur le pilotage et l'automatisation du module mcs-620/EV disponible dans le laboratoire de Tlemcen MELT. Dans un premier temps, nous avons examiné la structure et le mode de fonctionnement du système. Du module MCS 620-Ev. Nous avons aussi programmé ces fonctions telles que le déplacement du système MCS-620/EV et le déstockage du produit désiré en programmant à l'aide du langage Ladder. Nous avons utilisé plusieurs réseaux pour les fonctions du système. Par la suite Nous avons créé IHM. pour superviser et contrôler notre système grâce à des boutons (marche, arrêt et reset, avec les 12 positions) pour faciliter la manipulation du système grâce au WinCC dans TIA PORTALE. À la fin, nous avons câblé notre système à l'API S7-1200 et nous avons chargé notre programme depuis un ordinateur.

Mots clés: WinCC, IHM, TIA PORTALE, supervision, S7-1200.

Abstract:

This project is focused on the control and automation of the mcs-620/EV module available in the Tlemcen MELT laboratory. First, we examined the structure and mode of operation of the system. From the MCS 620-Ev module. We have also programmed these functions such as the movement of the MCS-620/EV system and the removal of the desired product by programming using the Ladder language. We used multiple networks for system functions. Subsequently We created HMI. to supervise and control our system using buttons (start, stop and reset, with 12 positions) to facilitate handling of the system using WinCC in TIA PORTALE. In the end, we wired our system to the S7-1200 PLC and downloaded our program from a computer

Keywords: WinCC, IHM, TIA PORTALE, supervise, s7 1200.

ملخص:

يتعلق هذا المشروع بالتحكم والتشغيل الآلي لوحدة mcs-620 / EV المتوفرة في مختبر MELT في تلمسان. أولاً، قمنا بفحص هيكل وطريقة تشغيل النظام. من وحدة MCS 620-Ev. لقد قمنا أيضاً ببرمجة هذه الوظائف مثل نقل نظام MCS-620 / EV وإزالة المنتج المطلوب عن طريق البرمجة باستخدام لغة السلم. استخدمنا شبكات متعددة لوظائف النظام. بعد ذلك، أنشأنا HMI. للإشراف على نظامنا والتحكم فيه باستخدام الأزرار (بدء، وإيقاف، وإعادة ضبط، مع 12 موضعاً) لتسهيل التلاعب بالنظام باستخدام WinCC في TIA PORTALE. أخيراً، قمنا بتوصيل نظامنا بـ S7-1200 PLC وقمنا بتنزيل برنامجنا من جهاز كمبيوتر

الكلمات المفتاحية:

WinCC, IHM, TIA PORTALE, s7 1200.