

République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieure et de la Recherche Scientifique



جامعة بجاية
Tasdawit n Bgayet
Université de Béjaïa

Université Abderrahmane Mira - Bejaia

Faculté de Technologie



Département d'Automatique, Télécommunication et d'Electronique

Projet de Fin d'Etudes

Pour l'obtention du diplôme de Master

Filière : Automatique

Spécialité : Automatique et informatique industrielle

Thème

**Étude et automatisation du compresseur d'air SIAD 5 litres
Polyéthylène-Téréphtalate (Entreprise LABELLE – Bejaia)**

Préparé par :

Mlle. SENOUNE Dihia

Mlle. BACHA Lydia

Dirigé par :

Dr. A. OUARET

Mr. S. HOCINI

Examiné par :

Dr. N. NAIT MOHAND

Dr. H. LEHOUCHE

Année universitaire : 2022/2023

Remerciements

En ce moment chargé d'émotions et de fierté, nous remercions ALLAH tout puissant de nous avoir donné la volonté, la santé et la patience durant toutes ces longues années pour réaliser ce modeste travail.

Tout d'abord, nous adressons nos sincères remerciements à notre promoteur Mr OUARET AHMED, pour avoir accepté de guider nos travaux et pour son dévouement constant tout au long du processus. Ses connaissances et ses conseils ont été essentiels à notre réussite, et nous sommes honorés d'avoir pu bénéficier de sa supervision.

Nous adressons nos sincères remerciements à notre Co-promoteur, Mr. HOCINI SOFIANE, pour sa disponibilité sans faille, son soutien inconditionnel et pour avoir mis à notre disposition les ressources nécessaires à la réalisation de notre projet. Sa contribution a été inestimable et nous lui en sommes sincèrement reconnaissants.

Nous tenons à remercier chaleureusement le président et les membres du jury d'avoir accepté d'évaluer notre travail. Leur expertise et leurs commentaires constructifs nous ont permis de progresser et de perfectionner notre travail.

Enfin, nous voulons exprimer notre gratitude envers tous ceux qui, de près ou de loin, nous ont apporté leur aide, leurs conseils et leur soutien tout au long de notre parcours d'études. Votre présence et votre contribution ont été d'une importance capitale, et nous sommes sincèrement reconnaissants de vous avoir à nos côtés.



Dédicaces

Je dédie ce travail aux personnes les plus importantes à mes yeux

À

*Mes très **chers et précieux** parents qui m'ont toujours soutenue, et à l'intérêt qu'ils ont toujours porté pour mes études, je ne les remercierai jamais assez pour tout ce qu'ils font.
Que dieu me les protèges.*

À

*Mes chers frères, **Akli et Aris***

À

*Mes très chers grands-parents, **Amirouche, Louisa, et Rekia**, ainsi qu'à mon cher et regretté grand-père, **Belaid**. Votre amour inconditionnel a été ma plus grande force.*

À

Toute ma famille sans exception

À

*Mon binôme, **Dhia***

À

Touts mes amis(es) et toutes les personnes que j'aime, vous avez été à mes côtés aux moments les plus difficiles. Je vous remercie

À

À moi-même, je dédie ce travail acharné, cette détermination qui m'a permis d'atteindre ce jalon important de ma vie. Je suis fière de mon parcours et des obstacles que j'ai surmontés pour en arriver là.



Lydia



Dédicaces

Je dédie ce mémoire

À mes chers parents, que nulle dédicace ne puisse exprimer mes sincères sentiments et mon éternelle gratitude, pour leur patience illimitée, leurs encouragements continus, leur aide, en témoignage de mon profond amour et respect pour leurs grands sacrifices

À ma chère sœur : Thiziri

À mes frères : Lounis et Yacine

pour leur soutien et leur compréhension. Ils n'ont cessé d'être présents pour moi.

Sans oublier **mes ami(es) et** exceptionnellement : Kenza, Hanane, Lisa, Anaïs, Warda, Camilia, Nadine, Thinhinane, Laythmas, Idir, Sofiane, Lamine, Mohamed.

À mon binôme Lydia et toute sa famille

et tous ceux qui me connaissent de loin ou de près.

Dihia



Table des matières

Liste des abréviations.....	i
Liste des figures.....	ii
Introduction générale.....	1
Chapitre I : Présentation de l'entreprise et de la machine utilisée	
I.1. Introduction.....	2
I.2. Présentation de l'entreprise corps gras de Bejaia LABELLE.....	2
I.2.1. Description de l'entreprise.....	2
I.2.2. Historique de l'entreprise.....	2
I.2.3. Description et production de l'unité.....	3
I.2.4. Localisation géographique.....	3
I.2.5. Moyens de l'entreprise.....	4
I.2.6. Diagnostic de production de CO.G.B LABELLE.....	5
I.2.6.1. Service savonnerie.....	5
I.2.6.2. Service raffinage.....	5
I.2.6.3. Service conditionnement des huiles.....	5
I.2.6.4. Service margarinerie.....	5
I.2.7. L'unité de conditionnement d'huile de CO.G.B LABELLE.....	7
I.3. Présentation de la machine.....	7
I.3.1. Air comprimé.....	7
I.3.1.1. Avantages de l'air comprimé.....	7
I.3.1.2. Inconvénients de l'air comprimé.....	8
I.3.2. Compresseur.....	8
I.3.2.1. But de la compression.....	8
I.3.2.2. Types des compresseurs.....	9
I.3.2.3. Description d'un compresseur d'air SIAD Tempo2-950.....	11
I.3.2.4. Différentes parties d'un compresseur d'air SIAD.....	12
I.3.2.5. Etude des organes électriques.....	13
I.3.2.6. Description de fonctionnement d'un compresseur SIAD.....	16
I.3.2.7. Equipement entraîné par le compresseur.....	17
I.3.2.8. Maintenance périodique.....	17
I.4. Conclusion.....	18

Chapitre II : API et logiciels de programmation

II.1. Introduction.....	19
II.2. Systèmes automatisés.....	19
II.2.1. Définition d'un système automatisé.....	19
II.2.2. Structure d'un système automatisé.....	20
II.2.2.1. Partie opérative.....	20
II.2.2.2. Partie commande.....	20
II.3. Objectifs de l'automatisation.....	21
II.4. Généralité sur les automates programmables industriels.....	21
II.4.1. Définition d'un automate programmable industriel.....	21
II.4.2. Types d'automates.....	22
II.4.3. Architecture des automates.....	22
II.4.3.1. Structure interne.....	22
II.4.3.2. Structure externe.....	23
II.4.4. Langages de programmation pour API.....	24
II.4.5. Caractéristiques et fonctionnalités des logiciels de programmation.....	25
II.4.6. Rôle de l'API.....	26
II.4.7. Les avantages et inconvénients de l'API.....	26
II.4.8. Critère de choix d'un automate.....	27
II.4.9. Domaine d'applications.....	27
II.5. Simatic S7-1200.....	28
II.5.1. Description.....	28
II.5.2. Choix de l'automate S7-1200.....	29
II.5.3. Modules de communication.....	29
II.5.4. Mémoire.....	30
II.5.5. Les modules d'entrées/sorties.....	30
II.5.5.1. Choix des modules d'entrées/sorties.....	30
II.5.6. Régulation.....	30
II.5.7. Communication industrielle.....	31
II.5.7.1. Interface PROFINET intégrée.....	31
II.5.7.2. Mise en réseau simple.....	31
II.5.7.3. Communication avec d'autres automates et appareils IHM.....	31
II.5.8. Choix de la CPU.....	32
II.6. Logiciel de programmation STEP7.....	32
II.6.1. Création d'un projet sous STEP7.....	33

II.6.2 Différents types de blocs utilisateurs sur STEP7	33
II.7. Supervision	34
II.7.1. Définition.....	34
II.7.2. Avantages de la supervision	34
II.7.3. Logiciels de supervision	34
II.8. Présentation du logiciel de supervision WINCC flexible.....	35
II.8.1. Interface homme- machine	35
II.8.2. Principales fonctionnalités fournies par WinCC flexible	36
II.9. Conclusion.....	37

Chapitre III : Etude et programmation

III.1. Introduction	38
III.2. Fonctionnement du compresseur SIAD tempo2_950.....	38
III.3. Cahier des charges	39
III.3.1. Rôle de cahier des charges.....	42
III.4. Programmation.....	42
III.4.1. Table des variables	42
III.4.2. Blocs	45
III.4.2.1. Bloc d'organisation (OB1)	45
III.4.2.2. Bloc fonction (FC).....	46
III.4.3. Simulation.....	49
III.4.3.1. Simulation en mode manuel	50
III.4.3.2. Simulation de bloc des entrées analogiques	51
III.4.3.3. Régulation.....	51
III.4.3.4. Etats d'alarmes	55
III.4.4. Supervision	56
III.5. Conclusion.....	58
Conclusion générale.....	57
Bibliographie	58
Annexe	60

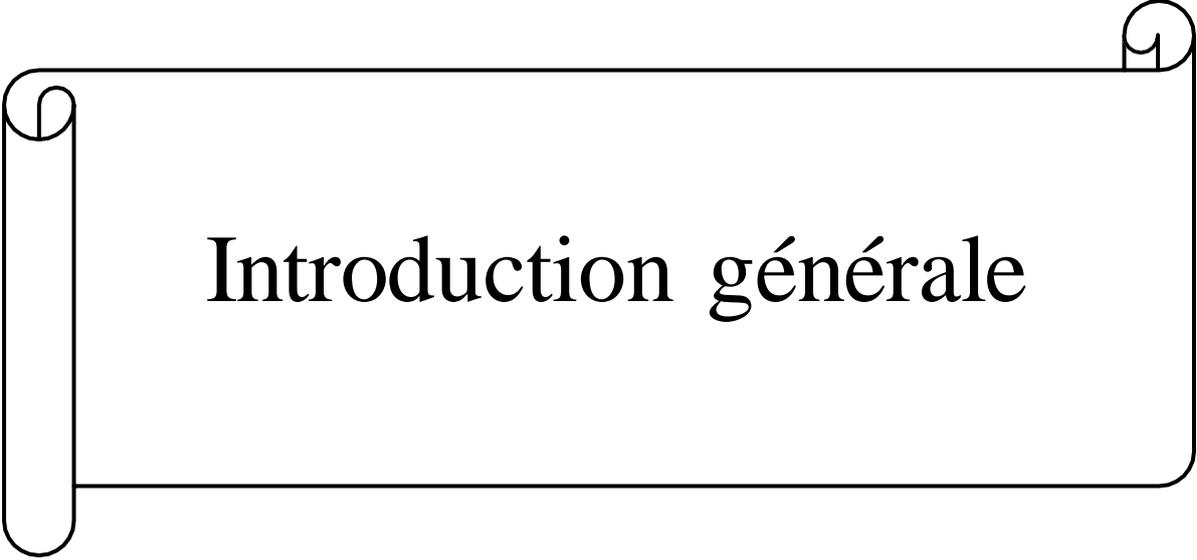
Liste des abréviations

API	Automate programmable industriel
ARU	Arrêt d'urgence
BAU	Bouton d'arrêt d'urgence
CDH	Service conditionnement des huiles
CGA	Compressed Gas Association
CO.GB	Corps Gras de Bejaïa
CPU	Central Processing Unit
DB	Data Block
ENAJUC	Entreprise Nationale des Jus et des Conserves
ENASUCRE	Entreprise Nationale du Sucre
ENCG	Entreprise Nationale des Corps Gras
FB	Function Block
FBD	Function Block Diagram
FC	Function
IHM	Interface Home Machine
IL	Instruction
LD	Langage Ladder
LED	Signalisation d'Etats et de Défaut
OB	Organization Blocks
PEDH	Polyéthylène Haute Densité
PET	Polyéthylène-Téréphtalate
PLC	Programmable Logic Controller
RAM	Random Access memory
ROM	Read only memory
SFC	Sequential Grafcet
SIAN	Société Industrielle de l'Afrique du Nord
SOGEDIA	Société de gestion et de développement des industries alimentaires
SPA	Central Processing Unit (Unité Centrale de Traitement).
ST	Structure de Texte
UP	Unité de production
SIMATIC	Siemens Automatique
MLI	Modulation de Largeur d'Impulsion
TIA	Totally Integrated Automation
TOR	Tout Ou Rien
WinCC	Windows Control Center

Liste des figures

Figure I.1 : Schéma montrant l'emplacement géographique de la CO.G.B LABELLE.....	4
Figure I.2 : Organigramme de départements de production de CO.G.B / UP	6
Figure I.3 : Etage de compression d'un compresseur à vis	10
Figure I.4 : Compresseur à piston	10
Figure I.5 : Compresseur à spirale	11
Figure I.6 : Compresseur centrifuge.....	11
Figure I.7 : Compresseur SIAD tempo2-950	12
Figure I.8 : Symbolisation du sectionneur porte fusible	14
Figure I.9 : Symbole d'un contacteur.....	14
Figure I.10 : Symbole relais thermique	14
Figure I.11 : Symbole d'un fusible	15
Figure I.12 : (a) Bouton d'arrêt d'urgence et (b) Symbole de bouton d'arrêt d'urgence	16
Figure II.1 : Structure d'un système automatisé	21
Figure II.2 : Architecture d'un API.....	22
Figure II.3 : Automate modulaire (Siemens)	24
Figure II.4 : Simatic S7-1200.....	28
Figure II.5 : Simatic S7-1200 et ses modules de communication.....	29
Figure II.6 : La CPU 1214C DC/DC/DC.....	32
Figure II.7 : Organigramme de création du projet	33
Figure II.8 : Interface IHM dans un processus automatisé	35
Figure III.1 : Table des variables	45
Figure III.2 : Bloc d'organisation	46
Figure III.3 : Mode manuel.....	48
Figure III.4 : Barre de simulation de TIA PORTAL	49
Figure III.5 : Chargement du programme dans l'appareil	50
Figure III.6 : Table de simulation par forçage	50
Figure III.7 : Démarrage de la pompe d'huile	51
Figure III.8 : Pression finale	51
Figure III.9 : Réglage de pression à 25 bars	52

Figure III.10 : Pression de 25 bars	52
Figure III.11 : Réglage de pression à 37 bars	53
Figure III.12 : Pression à 37 bars	53
Figure III.13 : Réglage de la température à 15°C	53
Figure III.14 : Température à 15°C	54
Figure III.15 : Réglage de température à 35°C	54
Figure III.16 : Température à 35°C	55
Figure III.17 : Désactivation des forçages du bouton d'arrêt d'urgence	55
Figure III.18 : Détection de défaut	56
Figure III.19 : Vue d'accueil	57
Figure III.20 : Vue de processus	57
Figure III.21 : Vue d'alarmes.....	58



Introduction générale

Introduction générale

L'automatisme peut être décrit comme un procédé qui permet d'accomplir des tâches répétitives et potentiellement dangereuses. Grâce à une précision extrême, il est capable d'effectuer des actions qui seraient impossibles pour un être humain. Afin de mettre en place un système automatisé, il est essentiel d'avoir un cahier des charges détaillé qui spécifie les tâches à accomplir et les actions à automatiser.

Les compresseurs à pistons jouent un rôle crucial dans l'industrie, fournissant de l'air comprimé. Une installation d'air comprimé comprend plusieurs composants tels qu'un moteur d'entraînement, des filtres, un réservoir d'air, une tour de refroidissement. Ainsi, maîtriser ces éléments est essentiel pour assurer une utilisation efficace des compresseurs à pistons [1].

Notre étude, réalisée pendant un stage pratique au niveau de l'entreprise LABELLE, repose sur l'étude et l'analyse de toutes les étapes de fonctionnement du compresseur SIAD et l'automatisation qui se fera avec un automate programmable industriel de type S7-1200 et réalisée grâce au logiciel « TIA PORTAL V14 » de SIEMENS.

Au sein de notre étude, on va poser un problème lié à investiguer la faisabilité de remplacer les modules d'extensions défaillants de l'automate ABB par l'automate S7-1200, en se concentrant spécifiquement sur le fonctionnement du compresseur à piston. Les objectifs principaux de cette étude seront les suivants :

- Étudier les caractéristiques et les capacités de l'automate S7-1200 pour déterminer s'il peut fournir une solution de remplacement adéquate pour les modules défaillants.
- Évaluer les exigences en termes de configuration et de programmation de l'automate S7-1200 pour assurer une intégration fluide avec le compresseur à piston.
- Proposer des recommandations et des mesures d'amélioration du fonctionnement.

A cet effet, le présent mémoire est réparti en trois chapitres décrivant les volets suivant :

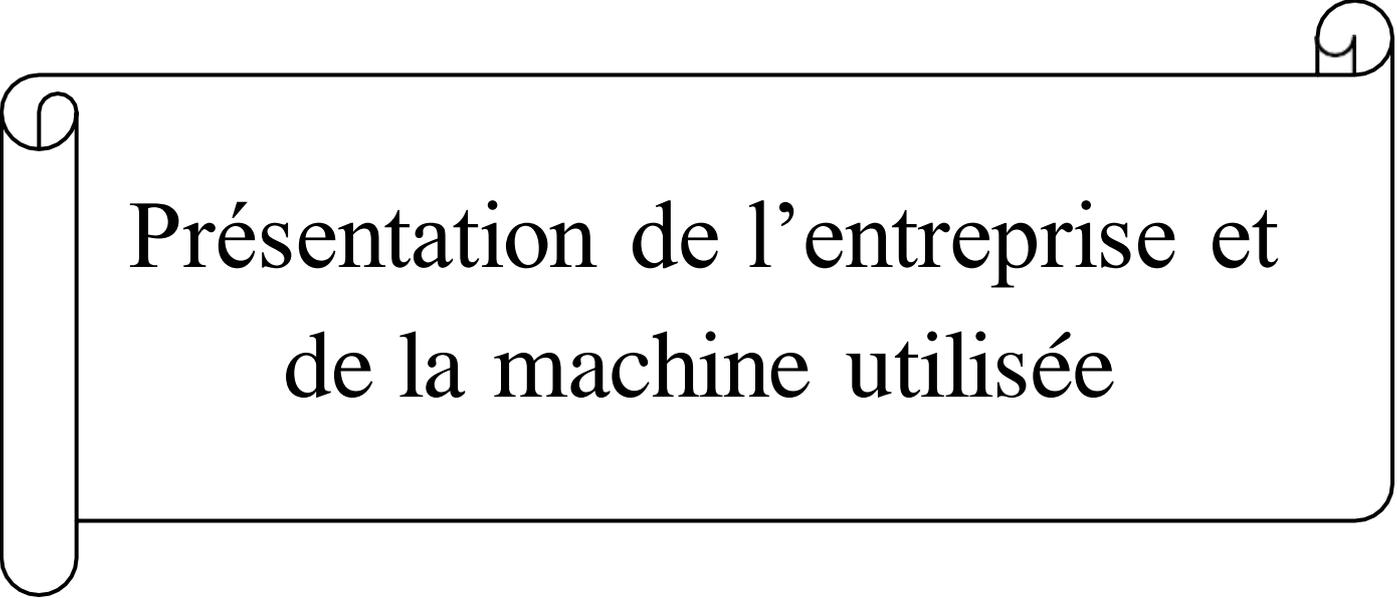
Le **premier chapitre** est consacré à la présentation de l'entreprise LABELLE, et la description du compresseur, son principe de fonctionnement, ses principaux éléments qui le constituent ainsi que les différents types existants dans l'industrie.

Le **deuxième chapitre** est consacré aux éléments de l'automatisation, l'automate S7-1200 a été présenté ainsi que le logiciel TIA PORTAL.

Le **troisième chapitre** est dédié à la programmation et à la supervision. Enfin,

on termine par une conclusion générale et les perspectives envisagées.

CHAPITRE I



Présentation de l'entreprise et
de la machine utilisée

I.1. Introduction

Le bon fonctionnement d'un système mécanique, électrique ou hydraulique requiert une source d'énergie, qui peut être fournie de différentes manières. Cependant, dans certains cas, l'air comprimé est la meilleure option pour alimenter ces systèmes en raison de sa rapidité d'exécution et de sa polyvalence.

Le présent chapitre a pour but de découvrir les différents types de compresseurs, les différentes parties du compresseur à étudier, ainsi que leurs avantages et inconvénients. Nous discuterons aussi son principe de fonctionnement.

I.2. Présentation de l'entreprise corps gras de Bejaia LABELLE

I.2.1. Description de l'entreprise

L'entreprise Corps Gras de Bejaia (CO.G.B) LABELLE est une société de droit algérien créée par acte notarié établi le 14/05/2006. LABELLE est un groupe agroalimentaire qui assure la commercialisation et la fabrication de produits alimentaires.

I.2.2. Historique de l'entreprise

Avant de commencer la présentation de l'entreprise des corps gras de Bejaia, il est intéressant de la situer dans l'histoire.

Début de 20ème siècle : Extraction de l'huile de grignon d'olive et fabrication de savon à base d'huile de grignon par la Société Industrielle de l'Afrique du Nord (SIAN) [1].

En 1940, elle fut l'installation d'une usine conçue pour le raffinage de l'huile de colza et de tournesol. Ce complexe s'est lancé dans la production du savon de ménage « MON SAVON » en 1953.

Cette unité de production a été nationalisée en 1974 afin de porter le nom de la SIAN, pour devenir la Société de Gestion et de Développement des Industries Alimentaires (SOGEDIA). Puis elle a été transformée en 1982 pour donner naissance à trois entreprises à savoir :

- Entreprise Nationale des Corps Gras (ENCG) ;
- Entreprise Nationale des Jus et des Conserves (ENAJUC) ;
- Entreprise Nationale du Sucre (ENASUCRE).

Une autre nouvelle unité de production est installée à la zone industrielle de Bejaia en 1988; il s'agit du complexe des corps gras de Bejaia qui est composé de deux unités de production et un seul siège social.

En Août 2006, l'Etat a cédé une grande partie de l'entreprise CO.GB au profit du groupe LABELLE pour finalement devenir «CO.G.B-LABELLE».

De nos jours, l'entreprise exerce son activité sous la direction du groupe CO.GB LABELLE qui dispose de différentes entreprises dans le secteur de l'agroalimentaire

I.2.3. Description et production de l'unité

L'entreprise LABELLE possède deux unités de production :

- *Unité de production N°7* : Où se trouve la direction générale ; elle est chargée du raffinage d'huile, de la production de savon de ménage et de savonnettes.
- *Unité de production N°8* : Chargée du raffinage d'huile, de la fabrication du savon de ménage, de savonnettes et de graisse végétale aromatisée.

L'unité de production (UP) est implantée dans la zone industrielle de la ville de Bejaia. Ils'étend sur une surface de 108800 m² dont 56500 m² couverte. L'unité est composée d'une raffinerie d'huile pour une production de 400 tonnes/jour, d'une savonnerie pour une production de 150 tonnes/jour, d'un conditionnement d'huile pour une production de 300 tonnes/jour en conditionnement de 5l, 100 tonnes/jour en conditionnement de 1l et d'une margarinerie pour une production de 80 tonnes/jour.

I.2.4. Localisation géographique

L'entreprise des corps gras de Bejaia a ouvert ses portes en 1987, sous tutelle du ministère des industries légères, elle est implantée dans la zone industrielle à IHADDADEN (Bejaia) et limitée par :

- Le complexe de costumes de Bejaia au Nord.
- L'entreprise de préfabrication légère et d'aluminium (ex : COPEMAD) au Sud.
- L'entreprise nationale des matériels de travaux publics à l'Ouest.

La figure I.1 illustre le schéma qui présente l'emplacement géographique de la CO.G.B LABELLE



Figure I.1 : Schéma montrant l'emplacement géographique de la CO.G.B LABELLE

I.2.5. Moyens de l'entreprise

CO.G.B LABELLE dispose des moyens matériels qui peuvent se résumer comme suit :

- Nature des moyens : Raffinerie, savonnerie, margarinerie et hydrogénation.
- Origine des moyens : Europe.
- Les différentes machines utilisées dans la production : Les souffleuses, les convoyeurs à air rafale, les remplisseuses, les bouchonneuses, le déviateur de bouteilles, la fardeleuse, l'encartonneuse, le palettiseur et les banderoleuses.
- Les investissements réalisés par l'entreprise se résument à la rénovation des ateliers, l'acquisition de nouveaux matériels de production, des machines de conditionnement et des moyens de transport.

I.2.6. Diagnostic de production de CO.G.B LABELLE

Le département de production ou bien l'assistant exploitation regroupe l'ensemble des ateliers qui ont pour mission de suivre le processus de transformation des matières en produits finis, en respectant les normes de production. Avec un effectif estimé à 142 agents qui travaillent 24h/24h, partagé en équipes de 8h/jour, ce département est composé de :

I.2.6.1. Service savonnerie

Son rôle est la fabrication du savon de ménage, savon de toilette ainsi que la glycérine pharmaceutique.

I.2.6.2. Service raffinage

Sa mission est la transformation de l'huile brute alimentaire destinée au conditionnement.

I.2.6.3. Service conditionnement des huiles

Ce service est partagé en deux ateliers :

- **Atelier plastique** : Son rôle est la fabrication de bouteilles en plastique.
- **Atelier conditionnement** : Son rôle est la mise en bouteilles de l'huile pour la commercialisation.

I.2.6.4. Service margarinerie

Sa mission est la production d'hydrogène, d'huile hydrogénée et de la margarine. La CO.G.B a pour objectif :

- ✓ D'améliorer la production qualitativement.
- ✓ De distribuer et de commercialiser tout produit relevant de son secteur d'activité.
- ✓ De développer l'industrie alimentaire.

La figure I.2 représente l'organigramme des départements de production de CO.G.B / UP.

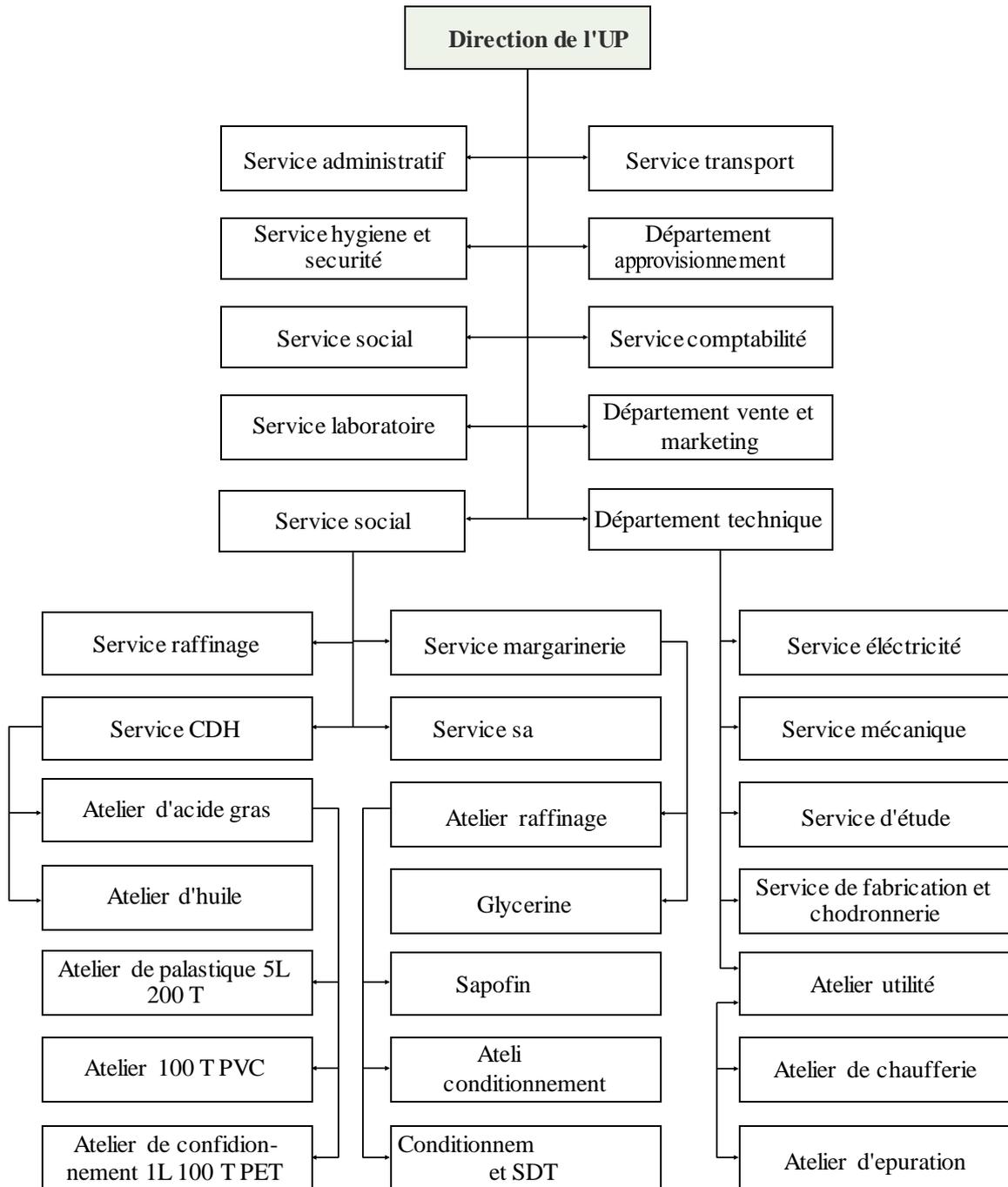


Figure I.2 : Organigramme de départements de production de CO.G.B / UP

I.2.7. L'unité de conditionnement d'huile de CO.G.B LABELLE

L'unité de conditionnement d'huile de CO.G.B LABELLE Constituée actuellement de cinq lignes de production : deux lignes pour la production des bouteilles Polyéthylène Haute Densité (PEHD) dont une pour les bouteilles de 5 litres et l'autre pour les bouteilles de 2 litres, et trois lignes pour les bouteilles Polyéthylène-Téréphtalate (PET) dont la première ligne pour la production des bouteilles de 5 litres, la deuxième ligne pour la production des bouteilles de 2 litres et la troisième pour la production des bouteilles d'un litre.

La matière utilisée pour la fabrication des bouteilles (5L, 2L, 1L) a une structure de tube. La matière utilisée pour la fabrication des bouteilles (5L, 2L) consiste à mélanger les déchets broyés (résidus de bouteilles) et le colorant dans un mélangeur /broyeur puis réchauffer le mélange à une température de 170°C pour donner le plastique fondu sous la forme d'une gaine.

I.3. Présentation de la machine

I.3.1. Air comprimé

L'air comprimé est une des plus anciennes formes d'énergie utilisées par l'humanité. Cette forme d'énergie est obtenue en prélevant de l'air dans l'atmosphère, en le dépolluant, en le comprimant, puis en le stockant dans des réservoirs pour être ensuite utilisé par les équipements. Cette transformation de l'air en énergie se déroule en deux étapes complémentaires et distinctes :

- La compression, qui fait appel à différentes technologies de compresseur.
- Le traitement, qui fait appel à différentes techniques.

I.3.1.1. Avantages de l'air comprimé

Les avantages de l'air comprimé sont nombreux :

- Disponibilité : L'air est partout présent en quantité illimitées.
- Transport : L'air comprimé peut être facilement transporté à l'aide de canalisations.
- Stockage : L'air peut être emmagasiné dans des cuves et prélevé à la demande.
- Antidéflagrant et ininflammable : aucun risque d'explosion.
- Propreté : Aucun risque de pollution, inutile de prévoir des canalisations de retour.
- Vitesse : L'air comprimé s'écoule très rapidement (vitesse habituelle des vérins : 2m/s).
- Tolérance à la surcharge : En cas de surcharge, les équipements pneumatiques fonctionnent jusqu'à l'arrêt sans risque de rupture ou détérioration [2].

I.3.1.2. Inconvénients de l'air comprimé

Bien que l'air comprimé présente de nombreux avantages, il y a également certains inconvénients à prendre en compte :

- Traitement : Obtenu à partir de l'air ambiant, l'air comprimé doit être purifié et séché pour éviter l'usure des équipements.
- Compressibilité : L'air étant, par nature, compressible, on ne peut obtenir facilement des vitesses de piston régulières.
- Pression limitée : De 6 à 8 bars. Au-delà, le coût serait beaucoup plus important.
- Bruit : Les échappements air sont bruyants et imposent l'installation de silencieux
- Coût : La production et le traitement restent d'un coup assez élevé [2].

I.3.2. Compresseur

Un compresseur est une machine conçue pour augmenter la pression d'un fluide qui le traverse. Cette augmentation de pression se produit lorsque le fluide se comprime, ce qui entraîne une réduction de son volume et une augmentation de sa pression.

Les compresseurs sont utilisés pour atteindre des niveaux de pression spécifiques dans le cadre de divers processus, tels que :

- Les réactions chimiques (pression convenable).
- Le stockage dans les cavités.
- La liquéfaction ou la séparation.
- Les cycles de réfrigération.

I.3.2.1. But de la compression

La compression peut être nécessaire d'un point de vue technique pour déplacer une quantité déterminée de gaz d'un système à une certaine pression vers un autre système à une pression plus élevée. Cela permet de répondre aux exigences spécifiques de transfert et de fournir le gaz comprimé à une pression adaptée à l'application souhaitée.

Cette opération a pour but de :

- Faire circuler un gaz dans un circuit fermé.
- Produire des conditions favorables (de pression) pour des réactions chimiques.
- Envoyer un gaz de la zone de production vers l'utilisateur.
- Obtenir de l'air comprimé pour la combustion.
- Récupérer du gaz.

I.3.2.2. Types des compresseurs

Il existe différents types de compresseurs utilisés dans les systèmes d'air comprimé. Chaque type présente des caractéristiques spécifiques et est adapté à des applications particulières. Voici quelques exemples des types de compresseurs couramment utilisés :

a) Compresseur à vis

Le compresseur à vis utilise des rotors en forme de vis qui tournent en sens inverse pour compresser l'air ou un autre gaz. L'air entre dans le compresseur par l'orifice d'admission et est comprimé en tournant avec les rotors. Ils sont largement utilisés dans l'industrie pour fournir de l'air comprimé ou d'autres gaz en raison de leur efficacité énergétique et de leurs fiabilités. La figure I.3 représente l'étage de compression d'un compresseur à vis.



Figure I.3 : Etage de compression d'un compresseur à vis

b) Compresseur à piston

Les compresseurs à piston sont les plus anciens et les plus répandus dans leur catégorie. Leur conception peut être simple ou double, et ils peuvent fonctionner avec ou sans lubrification à l'huile. Ils sont généralement utilisés pour la production de pressions élevées supérieures à 20 bars, en particulier pour les gaz spéciaux. La figure I.4 représente le compresseur à piston.



Figure I.4 : Compresseur à piston

c) Compresseurs à spirale

Les compresseurs en spirale, également connus sous le nom de compresseurs "scroll", se distinguent par leur forme allongée. Leur fonctionnement repose sur l'utilisation de deux spirales entrelacées qui agissent comme des palettes pour aspirer et comprimer les fluides. Généralement, l'une des spirales reste fixe tandis que l'autre se déplace de manière excentrique, sans rotation, afin de capturer des poches de fluide entre les spires. Ces compresseurs sont principalement utilisés dans le domaine de la climatisation en raison de leur faible niveau sonore. La figure I.5 représente le compresseur à spirale.



Figure I.5 : Compresseur à spirale

d) Compresseurs centrifuges

Les compresseurs centrifuges, également appelés turbocompresseurs, sont des machines à rotor qui tournent à une vitesse constante, permettant ainsi un écoulement continu du fluide. Selon le fluide réfrigérant utilisé, le compresseur centrifuge peut être équipé d'une seule roue ou de plusieurs roues. Dans le cas des compresseurs centrifuges à plusieurs roues, le diffuseur d'une roue est prolongé par un canal de retour qui redirige le fluide comprimé vers l'entrée de la roue suivante. La figure I.6 représente le compresseur centrifuge.



Figure I.6 : Compresseur centrifuge

I.3.2.3. Description d'un compresseur d'air SIAD Tempo2-950

Le compresseur Tempo2-950 est un équipement utilisé pour comprimer l'air ambiant afin de créer une source d'air comprimé pour diverses applications industrielles, commerciales et domestiques. Il est fabriqué par la société italienne ABAC, spécialisée dans la production de compresseurs d'air.

Le Tempo2-950 est un compresseur à piston, ce qui signifie qu'il utilise des pistons pour comprimer l'air. Il a une forte puissance et une capacité de réservoir de 24 litres, ce qui lui permet de fournir un débit d'air maximum de 195 litres par minute à une pression maximale de 10 bars.

Le compresseur est équipé d'un moteur électrique monophasé de 230 volts, ce qui le rend adapté à une utilisation domestique ou dans des petits ateliers. Il est également doté d'un système de refroidissement à air, qui évite la surchauffe pendant le processus de compression.

Le Tempo2-950 est compact et facile à transporter, grâce à ses roues et sa poignée de transport intégrées. Il est également équipé d'un régulateur de pression pour ajuster la pression de l'air selon les besoins de l'application.

La figure I.7 représente le compresseur SIAD tempo2-950



Figure I.7 : Compresseur SIAD tempo2-950

I.3.2.4. Différentes parties d'un compresseur d'air SIAD

Le compresseur Tempo2-950 est composé de plusieurs parties essentielles qui travaillent ensemble pour assurer son fonctionnement efficace. Voici les principales parties d'un compresseur comme le Tempo2-950 :

- **Moteur** : Le compresseur est équipé d'un moteur électrique monophasé de 230 volts. Le moteur fournit la puissance nécessaire pour entraîner les pistons et comprimer l'air.
- **Réservoir** : Également appelé cuve, est un réservoir d'air qui stocke l'air comprimé une fois qu'il a été comprimé par les pistons. Le réservoir permet de fournir un flux d'air continu et régulier, même lorsque le compresseur n'est pas en train de fonctionner.
- **Pompe à air** : Le compresseur Tempo2-950 comprend une pompe à air intégrée. Les pistons du compresseur fonctionnent comme des pompes à air, aspirant l'air ambiant dans le cylindre et le comprimant pour produire de l'air comprimé.
- **Pistons** : Le compresseur Tempo2-950 utilise des pistons pour comprimer l'air. Les pistons se déplacent de manière alternée à l'intérieur du cylindre, créant une compression de l'air ambiant.

- **Cylindre** : C'est la partie du compresseur où les pistons se déplacent. Il contient les chambres de compression où l'air est comprimé.
- **Soupapes d'admission et d'échappement** : Les soupapes d'admission permettent à l'air ambiant d'entrer dans le cylindre, tandis que les soupapes d'échappement permettent à l'air comprimé de sortir du cylindre et d'entrer dans le réservoir.
- **Régulateur de pression** : Le compresseur est équipé d'un régulateur de pression qui permet de contrôler et d'ajuster la pression de l'air comprimé selon les besoins de l'application.
- **Système de refroidissement** : Le Tempo2-950 est doté d'un système de refroidissement à air pour éviter la surchauffe pendant le processus de compression. Le refroidissement à air aide à maintenir une température de fonctionnement optimale.
- **Roues et poignée de transport** : Pour faciliter le déplacement et la portabilité du compresseur, il est équipé de roues et d'une poignée de transport intégré. Cela permet de transporter facilement le compresseur d'un endroit à un autre.

Ces composantes travaillent en synergie pour assurer le fonctionnement efficace du compresseur Tempo2-950, en aspirant, comprimant et stockant l'air comprimé pour une utilisation ultérieure.

I.3.2.5. Etude des organes électriques du compresseur

Les différents organes électriques :

➤ Sectionneur porte fusible

Le sectionneur est un dispositif électromécanique qui permet de séparer mécaniquement un circuit électrique de son alimentation, en assurant une distance de coupure adéquate sur le plan électrique. Son objectif principal peut être de garantir la sécurité des personnes travaillant sur la partie isolée du réseau électrique ou de mettre hors service une partie défectueuse du réseau tout en conservant les autres parties fonctionnelles.

Contrairement à un disjoncteur ou à un interrupteur, le sectionneur n'a pas la capacité de réaliser des coupures ou des fermetures électriques. Il est essentiel d'arrêter l'équipement aval afin d'éviter une interruption brusque de l'alimentation. Sinon, une ouverture non contrôlée pourrait provoquer des brûlures graves en raison de la formation d'un arc électrique lors de l'ouverture. La figure I.8 représente la symbolisation du sectionneur porte fusible.

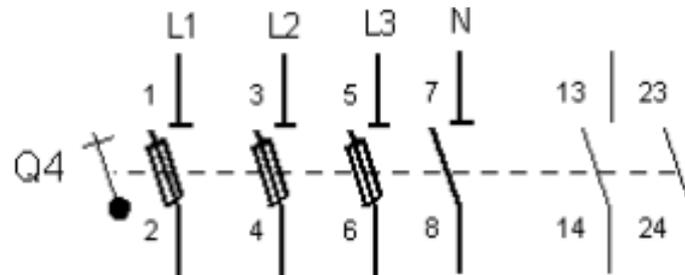


Figure I.8 : Symbolisation du sectionneur porte fusible

➤ Contacteur

Le contacteur est un dispositif de commande capable d'établir ou de couper le passage de l'énergie électrique, ce qui lui confère un pouvoir de coupure. Dans les systèmes de commande et d'automatisation, on le considère souvent comme un pré-actionneur car il se situe en amont des actionneurs dans la chaîne d'énergie.

Il peut être commandé à distance à l'aide de contacts actionnés manuellement, tels qu'un bouton-poussoir, ou automatiquement en étant asservi à une grandeur physique telle que la pression, la température, la vitesse, etc. Cela permet de contrôler et de réguler le flux d'énergie électrique en fonction des conditions requises dans le système.

La figure I.9 représente le symbole d'un contacteur

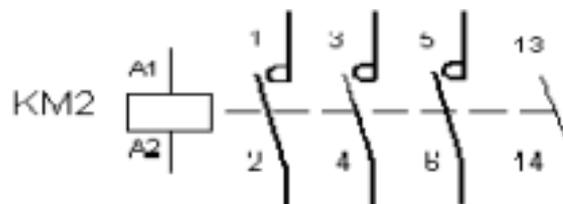


Figure I.9 : Symbole d'un contacteur

➤ Relais thermique

Le relais joue un rôle de protection en cas de surcharge prolongée, notamment pour les moteurs (en combinaison avec des fusibles). En cas de déclenchement, il est important de vérifier la cause avant de réarmer le relais. Il ne possède pas de contacts de puissance, mais il est généralement équipé de deux contacts de commande.

Il est composé d'un bilame métallique par phase, avec chaque lame ayant un coefficient de dilatation différent. Lorsque le courant dépasse la valeur calibrée du relais thermique, cela entraîne une élévation de température dans le circuit, ce qui déforme le bilame et ouvre ainsi le circuit de commande. Cette action protège le moteur ou le système contre une surcharge excessive qui pourrait endommager les composants.

La figure I.10 représente le symbole d'un relais thermique

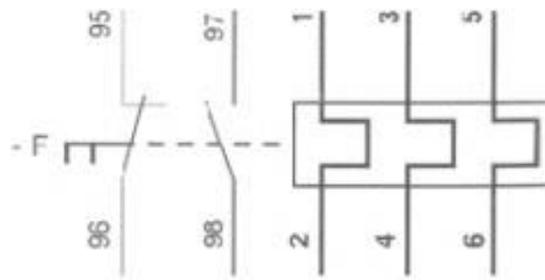


Figure I.10 : Symbole relais thermique

➤ Fusibles

Un coupe-circuit à fusible est un dispositif conçu pour interrompre le circuit dans lequel il est inséré en faisant fondre un élément fusible lorsque le courant qui le traverse atteint une valeur prédéterminée pendant une durée suffisante.

Le fusible se compose d'une lame fusible enfermée dans un boîtier fermé. Cette lame fusible fond si le courant qui la traverse dépasse sa capacité nominale. Le boîtier contient également du sable de silice pour faciliter la coupure en évitant le maintien du passage du courant à travers l'arc électrique.

Ainsi, lorsqu'un courant excessif ou anormal circule dans le circuit protégé, le fusible se fond, interrompant le flux de courant et protégeant les composants électriques ou électroniques contre les surcharges potentiellement dangereuses. La figure I.11 représente le symbole d'un fusible.

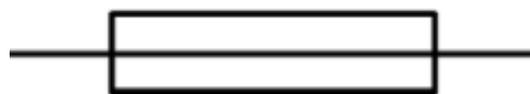


Figure I.11 : Symbole d'un fusible

➤ Bouton d'arrêt d'urgence

Le bouton d'arrêt d'urgence est un dispositif identifiable facilement grâce à sa forme ronde et sa couleur rouge sur le tableau de commande. On le surnomme également "bouton coup de poing" car il nécessite une certaine force pour être enfoncé, et en cas de situation d'urgence, il peut être activé en appuyant dessus brutalement, ce qui est à la portée de toute personne même non formée à l'utilisation de la machine.

Le symbole électrique représentant l'arrêt d'urgence est un champignon avec une tige. La tige est verticale et à sa partie inférieure, elle supporte un triangle rectangle dont la pointe est dirigée vers l'opposé du champignon et termine la tige. Les pointillés adjacents à la tige représentent les contacts de l'interrupteur.

Dans les documents techniques, le bouton d'arrêt d'urgence est généralement désigné par les acronymes ARU (Arrêt d'urgence) ou BAU (Bouton d'arrêt d'urgence). La figure I.12 représente le bouton d'arrêt d'urgence et son symbole.

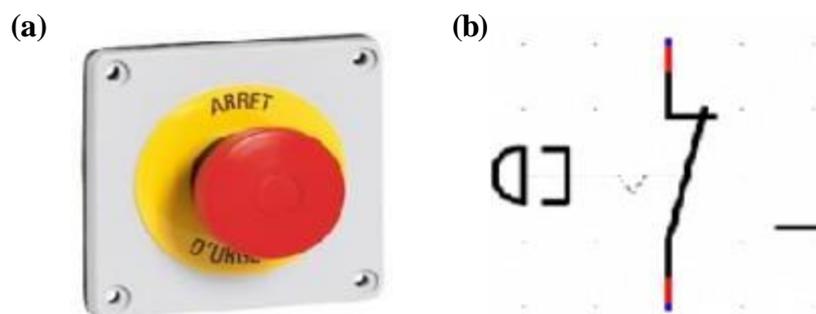


Figure I.12 : (a) Bouton d'arrêt d'urgence (b) Symbole de bouton d'arrêt d'urgence

I.3.2.6. Description de fonctionnement d'un compresseur SIAD

Lorsque le compresseur est en marche, le moteur électrique entraîne les pistons qui effectuent un mouvement alternatif à l'intérieur du cylindre. Ensuite les pistons se déplacent vers le bas, ils créent un espace dans le cylindre, ce qui permet à l'air ambiant d'être aspiré à travers les soupapes d'admission, les pistons remontent, l'air ambiant est comprimé dans le cylindre. Les pistons compriment l'air en réduisant l'espace disponible, ce qui augmente la pression de l'air. Pendant ce processus, les soupapes d'admission se ferment pour empêcher l'air de revenir en arrière. Une fois que l'air est comprimé, il est expulsé du cylindre par les soupapes d'échappement. L'air comprimé est ensuite dirigé vers le réservoir, également appelé cuve, où il est stocké à une pression élevée. Le réservoir agit comme un tampon pour fournir un flux d'air continu et régulier lorsque la demande en air comprimé fluctue.

I.3.2.7. Equipement entraîné par le compresseur

L'équipement entraîné par le compresseur étudié est la souffleuse, comme expliqué ci-dessous :

➤ Réception préforme

Les préformes sont chargées dans une trémie qui alimente la souffleuse à l'aide d'un convoyeur à préforme.

➤ Souffleuse

La souffleuse est une machine entraînée par le compresseur. Afin de réaliser le soufflage et de stériliser en même temps la préforme, un dispositif complet permet d'abord de chauffer la préforme sur toutes ses zones. La préforme passe directement dans le moule où elle sera étirée à basse pression par la tige d'étirage. Puis elle sera soufflée par l'air à haute pression (42 bars) pour obtenir à la fin une bouteille de forme complète.

I.3.2.8. Maintenance périodique

Les différents contrôles à effectuer sont les suivants :

a) Contrôles journaliers :

- Vérification des pressions et températures du circuit d'air.
- Vérification du bon fonctionnement des soupapes de sortie de l'air de condensation et nettoyage éventuel.
- Vérification de la pression et de la température du circuit d'eau.
- Vérification de la pression de l'air des instruments.
- Vérification du niveau d'huile.
- Vérification de la pression et de la température de l'huile.

b) Contrôles hebdomadaires :

- Purge de l'air du circuit de l'eau des culasses.
- Vérification des fuites au niveau des dispositifs de levage des lève-soupapes.
- Vérification de la tension des courroies de transmission.

c) Contrôles mensuels :

- Nettoyage du filtre à air.
- Contrôle du roulement extérieur du palier de vilebrequin prolongé.
- Lubrification des roulements du moteur électrique.

d) Contrôles semestriels :

- Vérification du bon fonctionnement des instruments.
- Lubrification des roulements du moteur électrique.
- Contrôle général de l'installation.

e) Operations annuelles :

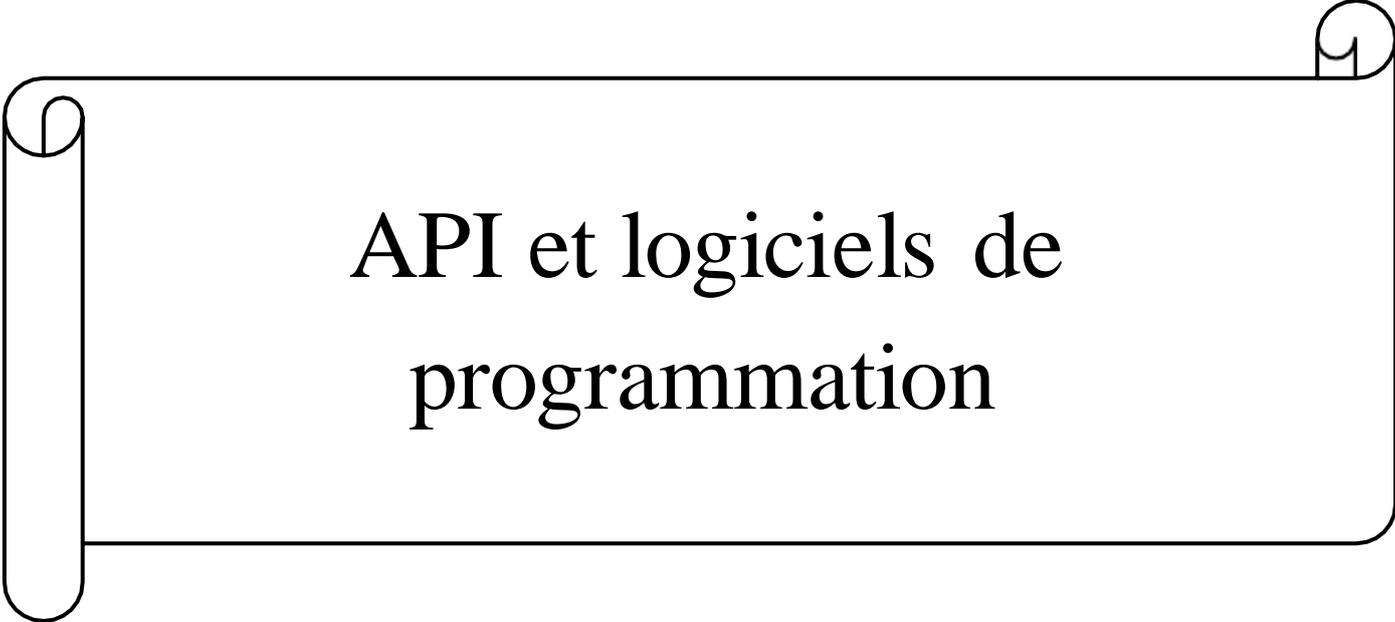
- Vidange de l'huile.
- Remplacement du filtre de refoulement d'huile.
- Nettoyage des surfaces internes du bâti.
- Lavage et détartrage du circuit de l'eau.

I.4. Conclusion

Dans ce chapitre, nous avons eu affaire à un compresseur, ses types, ses composants et son mode de fonctionnement. Nous avons aussi pu comprendre les différentes étapes de la production et de traitement de l'air comprimé grâce à sa description. Cela nous facilitera la tâche d'élaboration d'une étude fonctionnelles complète du cycle de fonctionnement du compresseur et de bien satisfaire les objectifs exigés par le cahier des charges, afin d'élaborer un programme de commande et de surveillance qui sera traitée dans le chapitre III.

Le prochain chapitre sera consacré à la présentation des différents éléments de l'automatisation et l'automate S7-1200.

CHAPITRE II



API et logiciels de programmation

II.1. Introduction

Les automates programmables sont des solutions de type robuste, ayant des entrées et des sorties physiques, utilisées dans de nombreux secteurs industriels. Ils peuvent avoir des applications différentes, notamment la surveillance et le contrôle des processus selon l'endroit où ils seront utilisés dans un environnement domestique ou professionnel.

Ce chapitre se consacre à une présentation générale des automates programmables, avec une attention particulière portée sur l'automate S7-1200 et le logiciel de programmation TIA PORTAL. Nous explorerons de manière détaillée ces deux éléments et leurs fonctionnalités.

II.2. Systèmes automatisés

II.2.1. Définition d'un système automatisé

Un système automatisé est un système dans lequel les tâches et les opérations sont effectuées de manière autonome, sans intervention humaine directe. Il repose sur l'utilisation de technologies telles que l'informatique, l'électronique, les capteurs et les actionneurs pour prendre des décisions et exécuter des actions de manière programmée.

Les systèmes automatisés peuvent être trouvés dans de nombreux domaines et industries, tels que la fabrication, la logistique, l'industrie automobile, l'aérospatiale, la domotique, les télécommunications, l'agriculture, la santé, etc. Ils sont conçus pour augmenter l'efficacité, la précision, la productivité et la sécurité des processus en éliminant ou en réduisant la dépendance à l'égard de l'intervention humaine.

II.2.2. Structure d'un système automatisé

Un système automatisé de production se compose fonctionnellement de deux parties principales : *partie opérative* et *partie commande*. Ces deux parties s'échangent les informations entre elles à l'aide des capteurs et près-actionneurs comme le montre la figure II.2.

II.2.2.1. Partie opérative

La partie opérative est la composante physique visible du système qui procède au traitement des matières premières afin d'élaborer le produit final. Elle regroupe :

- **Les actionneurs** : Ce sont des dispositifs qui permettent de produire un mouvement ou une force à partir d'un signal d'entrée (électrique, hydraulique, pneumatique, etc.). Ils sont souvent utilisés pour effectuer des actions physiques telles que l'ouverture et la fermeture des vannes ou le déplacement des vérins.
- **Les près-actionneurs** : Ce sont des dispositifs qui préparent et contrôlent le signal d'entrée de l'actionneur principal d'un système. Ils sont utilisés pour améliorer la précision, la vitesse ou la puissance de l'actionneur principal.
- **Les capteurs** : Ce sont des dispositifs qui permettent de mesurer et de transmettre à la partie commande des informations sur diverses grandeurs physiques telles que la position d'un objet, sa vitesse, la présence ou l'absence d'une pièce, la pression, etc.
- **Les effecteurs** : Ce sont des dispositifs finaux qui interagissent directement avec les matières premières dans le but de leur donner une valeur ajoutée, tels que des pompes, des têtes de soudure, etc [3].

II.2.2.2. Partie commande

Aussi connue sous le nom de "partie traitement des informations", la partie de commande est l'ensemble des composants qui permettent de traiter les informations nécessaires au bon fonctionnement de la partie opérative d'un système automatisé. Elle communique avec l'opérateur via un pupitre de commande [3].

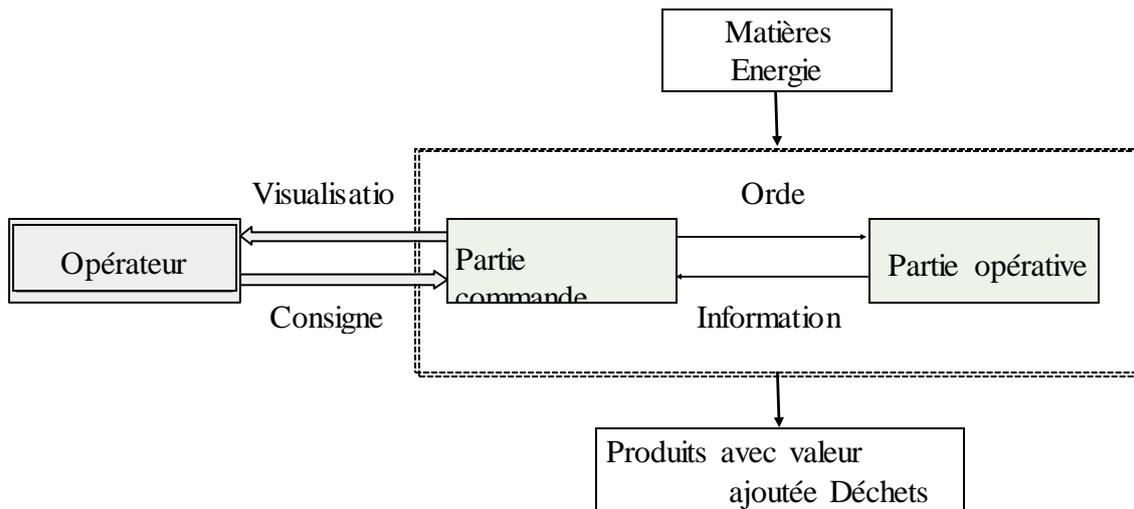


Figure II.1 : Structure d'un système automatisé

II.3. Objectifs de l'automatisation

L'automatisation permet d'apporter des éléments supplémentaires à la valeur ajoutée par le système, ces éléments sont exprimables en termes d'objectifs par :

- Accroître la productivité (rentabilité, compétitivité) du système.
- Améliorer la flexibilité de production.
- Améliorer la qualité du produit.
- Adapter à des contextes particuliers tels que les environnements hostiles pour l'homme (milieu toxique, dangereux, nucléaire...).
- Adapter à des tâches physiques ou intellectuelles pénibles pour l'homme (manipulation de lourdes charges, tâches répétitives parallélisées...).
- Augmenter la sécurité [4].

II.4. Généralité sur les automates programmables industriels

II.4.1. Définition d'un automate programmable industriel

Un automate programmable industriel (API) est une machine électronique programmable qui permet de piloter en temps réel des procédés logiques séquentiels dans un environnement industriel. Il est principalement utilisé par les automaticiens pour contrôler et commander un procédé industriel en s'assurant de l'adaptation nécessaire entre les composants de grande puissance et ceux de faible puissance côté commande. L'objectif principal de l'API est de maintenir le fonctionnement du système tel que prévu par l'utilisateur, en contrôlant les

sorties, en prenant des décisions et en agissant sur les entrées. Ce principe est connu sous le nom d'automatisme.

Un API est conçu pour automatiser la plupart des tâches de l'industrie, en assurant la commande des pré-actionneurs et des actionneurs à partir d'informations logiques, analogiques ou numériques.

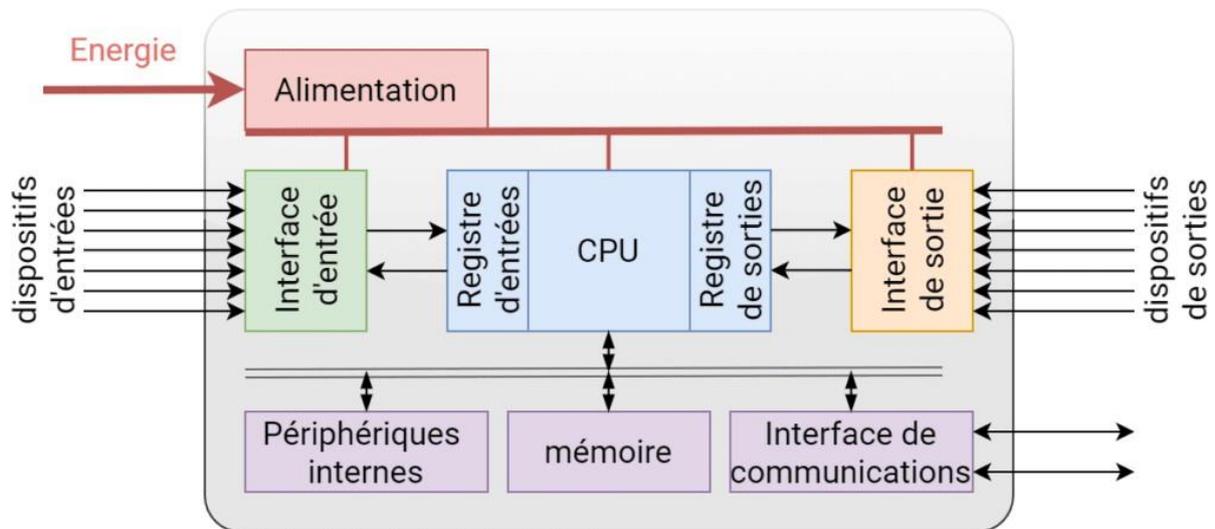


Figure II.2 : Architecture d'un API

II.4.2. Types d'automates

Les automates peuvent être de type compact ou modulaire.

- **Compact** : On distinguera les modules de programmation (LOGO de Siemens, ZELIO de Schneider, MILLENIUM de Crouzet ...) des micro-automates. Ils intègrent le processeur, l'alimentation, les entrées et les sorties. Selon les modèles et les fabricants, ils peuvent réaliser certaines fonctions supplémentaires (comptage rapide, E/S analogiques ...) et recevoir des extensions en nombre limité, généralement destinés à la commande de petits automatismes [5].
- **Modulaire** : Le processeur, l'alimentation et les interfaces d'entrées/sorties résident dans des unités séparées (modules) et sont fixées sur un ou plusieurs racks, Ces automates sont intégrés dans les automatismes complexes [5].

II.4.3. Architecture des automates

II.4.3.1. Structure interne

L'automate programmable reçoit des données qui décrivent l'état du système via une unité d'entrées et de sorties, qu'elles soient logiques, numériques ou analogiques. Il utilise ensuite le programme enregistré dans sa mémoire pour commander les pré-actionneurs en

conséquence.

Donc un API se compose de cinq grandes parties :

- **Une unité de traitement (un processeur CPU)** : Sa structure regroupe le processeur et la mémoire centrale, et elle est responsable de la gestion de l'interprétation et de l'exécution des instructions du programme.
- **Une mémoire** : Elle est spécifiquement conçue pour recevoir, gérer et stocker les informations provenant de différents secteurs du système. Ces secteurs incluent notamment :
 - Le terminal de programmation : C'est par le biais du terminal de programmation que le programme est introduit dans la mémoire.
 - Le processeur : Le processeur joue un rôle clé dans la gestion et l'exécution du programme stocké dans la mémoire.
- **Des modules d'entrées** : elle permet de recevoir les informations du système automatisé de production ou du pupitre et de mettre en forme (filtrage, ...) ce signal tout en l'isolant électriquement.
- **Des interfaces sorties** : Elles sont responsables de la commande des différents actionneurs et des éléments de signalisation du système automatisé programmable (S.A.P.). Elle assure également l'isolement électrique nécessaire pour garantir un fonctionnement sécurisé.
- **Une alimentation électrique** : Tous les automates actuels utilisent la tension 24 V
 - Une alimentation 240 VCA fournit un courant 24 VCC aux capteurs.
 - Les entrées sont également en 24 VCC Une mise à la terre doit également être prévue [3].

II.4.3.2. Structure externe

Les caractéristiques principales d'un API sont les suivantes :

- Compact ou modulaire.
- Tension d'alimentation.
- Taille mémoire.
- Nombre d'entrées/sorties.
- Modules complémentaires (analogique, communication...).
- Langage de programmation.

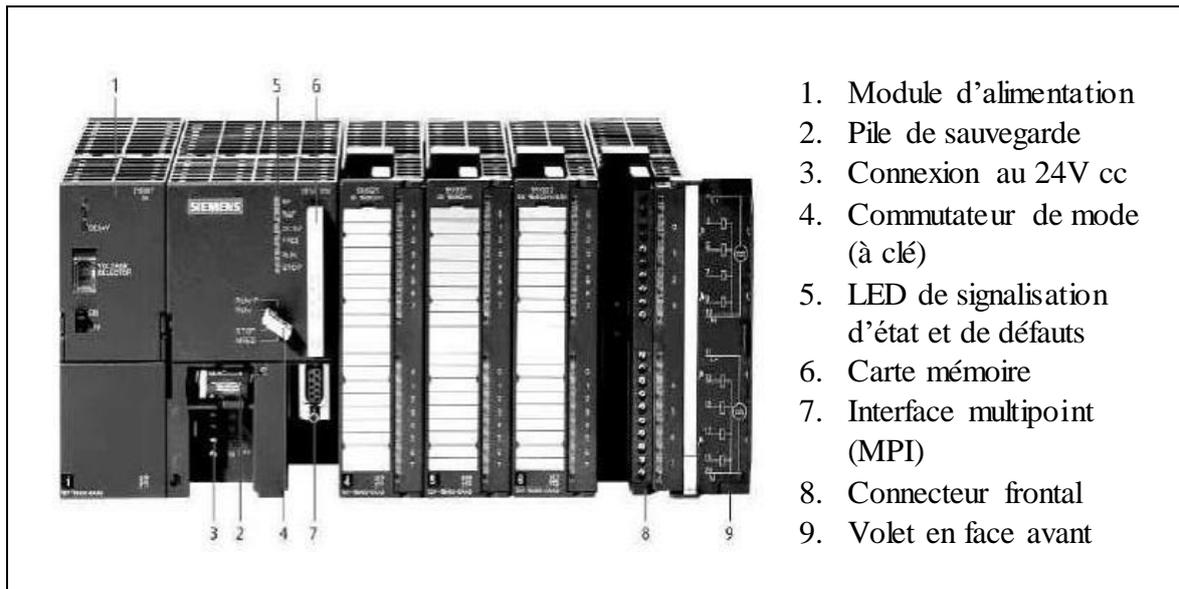


Figure II.3 : Automate modulaire (Siemens) [5]

II.4.4. Langages de programmation pour API

Selon la norme IEC 1131-3, il existe cinq langages de programmation pour les API qui sont :

- Langage Ladder (LD) : Il s'agit d'un langage graphique principalement utilisé pour la programmation d'équations booléennes (true/false) en utilisant des symboles de relais.
- Liste d'instructions (IL) : C'est un langage textuel de bas niveau. Ce langage se compose d'une seule instruction par ligne et peut être considéré comme étant similaire au langage assembleur.
- Diagramme de Blocs Fonctionnels (FBD) : Ce langage permet une programmation graphique en utilisant des blocs représentant des variables, des opérateurs ou des fonctions. Ce langage est capable de manipuler tous les types de variables.
- Texte Structuré (ST) : Il s'agit d'est un langage textuel de haut niveau. Ce langage permet la programmation d'algorithmes de tout type et de toute complexité.
- Grafset Séquentiel (SFC) : Ce langage de haut niveau est dérivé du langage GRAFCET. Il permet une programmation facile de tous les processus séquentiels.

II.4.5. Caractéristiques et fonctionnalités des logiciels de programmation

Les logiciels jouent un rôle essentiel dans notre société moderne en exécutant des programmes et des applications sur des ordinateurs et des appareils mobiles. Ces programmes sont composés d'un ensemble d'instructions qui permettent au matériel informatique d'accomplir des tâches spécifiques. Les logiciels sont omniprésents et utilisés dans presque tous les aspects de notre vie quotidienne, que ce soit pour la communication, le divertissement, l'éducation, les soins de santé ou d'autres domaines. Leur importance est indéniable dans le fonctionnement et le progrès de notre monde connecté.

Les logiciels présentent plusieurs caractéristiques essentielles qui les différencient du matériel informatique. L'une des caractéristiques les plus remarquables des logiciels est qu'ils sont intangibles. Contrairement au matériel, que l'on peut toucher et voir, le logiciel est un ensemble de codes et d'instructions stockés électroniquement. Cette intangibilité rend les logiciels plus flexibles et plus faciles à mettre à jour et à modifier que le matériel.

- La création d'un logiciel est un processus complexe qui implique plusieurs étapes. La première étape consiste à identifier le problème spécifique que le logiciel vise à résoudre. Une fois que le problème est clairement défini, l'équipe de développement se lance dans l'élaboration d'un plan de conception et de réalisation du logiciel. Ce plan comprendra des informations détaillées sur les caractéristiques et les fonctionnalités du programme, ainsi que sur le choix du langage de programmation et des outils qui seront utilisés pour sa mise en œuvre.
- Bibliothèques de fonctions et de blocs fonctionnels : Les logiciels API comprennent des bibliothèques de fonctions et de blocs fonctionnels prédéfinis. Ces bibliothèques contiennent des modules de code réutilisables pour des tâches courantes telles que la gestion des entrées/sorties, le traitement des signaux, les opérations mathématiques, la temporisation, etc. Elles permettent aux programmeurs de gagner du temps en utilisant des fonctions déjà développées.
- Simulation et débogage : Les logiciels API offrent des outils de simulation et de débogage puissants. Ils permettent aux programmeurs de tester et de vérifier leurs programmes avant de les déployer sur les automates réels. Les fonctionnalités de simulation permettent de simuler les entrées/sorties, d'observer l'exécution du programme et de détecter d'éventuelles erreurs ou dysfonctionnements. Les outils de débogage facilitent l'identification et la correction des erreurs dans le code.
- Communication et connectivité : Les logiciels API offrent des fonctionnalités de communication et de connectivité pour interagir avec d'autres systèmes. Ils prennent en

charge différents protocoles de communication telle que Modbus, Profibus, Ethernet/IP, etc., permettant aux API de communiquer avec d'autres automates, systèmes de supervision, bases de données ou équipements externes [8].

II.4.6. Rôle de l'API

- Permettre la connexion des logiciels, applications, et services à leurs données.
- Permettre la communication entre deux applications distinctes.
- L'échange des fonctionnalités et données à l'intérieur ou l'extérieur des entreprises.

II.4.7. Les avantages et inconvénients de l'API

Les API offrent de nombreux avantages, on peut les résumer dans les points suivants :

- La capacité de production accélérée.
- La simplicité de mise en œuvre.
- La souplesse d'utilisation.
- L'intégration dans un système de production.

Les API présentent également certains inconvénients, notamment :

- La complexité de configuration et de programmation, qui requiert souvent des compétences techniques spécialisées.
- Les risques de pannes ou de dysfonctionnements, qui peuvent entraîner des coûts importants de temps d'arrêt de la production.
- La maintenance coûteuse des API, qui nécessite souvent des pièces de rechange spécifiques.
- La limitation de leur flexibilité et de leur capacité de stockage et de traitement des données, qui peuvent limiter leur utilisation pour des applications nécessitant des calculs complexes ou la manipulation de grandes quantités de données.

II.4.8. Critère de choix d'un automate

Lorsqu'on choisit un automate programmable, il est courant de se baser sur plusieurs critères qu'on peut citer :

- **Le nombre d'entrées/sorties** : Nombre de cartes nécessaires peut rapidement augmenter si le nombre d'entrées/sorties requis est important. Ce facteur est donc à prendre en compte dès que la quantité d'entrées/sorties nécessaires devient élevée.
- **Type de processeur** : Les spécifications du processeur, notamment sa capacité de stockage, sa vitesse de traitement et ses fonctionnalités supplémentaires.
- **Fonctions ou modules spéciaux** : Commande d'axe ou pesage, afin d'alléger la charge du processeur et de répondre aux exigences de résolution et d'autres caractéristiques nécessaires.
- **Fonctions de communication** : l'automate doit être en mesure de communiquer avec les autres systèmes de commande tels que les API et la supervision, tout en offrant des possibilités de communication conformes aux normes normalisées.

II.4.9. Domaine d'applications

De nos jours, il est impossible de concevoir un système de production sans intégrer les différentes technologies et composants qui constituent les systèmes automatisés. Ces technologies sont omniprésentes dans de nombreux domaines, tels que l'automobile, l'aviation, l'industrie, la médecine et les transports.

II.5. Simatic S7-1200

II.5.1. Description

Le SIMATIC S7-1200 est un mini-contrôleur compact et modulaire, conçu pour des performances précises dans les petites applications d'automatisation. Grâce à sa configuration flexible et à son large éventail d'instructions, il offre une puissance et une souplesse qui permettent de commander une variété d'appareils. C'est donc une solution idéale pour des applications très diverses.

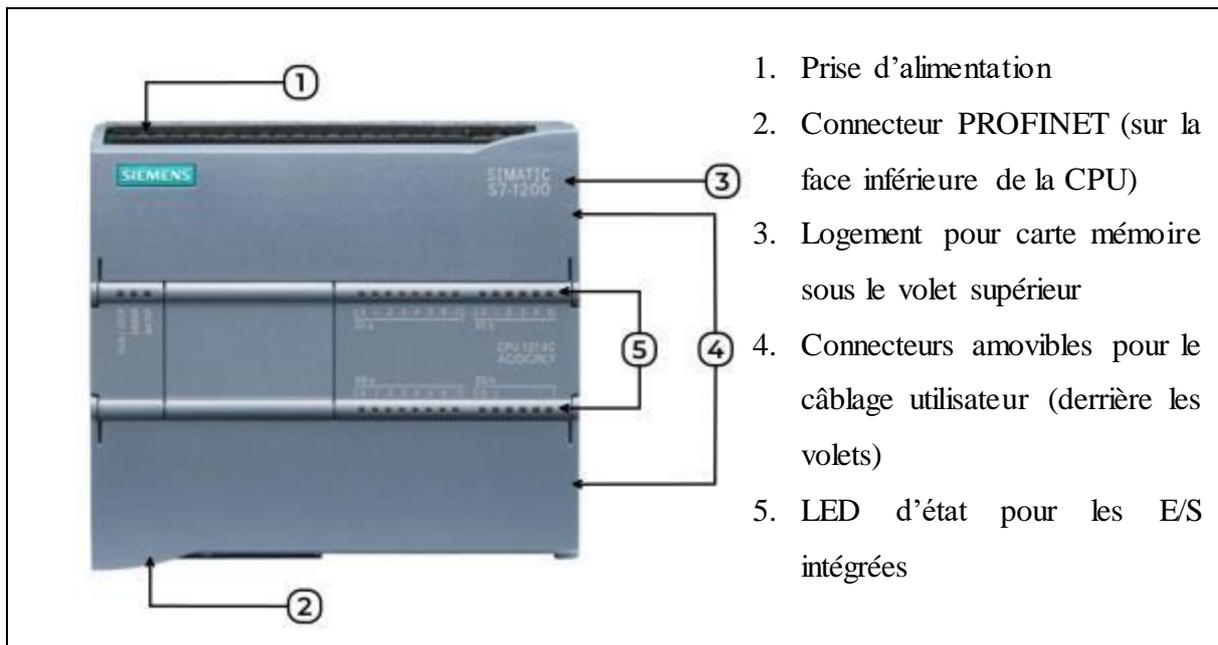


Figure II.4 : Simatic S7-1200 [7]

Ce contrôleur est composé d'une CPU équipée d'un microprocesseur, une alimentation intégrée, des circuits d'entrée et de sortie, un module PROFINET intégré, des entrées/sorties rapides de commande de mouvement, ainsi que des entrées analogiques intégrées dans un boîtier compact en vue de créer un contrôleur puissant.

La CPU fournit un port PROFINET permettant de communiquer par le biais d'un réseau PROFINET. Si besoin des modules supplémentaires sont disponibles pour communiquer via les réseaux PROFIBUS, GPRS, RS485 ou RS232 [8].

II.5.2. Choix de l'automate S7-1200

Nous avons choisi l'automate en fonction des critères suivants :

- Taille compacte : L'automate S7-1200 est de petite taille, ce qui permet de l'intégrer facilement dans des espaces restreints.
- Facilité d'utilisation : L'automate offre une interface conviviale et intuitive pour faciliter la programmation et la configuration.
- Disponibilité des pièces de rechange : Les pièces de rechange pour l'automate sont facilement disponibles, ce qui facilitera la maintenance et la réparation en cas de besoin.
- Large éventail d'applications : Il convient à de nombreuses applications industrielles, telles que le contrôle de machines, la surveillance de processus, l'automatisation de lignes de production, etc.
- Évolutivité : L'automate S7-1200 est évolutif, ce qui signifie qu'il peut être étendu avec des modules d'extension pour répondre aux besoins spécifiques d'une application donnée.

II.5.3. Modules de communication

On peut ajouter à la CPU SIMATIC S7-1200 jusqu'à trois modules de communication maximum.



Figure II.5 : Simatic S7-1200 et ses modules de communication [9]

II.5.4. Mémoire

Jusqu'à 50 Ko de RAM est présentée avec une part ajustable entre le programme utilisateur et les données utilisateur avec un maximum de 2 Mo de mémoire de chargement intégrée et 2 Ko de mémoire de données rémanente. Il y a la possibilité d'ajouter une carte mémoire SIMATIC, elle permet de transférer facilement des programmes sur plusieurs CPU et d'enregistrer des différents fichiers elle est aussi utilisée pour actualiser le microprogramme de l'API.

II.5.5. Les modules d'entrées/sorties

Il est possible d'avoir jusqu'à huit modules d'entrées/sorties et d'ajouter des entrées/sorties TOR et analogiques supplémentaires.

II.5.5.1. Choix des modules d'entrées/sorties

Le choix des modules d'entrées/sorties est basé sur les critères suivants :

- Type et valeur de la tension d'entrée ou de sortie.
- Nombre de voies.
- Type d'entrée ou de sortie (sonde, thermocouple, etc.) [10].

II.5.6. Régulation

Ces sorties peuvent servir à réguler la vitesse d'un moteur en utilisant la Modulation de Largeur d'Impulsion (MLI) ou à positionner une vanne. Il est possible de réaliser des applications process simples avec une régulation en circuit fermé en utilisant des boucles de régulation PID dotées d'une fonctionnalité d'auto-ajustement.

II.5.7. Communication industrielle

II.5.7.1. Interface PROFINET intégrée

Le SIMATIC S7-1200 est équipé d'une interface PROFINET intégrée qui garantit une communication transparente et optimale avec le système d'ingénierie intégré pour la programmation, avec des pupitres IHM pour la visualisation, avec des contrôleurs supplémentaires pour la communication API-API et avec des appareils d'autres constructeurs pour des possibilités d'intégration avancées.

L'interface de communication SIMATIC S7-1200 se compose d'un connecteur RJ45 insensible au bruit avec fonction de croisement automatique prenant en charge jusqu'à 16 connexions Ethernet et fournissant des débits de transmission de données jusqu'à 10/100 M bit/s [6].

II.5.7.2. Mise en réseau simple

Le module de commutation compact CSM 1277 a été conçu pour réduire le câblage et offrir une grande flexibilité de réseau. Il permet de configurer facilement des réseaux homogènes ou mixtes composés de topologies linéaires, arborescentes ou en étoile. Ce commutateur non géré à quatre ports permet de connecter jusqu'à trois appareils supplémentaires au SIMATIC S7-1200.

II.5.7.3. Communication avec d'autres automates et appareils IHM

Pour établir une communication avec d'autres automates SIMATIC et terminaux IHM, le SIMATIC S7-1200 peut se connecter à plusieurs appareils utilisant le protocole de communication Profinet (TCP/IP). Ce protocole est reconnu pour sa fiabilité et son efficacité, garantissant ainsi une communication stable et efficace entre les différents équipements.

II.5.8. Choix de la CPU

Les CPU du système SIMATIC S7-1200 se déclinent en trois classes de performances : CPU1211 C, CPU1212 C et CPU1214 C, chacune d'elles pouvant être étendue en fonction des besoins de la station. Sur chaque CPU, il est possible de greffer une platine d'extension pour ajouter des entrées/sorties, TOR ou analogiques supplémentaires sans modification de l'encombrement de l'automate.

Des modules d'entrées/sorties supplémentaires peuvent être ajoutés du côté droit de la CPU pour étendre la capacité d'entrées/sorties, TOR ou analogiques [8].



Figure II.6 : La CPU 1214C DC/DC/DC [10]

II.6. Logiciel de programmation STEP7

La programmation de L'API se fait à l'aide d'un PC avec le logiciel STEP 7 (La plateforme Siemens TIA Portal (Totally Integrated Automation)) qui est le logiciel de base de la configuration de systèmes d'automatisation SIMATIC. Il offre de différentes fonctions pour l'automatisation d'une installation :

- Configuration et paramétrage du matériel
- Paramétrage de la communication
- Création des programmes
- Test, mise en service et maintenance
- Documentation, archivage
- La gestion des mnémoniques
- Fonctions de diagnostic et d'exploitation [11].

II.6.1 Création d'un projet sous STEP7

Pour la création d'un projet sous STEP7 on a deux solutions possibles :

- **Alternative 1** : commencer par la configuration matérielle.
- **Alternative 2** : commencer par la création du programme.

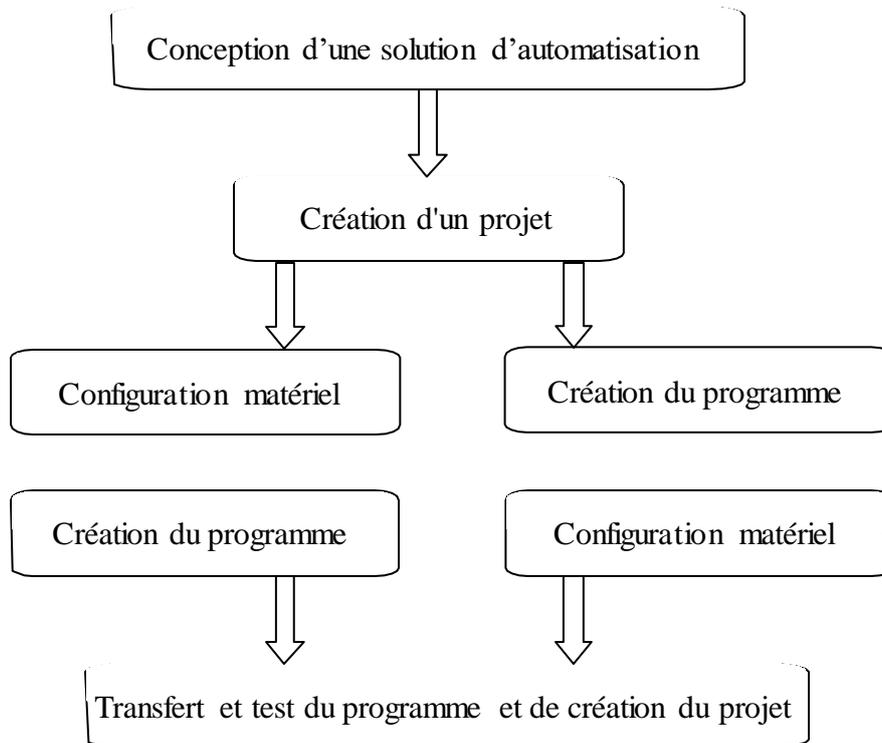


Figure II.7 : Organigramme de création du projet [12]

II.6.2 Différents types de blocs utilisateurs sur STEP7

- **Blocs d'organisation (OB)** : Ils servent d'interface entre le système d'exploitation et le programme utilisateur.
- **Fonctions (FC)** : Ils sont responsables de fonctionnalités spécifiques du programme et peuvent être configurables en fonction des paramètres.
- **Blocs fonctionnels (FB)** : Du point de vue du programme, les blocs fonctionnels sont similaires aux fonctions FB et SFB, mais ils possèdent également des zones mémoires spécifiques sous la forme de blocs de données d'instance. Les blocs fonctionnels sont utilisés pour programmer des fonctionnalités récurrentes encore plus complexes, telles que l'accomplissement de tâches de régulation.

- **Blocs de données (DB)** : Ils représentent des zones de données dans le programme. Les DB utilisateur fournissent de l'espace mémoire pour les variables de différents types de données. Il y a deux types de blocs de données :
- **Bloc de données globales** : Ce bloc est accessible par tous les blocs (FB, FC et OB), et il peut être créé indépendamment des autres blocs. Aucune donnée locale statique ne peut être déclarée dans ce bloc. Les variables peuvent être insérées, supervisées et modifiées dans les blocs de données globales. Il est également possible de modifier les valeurs initiales et les valeurs en cours dans ce bloc. La structure de ce bloc est librement définie.
- **Blocs de données d'instance** : Ce bloc est associé à un FB et ne peut être créé que si le FB associé est présent dans le programme. Les données locales et statiques peuvent être déclarées dans le FB, et ces données ne sont pas perdues après la fin de l'exécution du bloc. Elles sont conservées sur plusieurs cycles jusqu'à ce qu'elles soient à nouveau écrites.

II.7. Supervision

II.7.1. Définition

Il s'agit d'une méthode industrielle utilisée pour surveiller et contrôler informatiquement les processus de fabrication automatisés.

La supervision implique la collecte de données (telles que des mesures, des alarmes et des retours d'état de fonctionnement) ainsi que le contrôle des paramètres des processus, généralement effectués par des automates programmables [12].

II.7.2. Avantages de la supervision

Un système de supervision assiste l'opérateur dans la gestion du processus en fournissant des résultats expliqués et interprétés. Ses principaux avantages sont les suivants :

- La surveillance à distance du processus.
- La détection des défauts.
- Le diagnostic et la gestion des alarmes.
- Le traitement des données.

II.7.3. Logiciels de supervision

Les logiciels de supervision offrent la possibilité de créer des interfaces homme-machine pour les panels de supervision ou les écrans. De nombreux fabricants de cartes IHM

proposent leurs propres logiciels de supervision. Certains éditeurs, tels qu'Arc Informatique ou Wonderware, se spécialisent dans le développement de ces logiciels. Voici une liste de programmes de supervision largement utilisés dans l'industrie.

- WinCC de Siemens.
- Vijeo Designer de Schneider Electric.
- FactoryTalk de Rockwell Automation.
- PCVUE d'Arc Informatique.
- InTouch de Wonderware [13].

II.8. Présentation du logiciel de supervision WINCC flexible

Le logiciel WinCC de Siemens est utilisé comme système d'acquisition de données et de surveillance en temps réel, ainsi que comme interface homme-machine dans le domaine industriel. Il offre un guidage intuitif aux opérateurs en utilisant une gamme variée de produits et de systèmes standardisés, flexibles et évolutifs, notamment pour la surveillance des processus industriels [14].

II.8.1. Interface homme-machine

L'opérateur utilise une Interface Homme-Machine (IHM) pour exercer le contrôle et la supervision du processus.

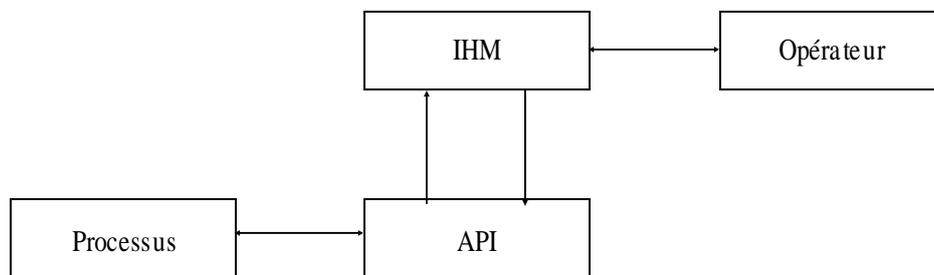


Figure II.8 : Interface HMI dans un processus automatisé

Un système d'interface homme-machine (IHM) permet la communication entre un opérateur humain et un processus géré par une machine ou une installation. Le contrôle effectif du processus est assuré par un système d'automatisation. Ainsi, il existe une interface entre l'opérateur et WinCC flexible, qui est située sur le pupitre opérateur, et une autre interface entre WinCC flexible et le système d'automatisation.

- Représentation du processus : se fait à travers le pupitre opérateur, qui affiche les états en temps réel et se met à jour lorsque le processus évolue.
- Commande du processus : est effectuée par l'opérateur via une interface utilisateur graphique. Il peut par exemple définir une valeur de consigne pour un automate ou démarrer un moteur.
- En ce qui concerne les alarmes, lorsque des états critiques se produisent dans le processus, une alarme est immédiatement déclenchée. Par exemple, lorsque certaines valeurs limites sont dépassées.

Le SIMATIC WinCC flexible est un logiciel IHM qui permet de mettre en place des concepts d'automatisation évolutifs au niveau machine grâce à des méthodes d'ingénierie simples et efficaces. WinCC flexible offre les avantages suivants :

- Simplicité : Le logiciel est conçu de manière à être facile à utiliser et à configurer. Il propose une interface utilisateur conviviale qui simplifie la gestion du processus.
- Ouverture : WinCC flexible est compatible avec divers systèmes d'automatisation, ce qui lui confère une grande flexibilité en termes de connectivité. Il peut se connecter à différentes marques d'automates programmables, de contrôleurs et d'autres périphériques, favorisant ainsi l'intégration avec les équipements existants et l'échange de données.
- Flexibilité : WinCC flexible permet de personnaliser les interfaces utilisateur en fonction des besoins spécifiques de l'application. Les opérateurs peuvent créer des écrans de visualisation, des panneaux de commande et des tableaux de bord adaptés à leurs préférences et aux informations requises. De plus, il est possible de configurer des alarmes pour une détection rapide des situations critiques [15].

II.8.2. Principales fonctionnalités fournies par WinCC flexible

Les Principales fonctionnalités fournies par WinCC flexible sont les suivantes :

- WinCC flexible RT : Il s'agit d'un système de gestion centralisé du projet qui permet un accès rapide à toutes les données du projet et aux paramètres centraux.
- Communication : Il offre la possibilité de déclarer et d'adresser des variables, qu'elles soient internes à WinCC ou externes.
- Visualisation : Il propose un système graphique permettant une visualisation configurable de manière flexible, ainsi que le pilotage à l'aide d'objets entièrement graphiques.

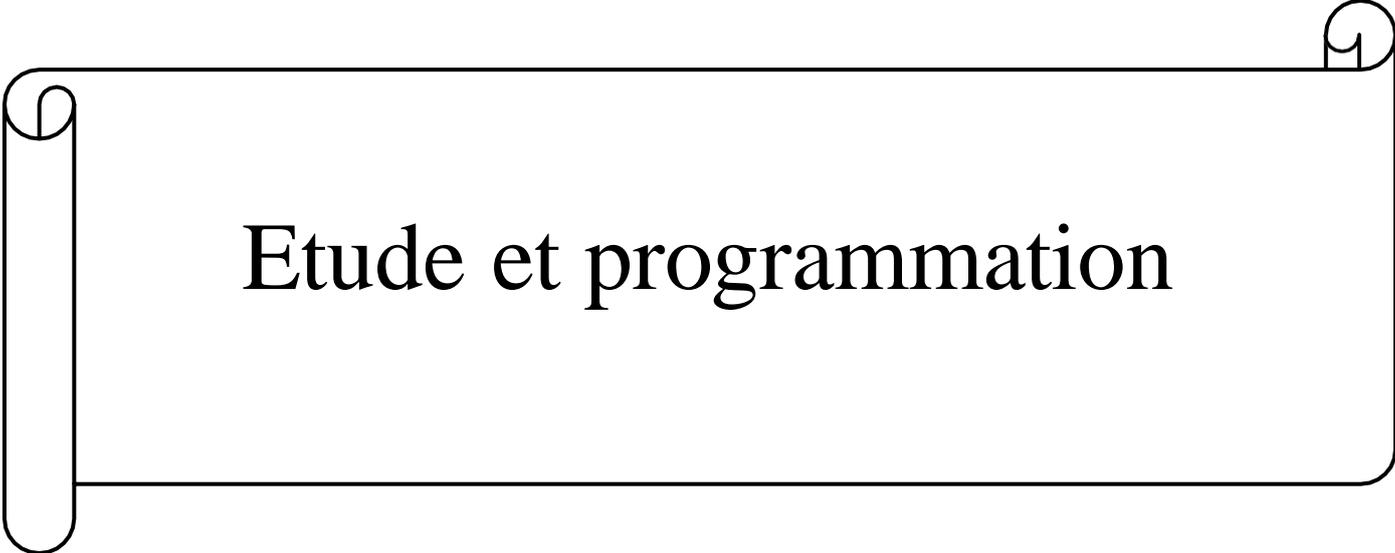
- Historique : Il permet la compression et l'archivage des mesures et des alarmes au fil du temps.
- Gestion des alarmes : Il intègre un système de messagerie pour la saisie et la visualisation des alarmes analogiques et des alarmes TOR (Tout Ou Rien).
- Paramétrages : Il offre un système permettant de configurer les paramètres des alarmes.
- Journaux : Il propose des systèmes de journalisation pour la documentation des processus en temps réel, que ce soit à déclenchement temporel ou événementiel, sous forme de rapports utilisateur ou de documentation de projet avec une mise en page personnalisable.

II.9. Conclusion

Au cours de ce chapitre, nous avons exploré divers éléments de l'Automate Programmable Industriel. Nous avons commencé par analyser sa structure modulaire et son architecture, en soulignant les aspects internes et externes. Ensuite, nous avons approfondi notre compréhension du logiciel TIA PORTAL V14, un outil essentiel qui permet la programmation et la supervision avancées des API.

Le prochain chapitre sera consacré à l'application de langage de programmation Ladder dans le contexte spécifique du compresseur SIAD Tempo2-950.

CHAPITRE III



Etude et programmation

III.1. Introduction

Dans le domaine de l'automatisation industrielle, la programmation Ladder joue un rôle essentiel dans le contrôle et la supervision des systèmes complexes.

Dans ce chapitre, nous explorerons l'application de ce langage de programmation dans le contexte spécifique du compresseur SIAD Tempo2-950. Ce compresseur, largement utilisé dans l'industrie, permet de comprimer des fluides afin de répondre à divers besoins de production.

III.2. Fonctionnement du compresseur SIAD tempo2_950

Lorsque le bouton START au tableau est enclenché, le compresseur va se met en marche, et une sirène d'alarme générique retentit pendant 5 secondes pour signaler son démarrage.

Lorsque le processus de démarrage est initié, chaque élément clé s'active automatiquement pour assurer un fonctionnement harmonieux. Tout d'abord, la pompe d'huile démarre automatiquement, garantissant une lubrification adéquate du moteur, Ensuite, la tour se met en marche, favorisant une circulation efficace de l'air et une ventilation optimale. Parallèlement, la pompe d'eau se met également en route, et enfin, le compresseur se démarre automatiquement.

Le compresseur démarre d'abord à vide, en appuyant sur le bouton AUTO-START-100 % le compresseur va se mettre à charge, et une indication de cette sélection s'affiche au panneau.

Le compresseur est considéré comme chargé lorsque les trois électrovannes sont activées. Les trois électrovannes sont montées sur le circuit de l'air et sont toutes excitées.

Lorsque le compresseur atteint la valeur la plus haute 36bars, il se remet immédiatement à vide.

A l'obtention de la plus basse valeur de consigne, le compresseur redémarre automatiquement. Au cas où la valeur la plus basse ne serait pas atteinte, le compresseur s'arrête automatiquement.

En cliquant sur le bouton AUTO-START-0%, le compresseur va se mettre à vide.

Le compresseur est à vide lorsque les trois électrovannes sont désactivées (électrovannes désexcités).

En appuyant sur le bouton STOP ou le bouton d'URGENCE, la machine va s'arrêter.

L'air passe par trois étages de compression :

- Premier étage de compression : Dans cette première étape, l'air est comprimé par un piston se déplace vers le haut dans le cylindre. Ce réduit le volume de l'air, augmente ainsi sa pression à 2 bars. L'air comprimé est ensuite dirigé vers un intercooler pour le refroidissement avant de passer au deuxième étage.
- Deuxième étage de compression : L'air comprimé provenant du premier étage est dirigé vers le deuxième étage de compression. Une fois de plus, le piston comprime l'air à 12 bars. Après la compression, l'air est renvoyé vers l'intercooler pour un refroidissement supplémentaire avant de passer au troisième et dernier étage.
- Troisième étage de compression : Dans cette dernière étape, l'air comprimé provenant du deuxième étage est comprimé une fois de plus. Le piston se déplace vers le haut, augmente encore sa pression jusqu'à 40 bars. L'air comprimé est ensuite acheminé vers un défi final pour éliminer la chaleur générée par la compression.

Une fois comprimé et refroidi, l'air comprimé peut être stocké dans un réservoir d'air comprimé pour une utilisation ultérieure.

III.3. Cahier des charges

Le cahier de charge du compresseur est composé de plusieurs étapes qui sont résumées dans l'ordre chronologique suivant :

a) Conditions de démarrage :

- Bouton d'arrêt d'urgence
- Bouton marche/arrêt
- Température de l'eau
- Température de l'huile
- Pressostat
- Pression initiale
- Niveau d'eau insuffisant
- Déclenchement de la pompe de l'huile
- Déclenchement de la pompe de l'eau (tour et compresseur)

- Déclenchement du relai thermique du moteur de la tour
- Déclenchement du relai thermique du moteur du compresseur
- Déclenchement du relai thermique du moteur de la pompe d'eau
- Déclenchement du relai thermique du moteur de la pompe de l'huile
- Electrovanes : excités (à charge), désexcités (à

vide) Si toutes les conditions sont satisfaites alors :

- Démarrage du compresseur à l'aide du bouton marche.
- Déclenchement de relais thermique de la pompe d'huile.
- Le moteur du compresseur couplé en étoile se met en route pendant 5 seconde puis il passe au couplage triangle, à ce moment le moteur tourne à vide avant qu'il fonctionne en charge.

b) Pompe d'huile :

- Déclenchement du relai thermique de la pompe d'huile dans le cas d'un défaut : FC_OIL_PUMP_OVERLOAD
- Réglage du thermostat à une certaine température qu'il ne doit pas dépasser : FC_OIL_TEMP_CONTROL
- Pressostat de basse pression : FC_OIL_PRESSION_LL (réglé à 10 bars)
- Pressostat de haute pression : FC_OIL_PRESSION_HH (règle à 60 bars)
- Pompe d'huile marche en automatique : AUTO_OIL_PUMP_M
- Pompe d'huile marche en manuel : MANUAL_OIL_PUMP_M

c) Pompe d'eau du compresseur :

- Déclenchement du relai thermique de la pompe d'eau du compresseur dans le cas d'un défaut : FC_WATER_PUMP_OVERLOAD
- Réglage du pressostat de deuxième étage à une certaine haute température qu'il ne faut pas dépasser sinon la phase se coupe et le compresseur s'arrête : FC_WATER_HH_TEMP_STAGE_2 (réglé à 40°C).
- Réglage du pressostat du premier et troisième étage à une certaine haute température qu'il ne faut pas dépasser sinon la phase se coupe et le compresseur s'arrête, le premier et le troisième étage sont reliés, ils travaillent en même temps : FC_WATER_HH_TEMP_STAGE_1_3
- Il ne faut pas avoir une basse pression d'eau : FC_WATER_LL_PRESSION
- La pompe d'eau du compresseur est en marche : FC_WATER_PUMP_RUNING

- La pompe d'eau du compresseur marche en automatique : AUTO_WATER_PUMP_M
- La pompe d'eau du compresseur marche en manuel : MANUAL_WATER_PUMP_M

d) Pompe d'eau de la tour :

- Déclenchement du relai thermique de sécheur d'air : FC_AIR_DRYER_OVERLOAD
- Déclenchement d'alarme de sécheur : FC_AIR_DRYER_ALARM
- Déclenchement du relai thermique de la pompe d'eau de la tour : FC_WATER_SPRAY_OVERLOAD
- Marche de la pompe d'eau de la tour en local : FC_WATER_SPRAY_LOCAL_RUNNING
- Quand le niveau d'eau est bon, le contact sera fermé, cela permet à une tension de 24vdc d'entrer dans l'automate. Et dans le cas contraire le contact sera ouvert et la tension ne passe pas : FC_WATER_SPRAY_LEVEL
- Déclenchement du relai thermique du moteur de la tour : FC_FUN_MOTOR_OVERLOAD
- Le moteur d'eau de la tour marche en automatique : AUTO_TOUR_M
- Le moteur d'eau de la tour marche en manuel : MANUAL_TOUR_M
- Les paramètres de fonctionnement qui sont répertoriés dans le tableau III.1

Tableau III.1 : Paramètres de fonctionnement

Etages	Pression	Température
Premier étage	Alarme : 3 bars	210°C
	Blocage : 4 bars	210°C
Deuxième étage	Alarme : 13 bars	210°C
	Blocage : 14 bars	210°C
Troisième étage	Alarme : 42 bars	210°C
	Blocage : 43 bars	210°C

- Pression du travail : Maximum : 36 bars
- Température d'huile : Consigne : 12°C / Blocage : 60°C
- Température eau : Consigne : 30°C / Blocage : 50°C
- Temps de marche à vide : 300secondes

- Temps de sirène : 5secondes
- Retard mise en marche : 15secondes
- Temps d'arrêt en automatique : 200secondes
- Pression d'huile : 10bars
- Pression premier étage : 2 bars
- Température premier étage : 40°C
- Pression deuxième étage : 12 bars
- Température du deuxième étage : 40°C
- Pression troisième étage : 40bars
- Température du troisième étage : 40°C

III.3.1. Rôle de cahier des charges

Le cahier de charge joue un rôle essentiel en définissant les besoins, les attentes et les spécifications d'un projet. Il permet de clarifier les objectifs, les fonctionnalités requises. En agissant comme un guide de référence, il favorise une compréhension commune entre les parties prenantes et facilite la planification, la conception et l'exécution du projet. Le cahier des charges assure également l'évaluation et la validation des résultats obtenus, garantissant ainsi la satisfaction des exigences établies.

III.4. Programmation

III.4.1. Table des variables

Lors de la programmation d'un programme, il est nécessaire de définir une liste de variables qui seront utilisées. L'utilisation de noms appropriés dans la table des mnémoniques facilite la compréhension et la manipulation du programme. Pour chaque variable, nous spécifions le nom, le type de données et l'adresse.

La figure III.1 illustre les variables utilisées dans le programme.

Variables API								
	Nom	Table des variables	Type de données	Adresse	Réma...	Acces...	Ecritu...	Visibl...
1	I620_BT_URGENCE	Table de variables s..	Bool	%I0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2	I621_BT_STOP	Table de variables s..	Bool	%I0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3	I622_BT_START	Table de variables s..	Bool	%I0.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
4	I623_BT_ANTICIPATED_START	Table de variables s..	Bool	%I0.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
5	I624_FC_COMPRESSOR_RUNING	Table de variables s..	Bool	%I0.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
6	I625_FC_MOTOR_OVERLOAD	Table de variables s..	Bool	%I0.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
7	I626_FC_OIL_PUMP_OVERLOAD	Table de variables s..	Bool	%I0.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
8	I627_FC_COOLING_FAULT	Table de variables s..	Bool	%I0.7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
9	I631_FC_WATER_PUMP_OVERL...	Table de variables s..	Bool	%I12.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
10	I630_FC_WATER_PUMP_RUNING	Table de variables s..	Bool	%I12.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
11	I634_FC_WATER_LL_PRESSION	Table de variables s..	Bool	%I12.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
12	I632_FC_AIR_DRYER_OVERLOAD	Table de variables s..	Bool	%I12.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
13	I633_FC_AIR_DRYER_ALARM	Table de variables s..	Bool	%I12.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
14	I635_FC_WATER_HH_TEMP_ST...	Table de variables s..	Bool	%I12.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
15	I636_FC_WATER_HH_TEMP_ST...	Table de variables s..	Bool	%I12.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
16	I637_FC_OIL_TEMP_CONTROL	Table de variables s..	Bool	%I12.7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
17	I642_FC_OIL_LL_PRSSION	Table de variables s..	Bool	%I16.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
18	I641_FC_STAGE_3_HH_PRESSION	Table de variables s..	Bool	%I16.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
19	I640_FC_STAGE_3_DELIVERY_H...	Table de variables s..	Bool	%I16.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
20	I638_FC_FUN_MOTOR_OVERL...	Table de variables s..	Bool	%I20.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
21	I639_FC_WATER_SPRAY_OVERL...	Table de variables s..	Bool	%I20.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
22	I6310_FC_SELECTOR_REMOE_L...	Table de variables s..	Bool	%I20.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
23	I6311_FC_WATER_SPRAY_LOC...	Table de variables s..	Bool	%I20.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
24	I6312_FC_FUN_LOCAL_RUNING	Table de variables s..	Bool	%I20.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
25	I6313_FC_WATER_SPRAY_LEVEL	Table de variables s..	Bool	%I20.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
26	FINAL_PRESSION_RAW	Table de variables s..	Word	%IW96	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
27	WATER_TEMP_RAW	Table de variables s..	Word	%IW98	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
28	OIL_TEMP_RAW	Table de variables s..	Word	%IW100	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
29	Q622_MOTOR_OIL_PUMP_Q	Table de variables s..	Bool	%Q0.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
30	Q623_MOTOR_WATER_PUMP_Q	Table de variables s..	Bool	%Q0.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
31	Q620_COMPRESSOR_START_ST...	Table de variables s..	Bool	%Q0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
32	Q624_HORN	Table de variables s..	Bool	%Q0.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
33	Q625_START_UP_FLASHER	Table de variables s..	Bool	%Q0.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
34	Q630_ALARM_TRIP_FLASHER	Table de variables s..	Bool	%Q12.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
35	Q633_COMPRESSOR_RUNING	Table de variables s..	Bool	%Q12.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
36	Q634_UNLOAD_SELOID_YV	Table de variables s..	Bool	%Q12.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
37	Q636_WATER_SPRAY_PUMP_Q	Table de variables s..	Bool	%Q12.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
38	Q637_MOTOR_FUN_CONTROL...	Table de variables s..	Bool	%Q12.7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
39	AUTO_TOUR_M	Table de variables s..	Bool	%M200.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
40	AUTO_OIL_PUMP_M	Table de variables s..	Bool	%M200.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
41	START_STEP_0	Table de variables s..	Bool	%M300.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
42	START_STEP_1	Table de variables s..	Bool	%M300.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
43	AUTO_WATER_PUMP_M	Table de variables s..	Bool	%M200.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
44	START_STEP_2	Table de variables s..	Bool	%M300.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
45	START_STEP_3	Table de variables s..	Bool	%M300.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
46	START_STEP_4	Table de variables s..	Bool	%M300.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
47	AUTO_COMPRESSOR_M	Table de variables s..	Bool	%M200.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
48	MANUAL_OIL_PUMP_M	Table de variables s..	Bool	%M201.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
49	MANUAL_WATER_PUMP_M	Table de variables s..	Bool	%M201.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
50	MANUAL_TOUR_M	Table de variables s..	Bool	%M201.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

51		FINAL_PRESSION_IN	Table de variables s..	UInt	%MW500	<input type="checkbox"/>			
52		WATER_TEMP_IN	Table de variables s..	UInt	%MW502	<input type="checkbox"/>			
53		OIL_TEMP_IN	Table de variables s..	UInt	%MW504	<input type="checkbox"/>			
54		FINAL_PRESSION_OFFSET	Table de variables s..	UInt	%MW10	<input checked="" type="checkbox"/>			
55		FINAL_PRESSION_FACTEUR	Table de variables s..	Real	%MD30	<input checked="" type="checkbox"/>			
56		WATER_TEMP_OFFSET	Table de variables s..	UInt	%MW12	<input checked="" type="checkbox"/>			
57		OIL_TEMP_OFFSET	Table de variables s..	UInt	%MW14	<input checked="" type="checkbox"/>			
58		WATER_TEMP_FACTEUR	Table de variables s..	Real	%MD34	<input checked="" type="checkbox"/>			
59		OIL_TEMP_FACTEUR	Table de variables s..	Real	%MD38	<input checked="" type="checkbox"/>			
60		FINAL_PRESSION_REAL	Table de variables s..	Real	%MD506	<input type="checkbox"/>			
61		WATER_TEMP_REAL	Table de variables s..	Real	%MD510	<input type="checkbox"/>			
62		OIL_TEMP_REAL	Table de variables s..	Real	%MD514	<input type="checkbox"/>			
63		FINAL_PRESSION_TMP	Table de variables s..	UInt	%MW518	<input type="checkbox"/>			
64		WATER_TEMP_TMP	Table de variables s..	UInt	%MW520	<input type="checkbox"/>			
65		OIL_TEMP_TMP	Table de variables s..	UInt	%MW522	<input type="checkbox"/>			
66		AUTO_LOAD_UNLOAD	Table de variables s..	Bool	%M200.5	<input type="checkbox"/>			
67		START_STEP_5	Table de variables s..	Bool	%M300.5	<input type="checkbox"/>			
68		Q633_COMPRESSOR_RUNNING...	Table de variables s..	Bool	%M300.6	<input type="checkbox"/>			
69		AUTO_REGULE_ENABLE	Table de variables s..	Bool	%M200.6	<input type="checkbox"/>			
70		STOP_SEQUENCE_1	Table de variables s..	Bool	%M201.3	<input type="checkbox"/>			
71		STOP_SEQUENCE_ALL	Table de variables s..	Bool	%M201.4	<input type="checkbox"/>			
72		FINAL_PRESSION_HH	Table de variables s..	Real	%MD40	<input checked="" type="checkbox"/>			
73		WATER_TEMP_HH	Table de variables s..	Real	%MD44	<input checked="" type="checkbox"/>			
74		OIL_TEMP_HH	Table de variables s..	Real	%MD48	<input checked="" type="checkbox"/>			
75		ALARM_FINAL_PRESSION_HH	Table de variables s..	Bool	%M201.5	<input type="checkbox"/>			
76		ALARM_WATER_TEMP_HH	Table de variables s..	Bool	%M201.6	<input type="checkbox"/>			
77		ALARM_OIL_TEMP_HH	Table de variables s..	Bool	%M201.7	<input type="checkbox"/>			
78		FINAL_PRESSION_CONSIGNE_H	Table de variables s..	Real	%MD52	<input checked="" type="checkbox"/>			
79		FINAL_PRESSION_CONSIGNE_L	Table de variables s..	Real	%MD56	<input checked="" type="checkbox"/>			
80		AUTO_LOAD_LAMP	Table de variables s..	Bool	%M202.0	<input type="checkbox"/>			
81		AUTO_UNLOAD_LAMP	Table de variables s..	Bool	%M202.1	<input type="checkbox"/>			
82		ALARME_BT_URGENCE	Table de variables s..	Bool	%M310.2	<input type="checkbox"/>			
83		ALARME_COMPRESSOR	Table de variables s..	Bool	%M310.3	<input type="checkbox"/>			
84		ALARME_QF_OIL	Table de variables s..	Bool	%M310.4	<input type="checkbox"/>			
85		ALARME_QF_WATER	Table de variables s..	Bool	%M310.5	<input type="checkbox"/>			
86		ALARME_AIR_DRYER	Table de variables s..	Bool	%M310.6	<input type="checkbox"/>			
87		ALARME_WATER_PRESSION_LL	Table de variables s..	Bool	%M310.7	<input type="checkbox"/>			
88		ALARME_TEMP_HH_2_STAGE	Table de variables s..	Bool	%M311.0	<input type="checkbox"/>			
89		ALARME_TEMP_HH_1_3_STAGE	Table de variables s..	Bool	%M311.1	<input type="checkbox"/>			
90		ALARME_OIL_TEMP_CONTROLL...	Table de variables s..	Bool	%M311.2	<input type="checkbox"/>			
91		ALARME_QF_FUN	Table de variables s..	Bool	%M311.3	<input type="checkbox"/>			
92		ALARME_QF_WATER_PRAY	Table de variables s..	Bool	%M311.4	<input type="checkbox"/>			
93		ALARME_WATER_LEVEL_LL	Table de variables s..	Bool	%M311.5	<input type="checkbox"/>			
94		ALARME_TEMP_HH_3_STAGE	Table de variables s..	Bool	%M311.6	<input type="checkbox"/>			
95		ALARME_PRESSION_HH_3_STA...	Table de variables s..	Bool	%M311.7	<input type="checkbox"/>			
96		ALARME_PRESSION_LL_OIL	Table de variables s..	Bool	%M312.0	<input type="checkbox"/>			
97		ALARME_FINAL_PRESSION_HH_A	Table de variables s..	Bool	%M312.1	<input type="checkbox"/>			
98		ALARM_WATER_TEMP_HH_A	Table de variables s..	Bool	%M312.2	<input type="checkbox"/>			
99		ALARM_OIL_TEMP_HH_A	Table de variables s..	Bool	%M312.3	<input type="checkbox"/>			
100		AUTO_START_100	Table de variables s..	Bool	%M400.1	<input type="checkbox"/>			

101		AUTO_START_0	Table de variables s..	Bool	%M400.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
102		RETARD_COMP_START	Table de variables s..	UInt	%MW16	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
103		RETARD_AUTO_STOP	Table de variables s..	UInt	%MW18	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
104		TEMPS_MARCHE_VIDE	Table de variables s..	UInt	%MW20	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
105		TEMPS_SIRENE	Table de variables s..	UInt	%MW22	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
106		TEMPS_PRELUBRIFICATION	Table de variables s..	UInt	%MW24	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
107		RETARD_COMP_START_MD	Table de variables s..	DInt	%MD524	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
108		OIL_TEMP_CONSIGNE	Table de variables s..	Real	%MD66	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
109		WATER_TEMP_CONSIGNE	Table de variables s..	Real	%MD70	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
110		RETARD_AUTO_STOP_MD	Table de variables s..	DInt	%MD528	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
111		TEMPS_MARCHE_VIDE_MD	Table de variables s..	DInt	%MD532	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
112		TEMPS_SIRENE_MD	Table de variables s..	DInt	%MD536	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
113		TEMPS_PRELUBRIFICATION_MD	Table de variables s..	DInt	%MD540	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
114		WATER_TEMP_CONSIGNE_L	Table de variables s..	Real	%MD544	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
115		FUN_MOTOR_REGULE_S	Table de variables s..	Bool	%M301.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
116		FUN_MOTOR_REGULE_R	Table de variables s..	Bool	%M301.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
117		START_STEP_HORN_ON	Table de variables s..	Bool	%M301.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
118		START_STEP_HORN_OFF	Table de variables s..	Bool	%M301.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
119		ALARME GROUPE	Table de variables s..	Int	%MW310	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
120		<input type="text" value="<Ajouter>"/>				<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Figure III.1 : Table des variables

III.4.2. Blocs

Pour assurer le fonctionnement d'un système automatisé, il est nécessaire de charger les différents programmes et données dans l'automate. Les types de blocs utilisés pour cela sont les blocs suivants :

III.4.2.1. Bloc d'organisation (OB1)

Le bloc d'organisation est généré automatiquement lors de la création d'un projet II constitue l'interface entre le système d'exploitation et le programme que nous avons élaboré.

La figure III.2 représente le bloc d'organisation.

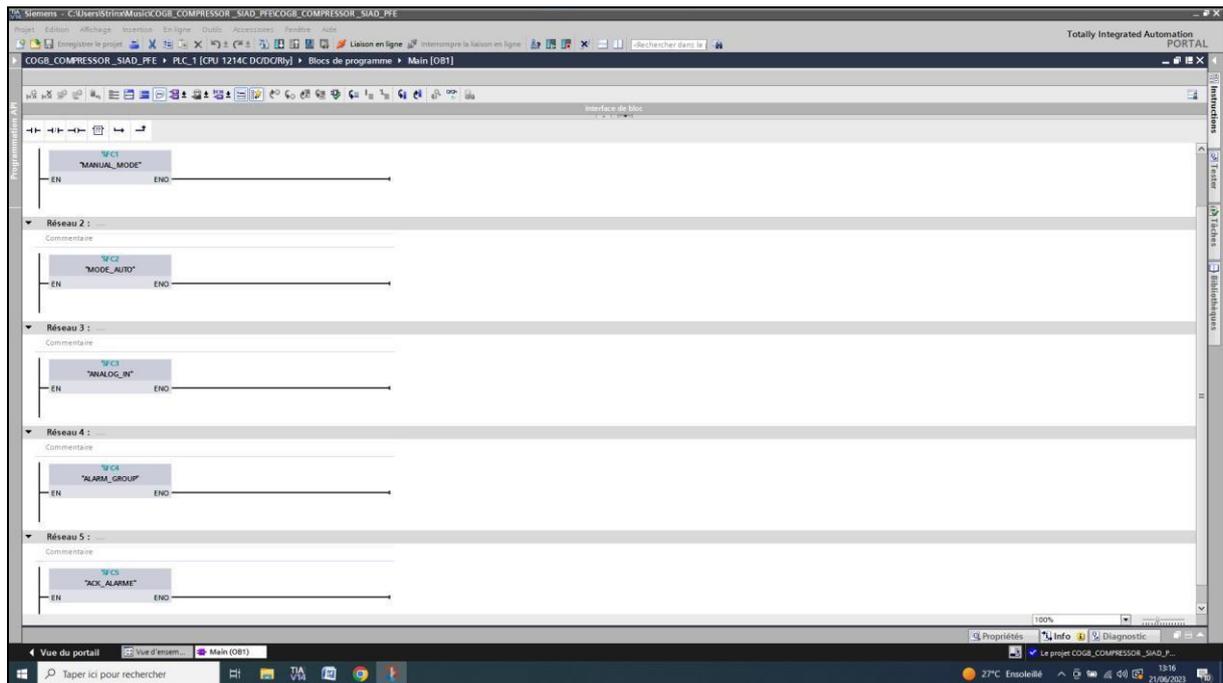


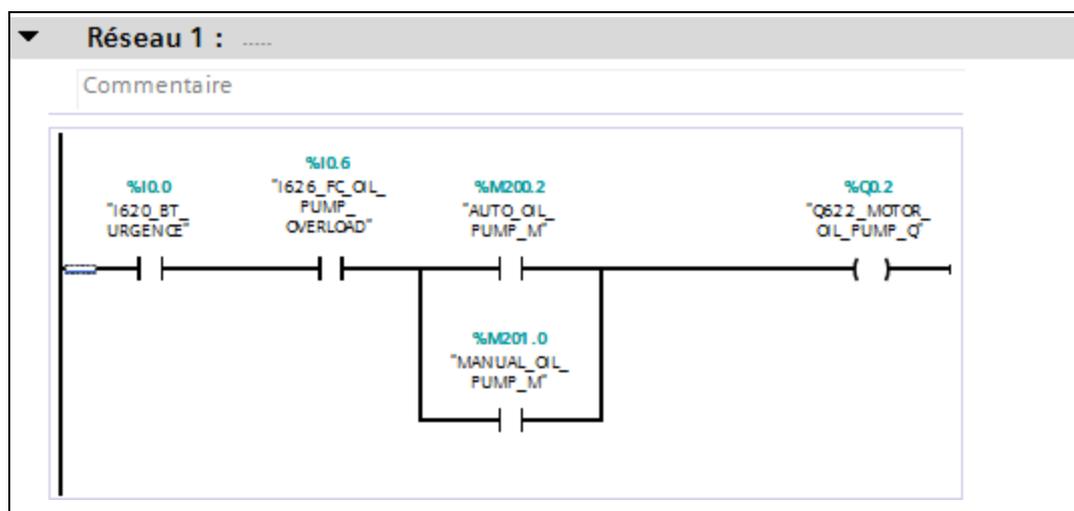
Figure III.2 : Bloc d'organisation

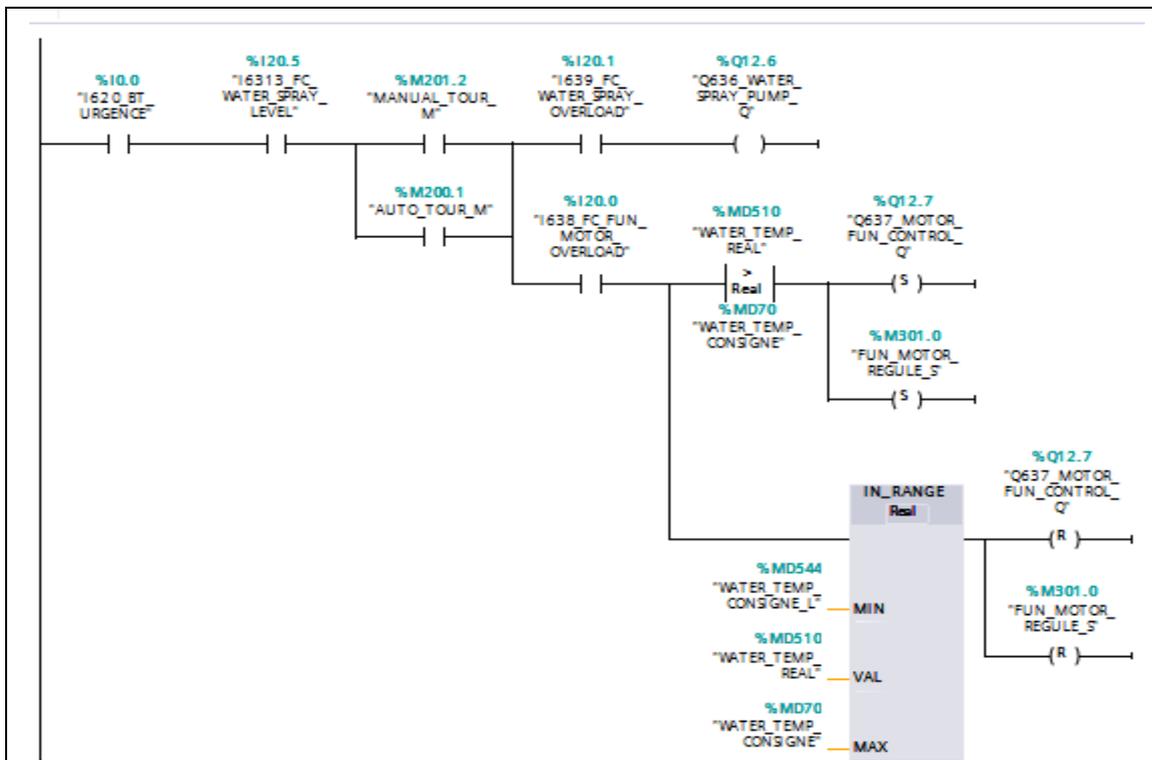
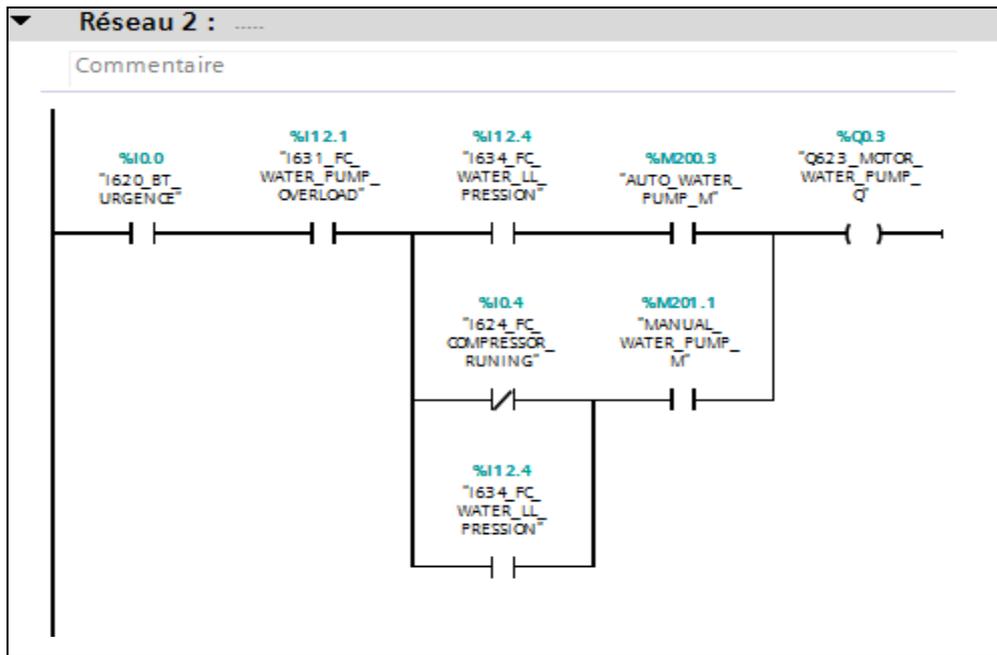
III.4.2.2. Bloc fonction (FC)

Ce sont des blocs de code qui ne conservent pas en mémoire les données des variables temporaires une fois que la fonction est exécutée.

Notre programme se constitue de plusieurs blocs, nous allons parler du mode manuel et le reste des blocs seront cités dans l'annexe.

La figure III.3 représente le mode manuel.





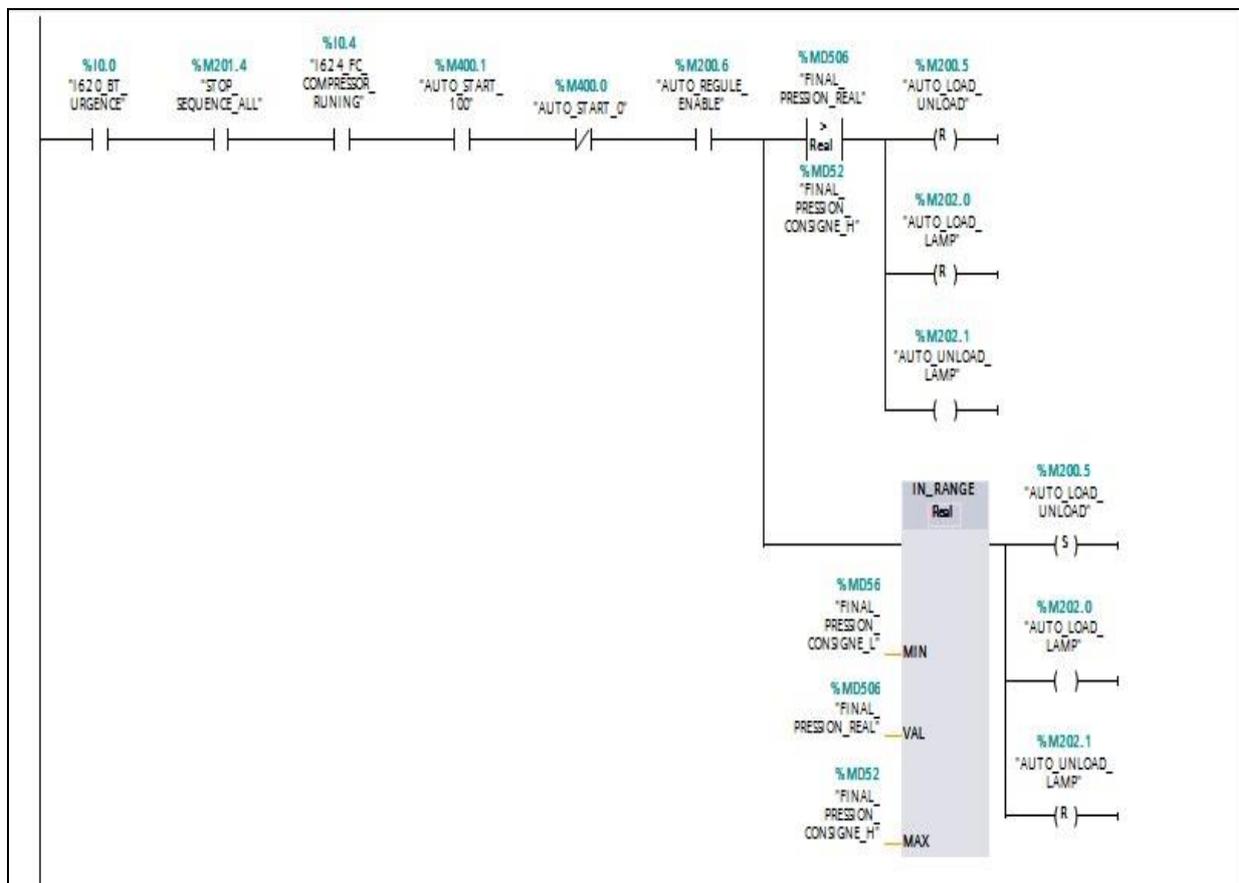
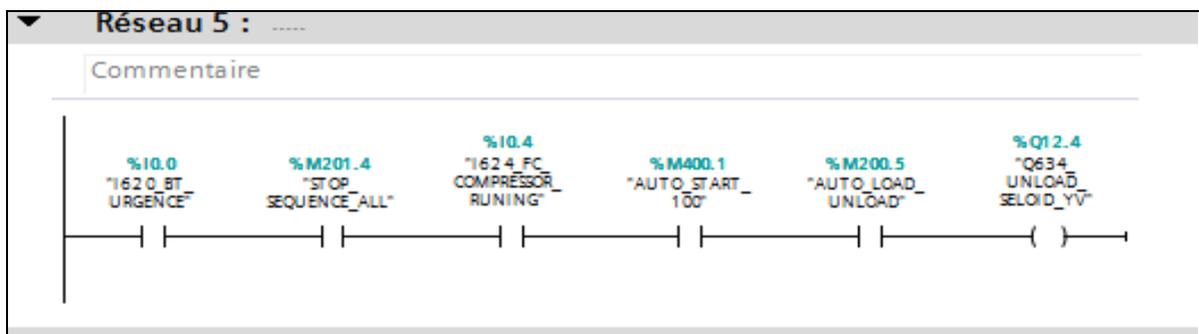
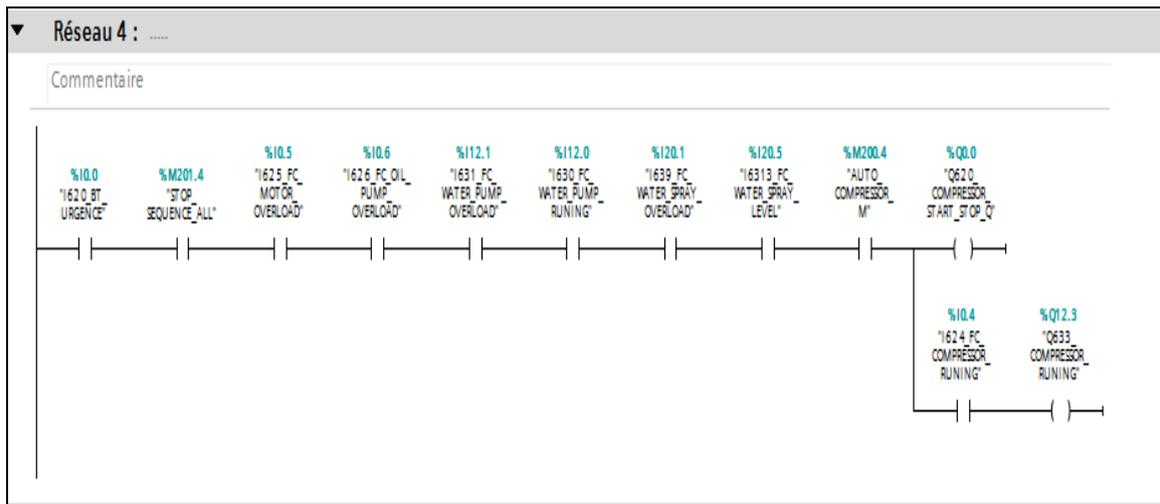


Figure III.3 : Mode manuel

Interprétations

- Le bouton d'arrêt d'urgence est obligatoire, la sécurité intervient avant toute autre chose, donc la priorité est pour l'urgence que ce soit en mode manuel ou automatique.
- Si le relai thermique de la pompe d'huile est déclenché donc il y a un problème, ce qui inclut la pompe d'huile ne marche pas. Sinon on appuis sur le bouton marche et on aura comme sortie moteur de la pompe d'huile démarre (Réseau 1).
- Pour que le moteur de la pompe d'eau du compresseur démarre, il faut un minimum de pression d'eau (Réseau 2).
- Pour que le moteur de la tour démarre, il doit comparer la température de l'eau par rapport à la consigne que nous avons attribué, si elle est supérieure ou égale on aura une sortie (S) inclut le moteur de la tour démarre. Dans le cas contraire on aura une sortie (R) et le moteur ne démarre pas (Réseau 3).
- Le compresseur se met en marche s'il Il n'y a pas d'erreurs qui vont interrompre les signaux que ça soit en mode manuel ou automatique et quand les conditions des réseaux précédents sont satisfaites (Réseau 4).
- En mode automatique, la vanne sera excitée (Réseau 5).
- Nous avons le capteur de pression finale, il doit comparer la pression finale d'air, à 36 bars, il se met à charge lorsque la pression finale est inférieure à 36bars, et à vide lorsqu'elle est supérieure. (Réseau 6).

III.4.3. Simulation

Une fois les programmes créent, TIA PORTAL offre la possibilité de les simuler à l'aide de son extension PLC SIM. Cette fonctionnalité permet de compiler le programme et de le charger dans un automate simulé en utilisant la barre de simulation située en haut de la fenêtre.

La figure III.4 représente la barre de simulation de TIA PORTAL.

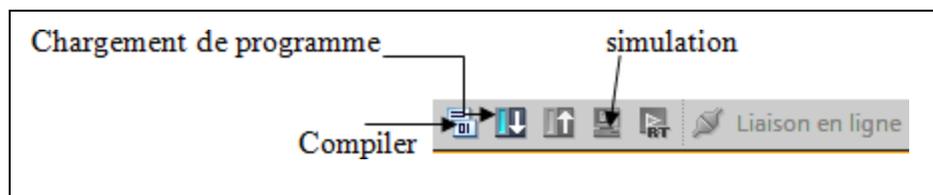


Figure III.4 : Barre de simulation de TIA PORTAL

Avant de simuler le fonctionnement de notre programme, nous allons le charger dans l'automate virtuel puis le compiler, comme le représente la figure III.5.

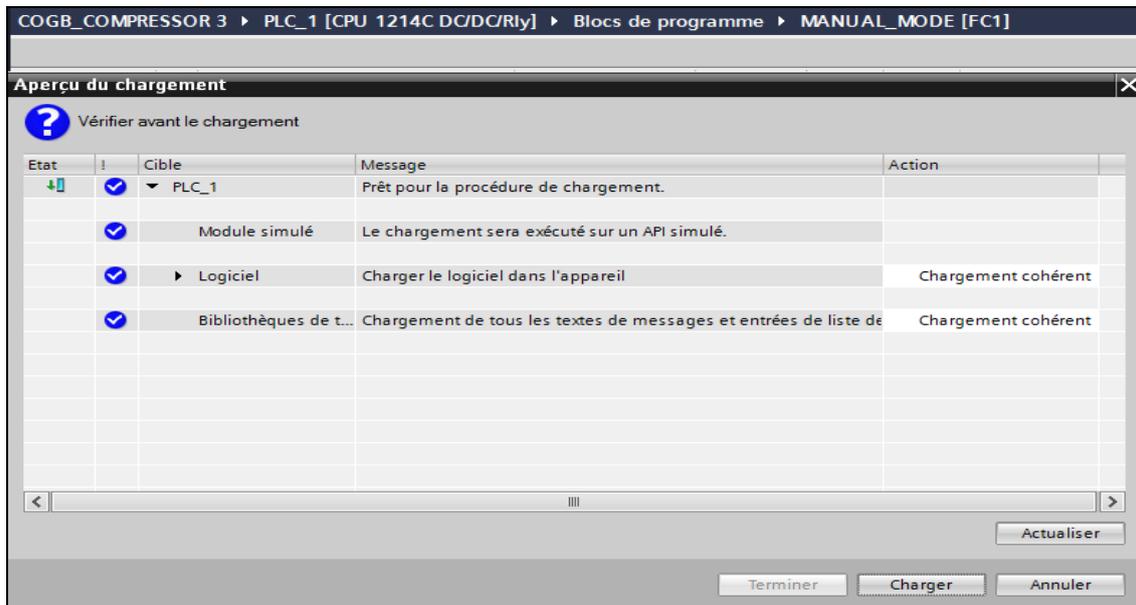


Figure III.5 : Chargement du programme dans l'appareil

Grâce à l'application PLCSIM, nous avons la possibilité de simuler le fonctionnement d'un automate programmable S7-1200. Cela nous permet de tester nos programmes de commande directement à partir de PLCSIM, sans avoir besoin d'établir une connexion avec le matériel S7-1200. PLCSIM offre une interface utilisateur graphique qui permet de visualiser et de modifier les variables du programme de commande. On peut exécuter la CPU de simulation en mode cycle unique ou en mode cycle continu, et également modifier l'état de fonctionnement de l'API de simulation.

La figure III.6 représente la table de simulation par forçage.

<input checked="" type="checkbox"/>	*I620_BT_URGEN...	%I0.0:P	Bool	TRUE	<input checked="" type="checkbox"/>	FALSE	<input checked="" type="checkbox"/>	
<input checked="" type="checkbox"/>	*I621_BT_STOP*:P	%I0.1:P	Bool	FALSE	<input type="checkbox"/>	FALSE	<input checked="" type="checkbox"/>	
<input checked="" type="checkbox"/>	*I622_BT_START*:P	%I0.2:P	Bool	FALSE	<input type="checkbox"/>	FALSE	<input checked="" type="checkbox"/>	
<input checked="" type="checkbox"/>	*I623_BT_ANTI...	%I0.3:P	Bool	FALSE	<input type="checkbox"/>	FALSE	<input type="checkbox"/>	
<input checked="" type="checkbox"/>	*I624_FC_COMP...	%I0.4:P	Bool	FALSE	<input type="checkbox"/>	FALSE	<input checked="" type="checkbox"/>	
<input checked="" type="checkbox"/>	*I625_FC_MOTO...	%I0.5:P	Bool	FALSE	<input type="checkbox"/>	FALSE	<input checked="" type="checkbox"/>	
<input checked="" type="checkbox"/>	*I626_FC_OIL...	%I0.6:P	Bool	TRUE	<input checked="" type="checkbox"/>	FALSE	<input checked="" type="checkbox"/>	
<input checked="" type="checkbox"/>	*I627_FC_COOLI...	%I0.7:P	Bool	FALSE	<input type="checkbox"/>	FALSE	<input type="checkbox"/>	
<input checked="" type="checkbox"/>	*I631_FC_WATER...	%I12.1:P	Bool	FALSE	<input type="checkbox"/>	FALSE	<input checked="" type="checkbox"/>	

Figure III.6 : Table de simulation par forçage

III.4.3.1. Simulation en mode manuel

Parmi les différentes simulations réalisées pour chaque mode, nous avons choisi une simulation représentative pour étayer les résultats obtenus.

La figure III.7 représente le démarrage de la pompe d'huile.

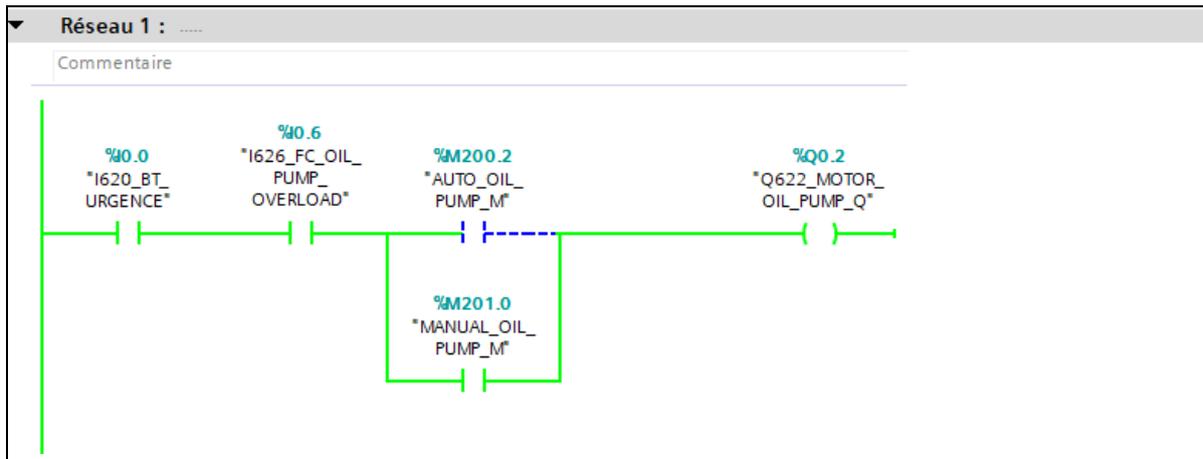


Figure III.7 : Démarrage de la pompe d'huile en mode manuel

Pour que notre pompe d'huile fonctionne il faut forcer les entrées suivantes :

"I620_BT_URGENCE", "I626_FC_OIL_PUMP_OVERLOAD", "AUTO_OIL_PUMP_M", "MANUAL_OIL_PUMP_M".

III.4.3.2. Simulation de bloc des entrées analogiques

La figure III.8 représente la pression finale.

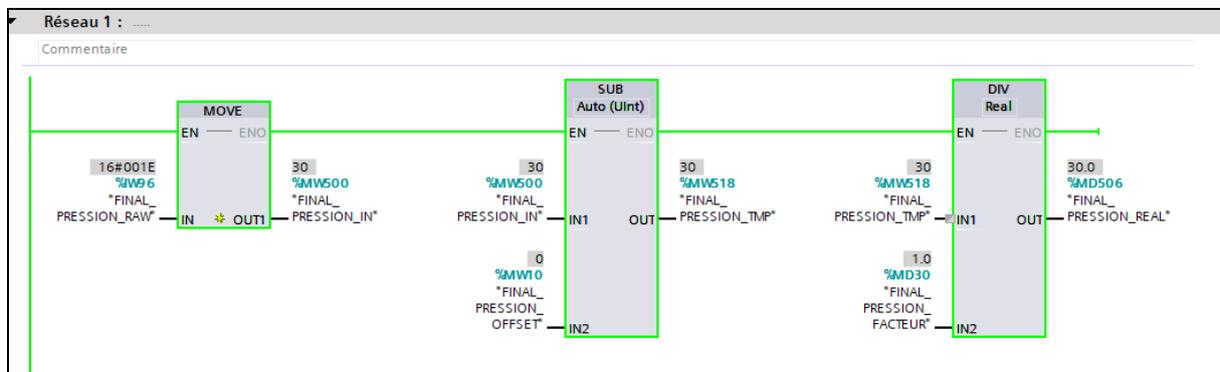


Figure III.8 : Pression finale

Interprétations

Les blocs de substraction et division sont utilisés pour minimiser le décalage entre la vraie pression et la pression affichée sur l'écran.

III.4.3.3. Régulation

La régulation automatique va nous permettre de découvrir l'effet de la pression sur la mise en charge du compresseur, et aussi l'effet de la température sur la tour.

III.4.3.3.1. Régulation de pression

1. Premier cas (Pression <36 bars)

Supposons qu'on a une pression de 25 bars, La figure III.10 montre les résultats de simulation :

Addresses				
	Nom	Adresse	Format d'affichage	Valeur visualisée/de .
01	*FINAL_PRESSION_RA...	%IW96:P	DEC	25
01	*WATER_TEMP_RAW":P	%IW98:P	DEC	32
01	*OIL_TEMP_RAW":P	%IW100...	DEC	18
	----	%IW102...	DEC	0

Figure III.9 : Réglage de pression à 25 bars

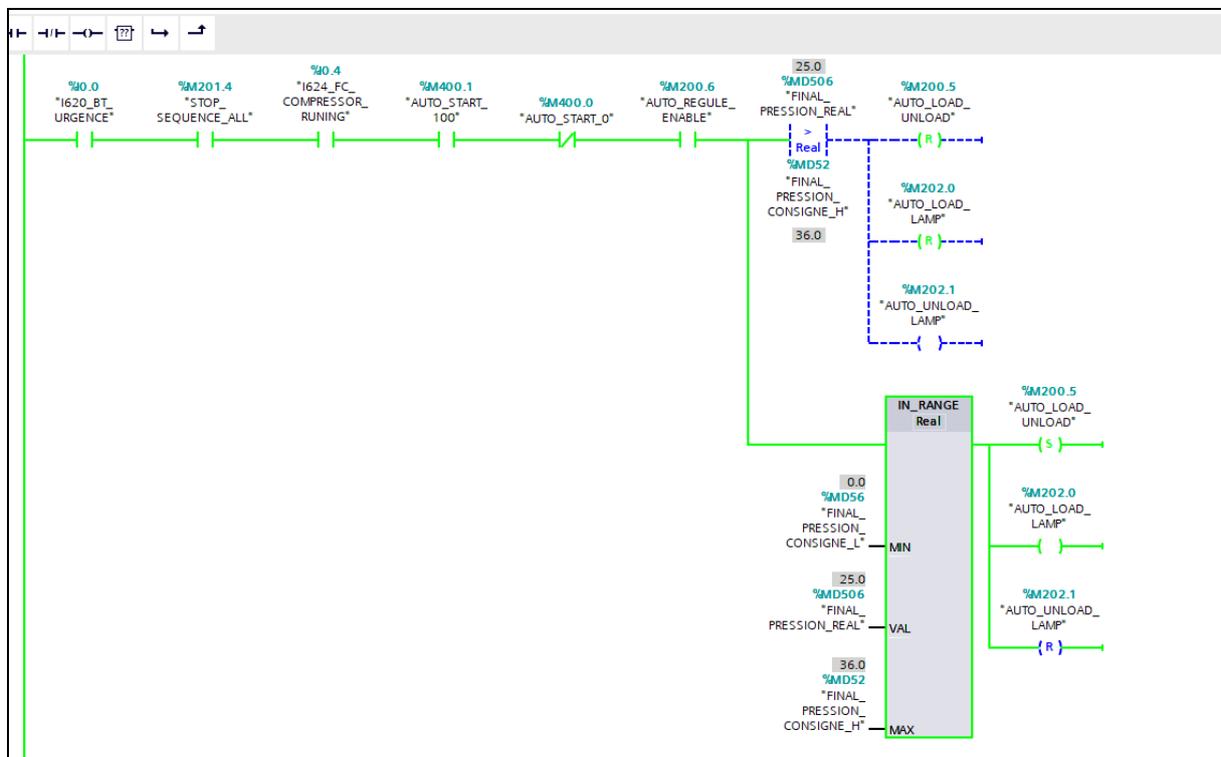


Figure III.10 : Pression de 25 bars

Interprétations

Lorsque la pression est inférieure à 36 bars, le compresseur va se mettre à charge.

2. Deuxième cas (Pression > 36 bars)

Supposons qu'on a une pression de 37 bars, La figure III.12 montre les résultats de simulation :

	Nom	Adresse	Format d'affichage	Valeur visualisée/de .
	*FINAL_PRESSION_RA...	%IW96:P	DEC	37
	WATER_TEMP_RAW:P	%IW98:P	DEC	32
	OIL_TEMP_RAW:P	%IW100...	DEC	18
	----	%IW102...	DEC	0

Figure III.11 : Réglage de pression à 37 bars

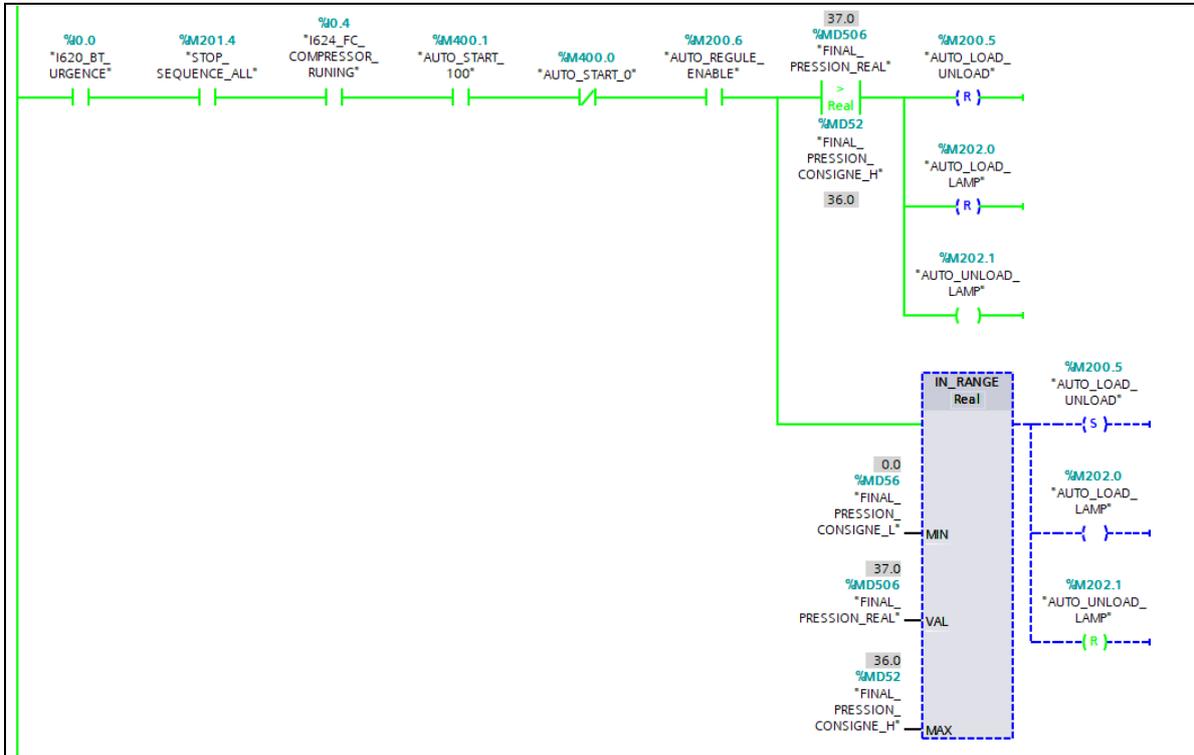


Figure III.12 : Pression à 37 bars

Interprétations

Lorsque la pression est inférieure à 36 bars, cela signifie que le compresseur va se mettre à vide (décharge).

III.4.3.3.2. Régulation de la température de la tour

1. Premier cas (Température < 30 °C)

Supposons qu'on a une température de 15 °C, la figure III.14 montre les résultats de simulation

	Nom	Adresse	Format d'affichage	Valeur visualisée/de .
	*FINAL_PRESSION_RA...	%IW96:P	DEC	35
	WATER_TEMP_RAW:P	%IW98:P	DEC	15
	OIL_TEMP_RAW:P	%IW100...	DEC	18
	----	%IW102...	DEC	0

Figure III.13 : Réglage de la température à 15°C

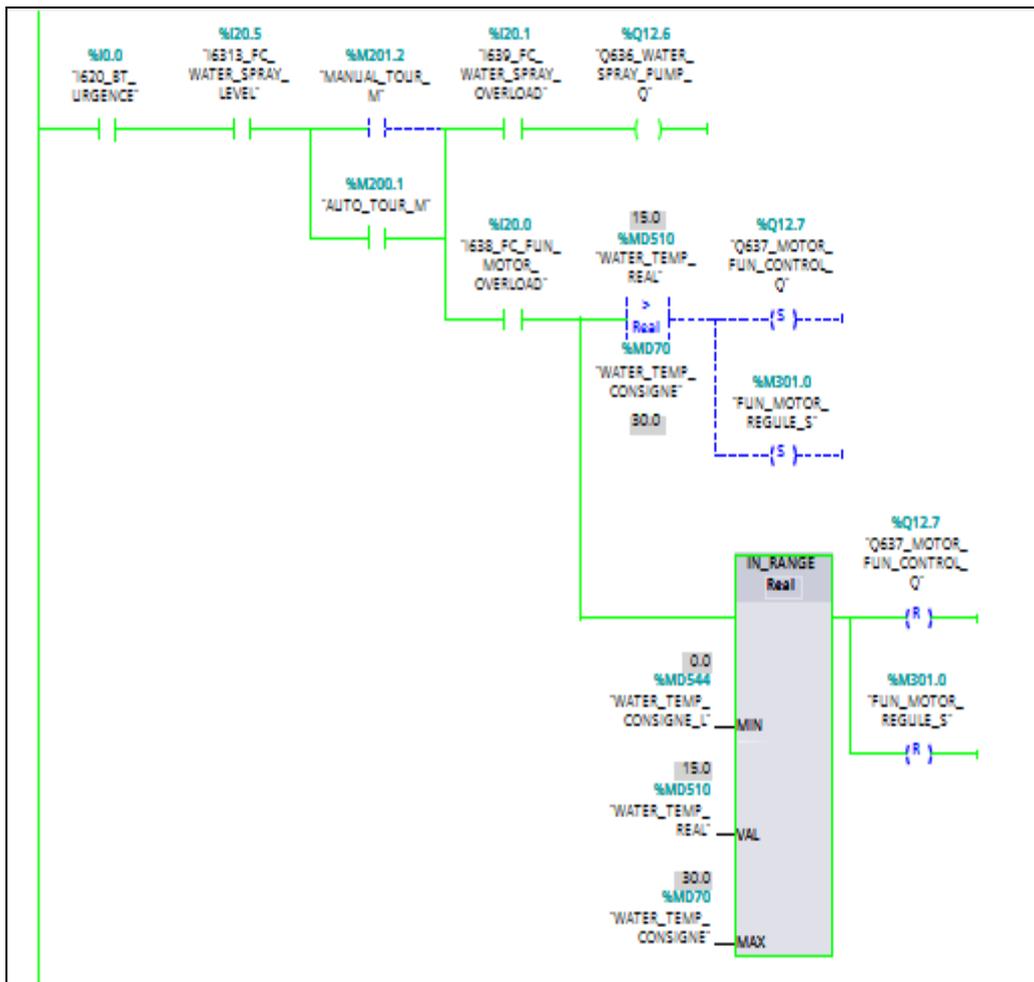


Figure III.14 : Température à 15°C

Interprétations

Lorsque la température est inférieure à 30°C, la tour va se mettre à l'arrêt.

2. Deuxième cas (Température > 30 °C)

Supposons qu'on a une température de 35 °C, la figure III.16 montre les résultats de simulation :

	Nom	Adresse	Format d'affichage	Valeur visualisée/de .
←01	"FINAL_PRESSION_RA...	%IW96:P	DEC	35
←01	"WATER_TEMP_RAW":P	%IW98:P	DEC	35
←01	"OIL_TEMP_RAW":P	%IW100...	DEC	18
	----	%IW102...	DEC	0

Figure III.15 : Réglage de température à 35°C

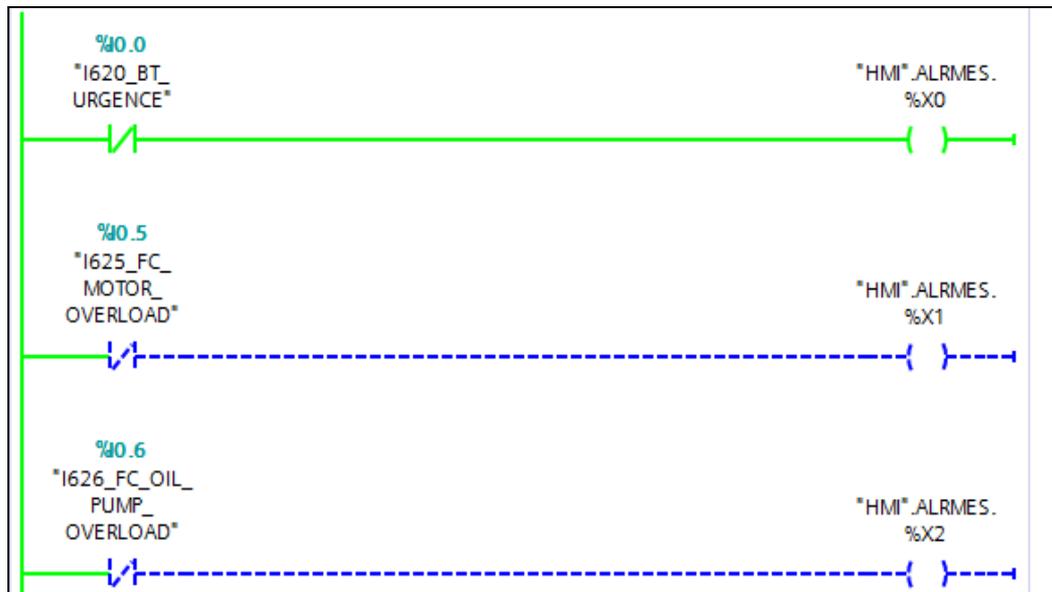


Figure III.18 : Détection de défaut.

Interprétations

Les alarmes servent à détecter et signaler tout dysfonctionnement ou problème potentiel (défaut) afin de les identifier rapidement et de prendre les mesures appropriées.

III.4.4. Supervision

Nous avons créé trois vues pour notre projet :

1. **Vue d'accueil** : Cette vue constitue la première interface avec notre application. Elle offre un aperçu global du système et permet aux utilisateurs de naviguer vers d'autres fonctionnalités.

La figure III.19 représente la vue d'accueil.

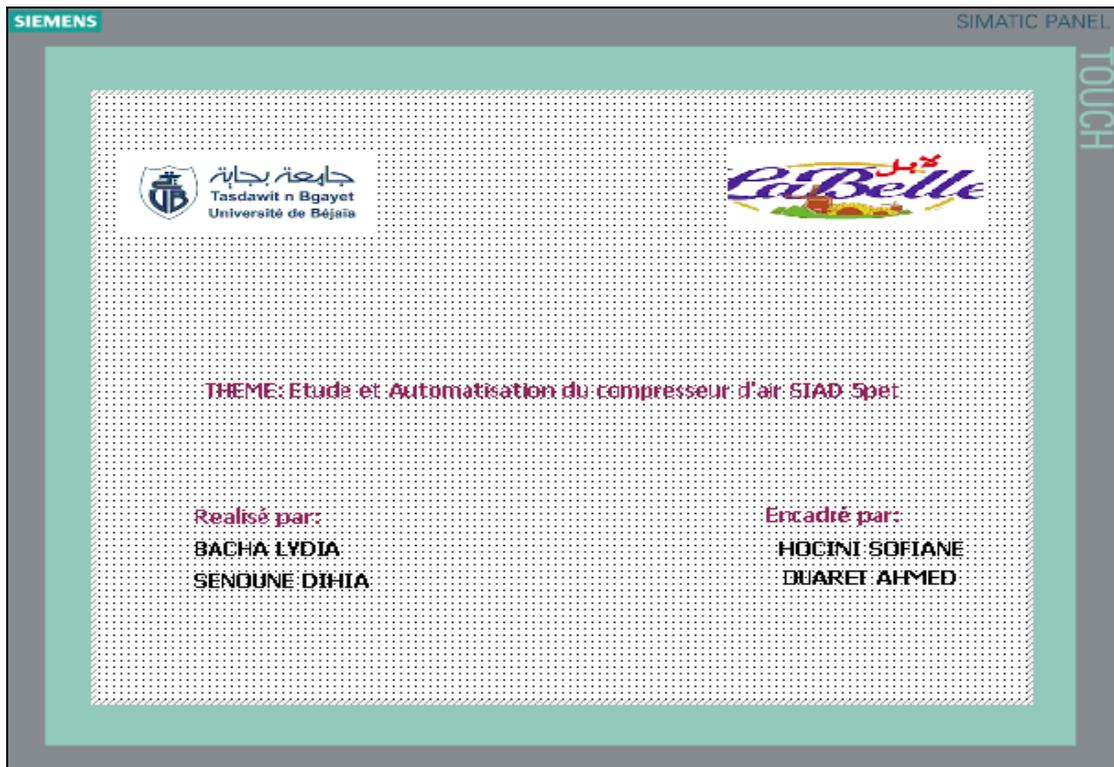


Figure III.19 : Vue d'accueil

2. **Vue du processus** : Cette vue présente en temps réel les données et les informations relatives au processus en cours. Elle affiche les différentes variables de contrôle et les mesures de performance.

La figure III.20 représente la vue de processus

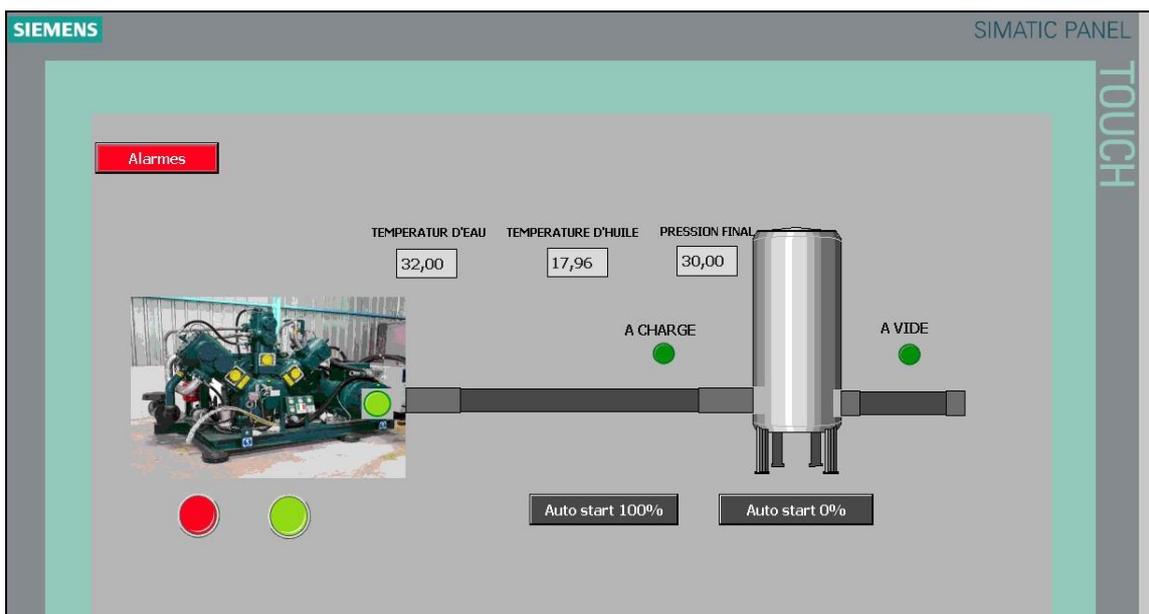


Figure III.20 : Vue de processus

- 3. Vue des alarmes :** Cette vue est dédiée à la gestion des alarmes. Elle affiche les alarmes actives, les alertes et les événements significatifs liés au fonctionnement du système. Les utilisateurs peuvent consulter les détails des alarmes, les traiter et prendre les mesures appropriées pour résoudre les problèmes éventuels.

La figure III.21 représente la vue d'alarmes



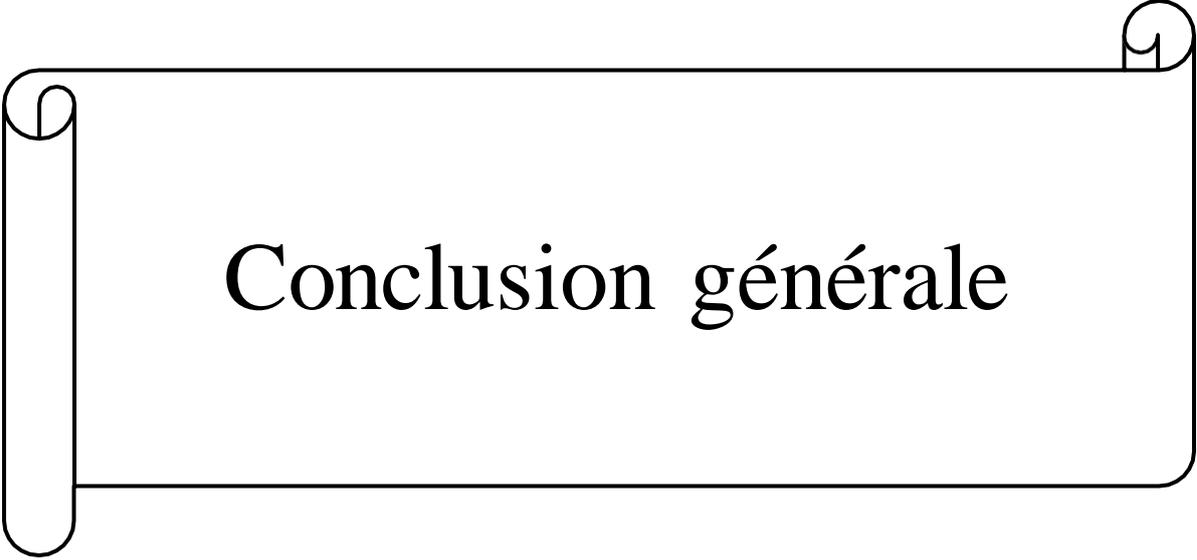
Figure III.21 : Vue d'alarmes

Ces trois vues combinées offrent une interface conviviale et intuitive pour visualiser, superviser et contrôler efficacement le fonctionnement du système.

III.5. Conclusion

Au cours de ce chapitre, nous avons abordé plusieurs aspects importants de la conception du programme du compresseur. Tout d'abord, nous avons développé et introduit les séquences définies dans le cahier des charges, une étape cruciale pour garantir un fonctionnement optimal de la machine.

Ensuite, nous avons présenté en détail les différents réseaux qui assurent le contrôle, la commande et le diagnostic de la machine, fournissant ainsi une vision complète du système. Une autre section a mis en évidence l'interface du compresseur, mettant en évidence les fonctionnalités et les paramètres accessibles aux utilisateurs. Les résultats de la simulation ont confirmé une intégration réussie avec le programme STEP7, tout en soulignant la possibilité d'améliorations en ajoutant certaines fonctionnalités pour enrichir davantage les performances du compresseur.



Conclusion générale

Conclusion générale

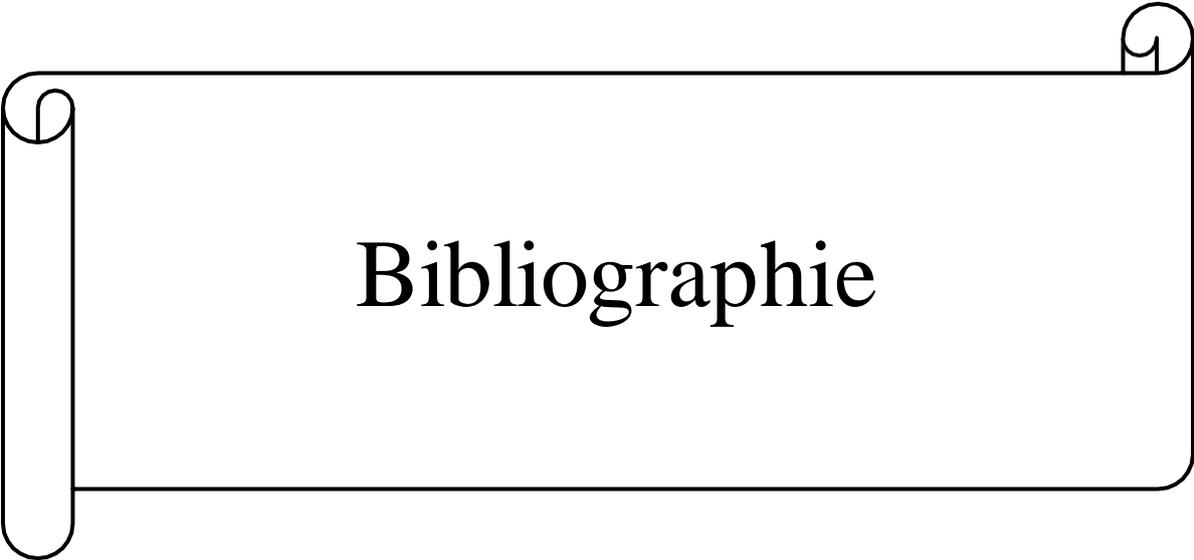
L'objectif de ce projet était d'automatiser et d'étudier la compression d'un compresseur SIAD tempo2-950 en utilisant la programmation sous STEP 7. Par la suite, nous avons supervisé le programme avec le logiciel WinCC flexible. Pour cela, nous avons choisi un automate programmable Siemens S7-1200 en raison de ses capacités de simulation et de supervision.

Au début, nous avons présenté l'entreprise LABELLE, puis nous avons identifié les équipements, les appareils et les matériels constituant l'installation. Ensuite, nous avons fourni des informations générales sur les systèmes automatisés et les API, ce projet a été une expérience enrichissante qui nous a permis d'atteindre plusieurs objectifs importants. Tout d'abord, nous avons réussi à automatiser avec succès la compression d'un compresseur SIAD tempo2-950 en utilisant la programmation sous STEP 7. Cette réalisation démontre notre capacité à mettre en œuvre des systèmes automatisés complexes.

De plus, nous avons également pu étudier en détail le processus de compression et comprendre les interactions entre les différents composants du compresseur. Cela nous a permis d'approfondir nos connaissances techniques et de développer des compétences spécifiques dans le domaine de la compression.

Par la suite, nous avons programmé notre système conformément aux spécifications détaillées du cahier des charges, en expliquant son fonctionnement. En fin de compte, nous avons supervisé notre système à l'aide du logiciel WinCC afin de suivre en temps réel l'état d'avancement du processus et de pouvoir intervenir directement sur le pupitre de commande depuis la salle de contrôle.

En perspective, il serait intéressant d'effectuer une étude approfondie de la conformité du compresseur Siad tempo2-950 aux normes et réglementations industrielles en vigueur. Cela pourrait inclure des normes de sécurité, des exigences environnementales, des directives de qualité, développement des compresseurs plus silencieux pour réduire le bruit. L'utilisation de technologies de régulation avancées pour un contrôle précis de la pression et l'intégration de fonctionnalités de connectivité pour une surveillance à distance.

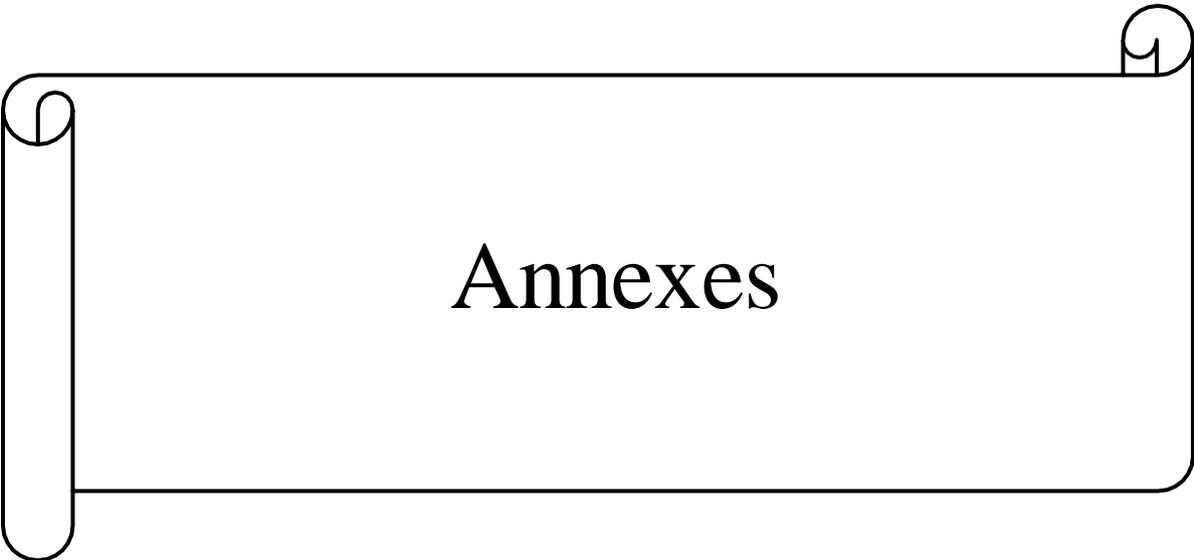
A decorative scroll-like frame with a black outline. The frame is horizontal and has a rounded right side. The top and bottom edges are slightly curved, and the left edge has a small circular detail at the top, suggesting a scroll binding. The word "Bibliographie" is centered within the frame in a black serif font.

Bibliographie

Bibliographie

- [1] Documentation interne de l'entreprise LABELLE (CO.G.B), catalogues et manuels d'utilisation.
- [2] B. BARKAT, L'ÉNERGIE PNEUMATIQUE, document. Université de Batna, 2020 disponible sur :
http://staff.univ-batna2.dz/sites/default/files/barkat-belkacem/files/07_complement_cours_lenergie_pneumatique.pdf, consulté le 15 mars 2023.
- [3] M. DJARRALLAH, INTRODUCTION AUX SYSTEMES AUTOMATISÉS, document. Université de Batna, 2017 disponible sur :
http://staff.univ-batna2.dz/sites/default/files/djarrallah_mohamed/files/chapitre_3_auto-mate_programmable_api.pdf, consulté le 20 avril 2023.
- [4] M. BERTRAND, AUTOMATES PROGRAMMABLES INDUSTRIELS, document, 2010 disponible sur :
<https://www.techniques-ingenieur.fr/base-documentaire/automatique-robotique-th16/automatique-sequentielle-42395210/automates-programmables-industriels-s8015/>, consulté le 21 mars 2023.
- [5] A. GONZAGA, LES AUTOMATES PROGRAMMABLES INDUSTRIELS, document. 2004 disponible sur :
https://www.geea.org/IMG/pdf/LES_AUTOMATES_PROGRAMMABLES_INDUSTRIELS_pour_GEEA.pdf, consulté le 22 mars 2023.
- [6] commentouvrir.com/info/caracteristiques-des-logiciels-comprendre-les-elements-constitutifs-des-programmes-informatiques/
- [7] 6ES7214-1BG31-0XB0 | Siemens Simatic S7-1200 - CPU 1214C | PLC-City, (n.d.) disponible sur :
<https://www.plc-city.com/shop/fr/siemens-simatic-s7-1200-cpu-1214c/6es7214-1bg31-0xb0.html>, consulté le 20 mars 2023.
- [8] AUTOMATE PROGRAMMABLE, document, 2014, disponible sur :
https://media.automation24.com/manual/fr/91696622_s71200_System_Manual_fr-FR_fr-FR.pdf, consulté le 21 mars 2023.

- [9] Libération de la famille des automates SIMATIC S7-1200 - ID: 36524193 - Industry Support Siemens, (n.d.), 2014, disponible sur :
<https://support.industry.siemens.com/cs/document/36524193/lib%C3%A9ration-de-la-famille-des-automates-simatic-s7-1200?dti=0&lc=fr-DZ>, consulté le 21 mars 2023.
- [10] s71200_System_Manual_fr-FR_fr-FR.pdf - SIMATIC S7 Automate programmable S7-1200 - ID: 91696622 - Industry Support Siemens, 2014, disponible sur :
<https://support.industry.siemens.com/cs/document/91696622/simatic-s7-automate-programmable-s7-1200?dti=0&lc=fr-CH>, consulté le 27 mars 2023.
- [11] http://www.univ_bejaia.dz/xmlui/bitstream/handle/123456789/8538/Etude%20et%20supervision%20de%20deux%20bacs%20d%27huile%20brute%20Cevital-Béjaia.pdf?sequence=1&isAllowed=y, consulté le 24 mars 2023.
- [12] M. BENALLEL, SUPERVISION DES SYSTEMES INDUSTRIELS, document, université de Tlemcen, disponible sur :
https://elearn.univ-tlemcen.dz/pluginfile.php/106806/mod_resource/content/1/chapitre1.pdf, consulté le 27 mars 2023
- [13] D. Nanping, « Application of S7-400 and WinCC in Dense Phase System », article, College of Automation of Beijing Union University Beijing / China, Troisième symposium international sur les ateliers d'application des technologies de l'information intelligentes.
- [14] SUPERVISION, document, 2017, disponible sur :
<https://m.20-bal.com/pravo/8796/index.html?page=8>, consulté le 2 avril 2023.
- [15] <https://www.copadata.com/fr/produits/zenon-software-platform/visualisation-contrôle/quest-ce-qu-une-ihm-interface-homme-machine-copa-data/>, consulté le 5 avril 2023.

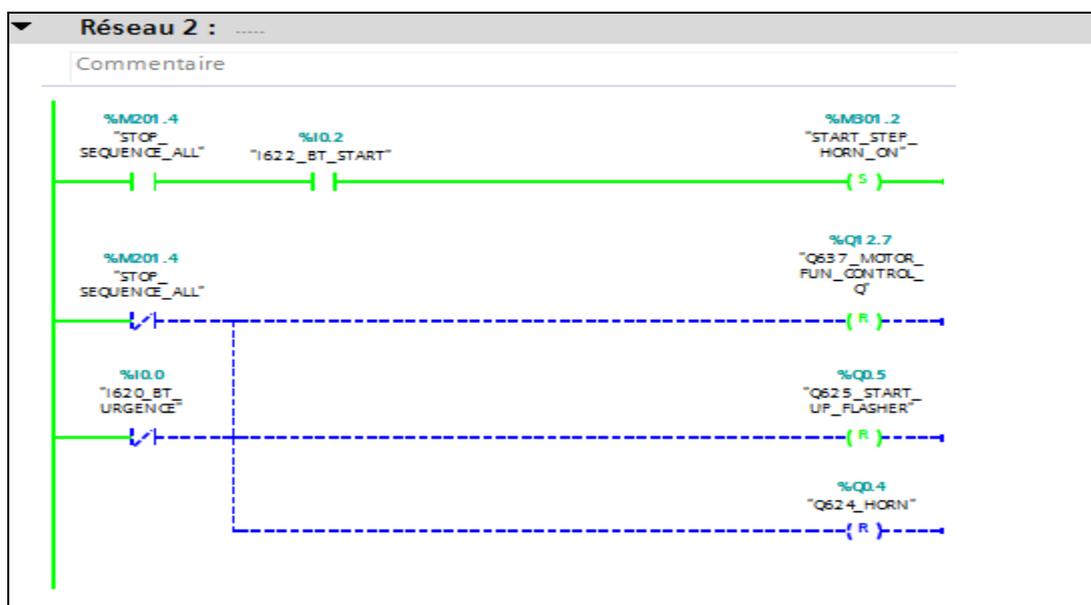
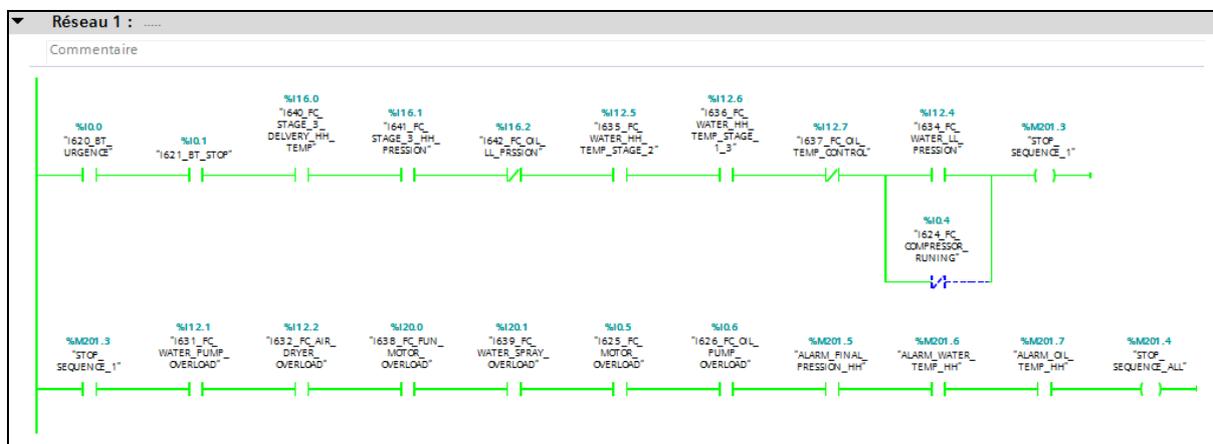


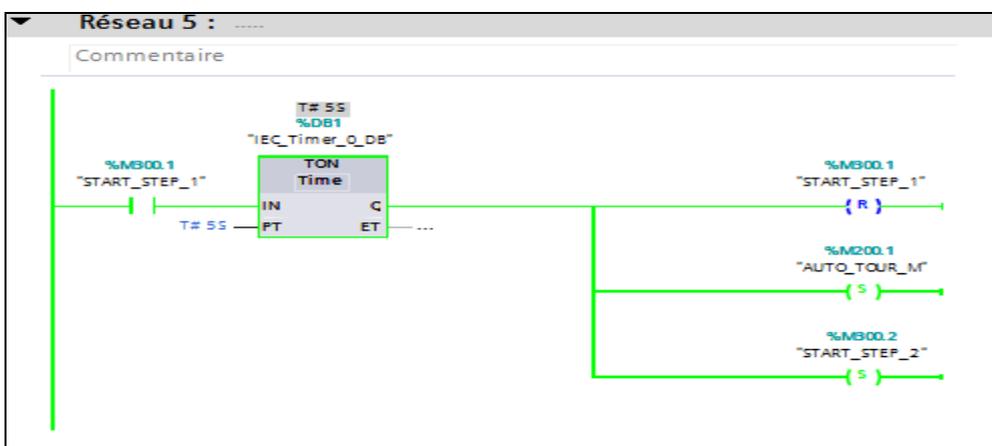
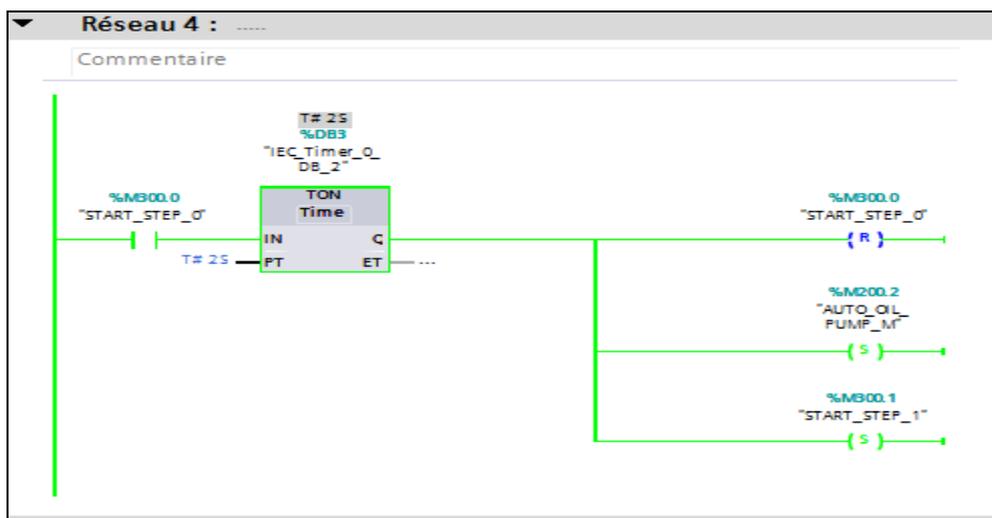
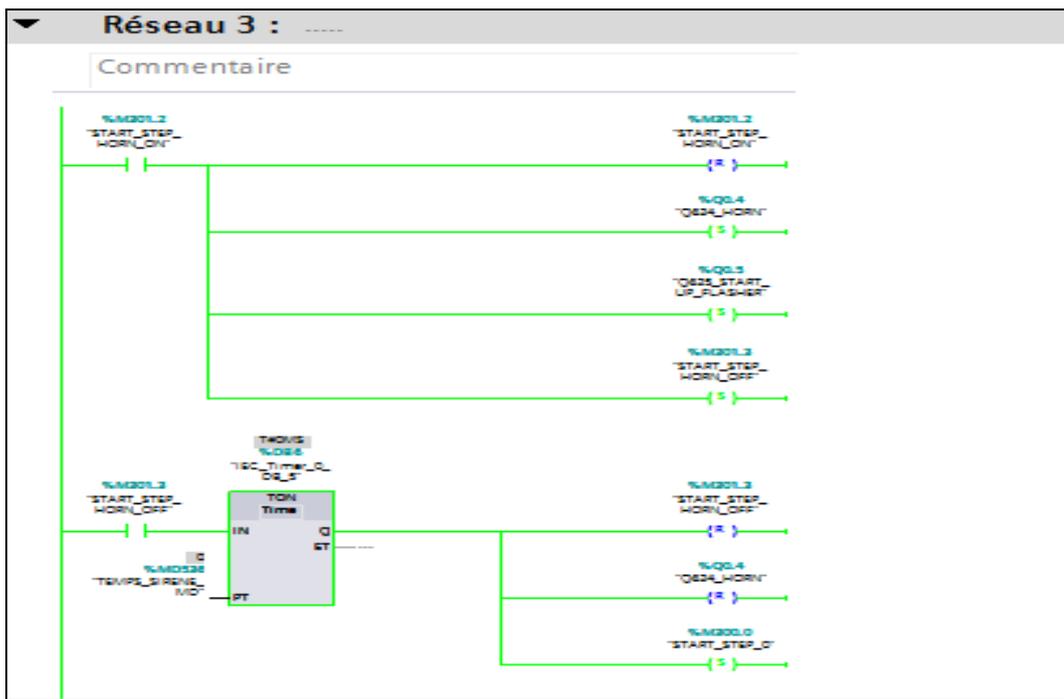
Annexes

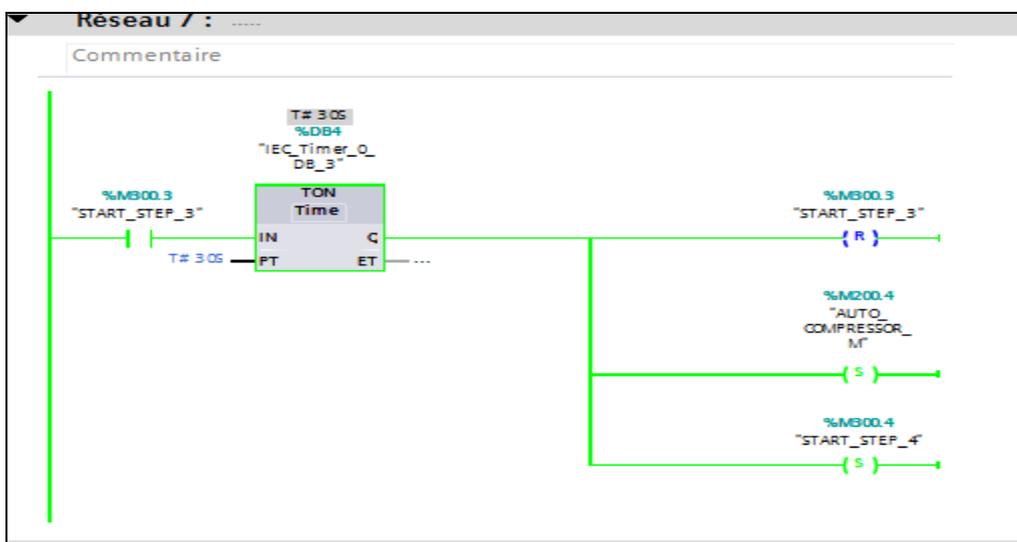
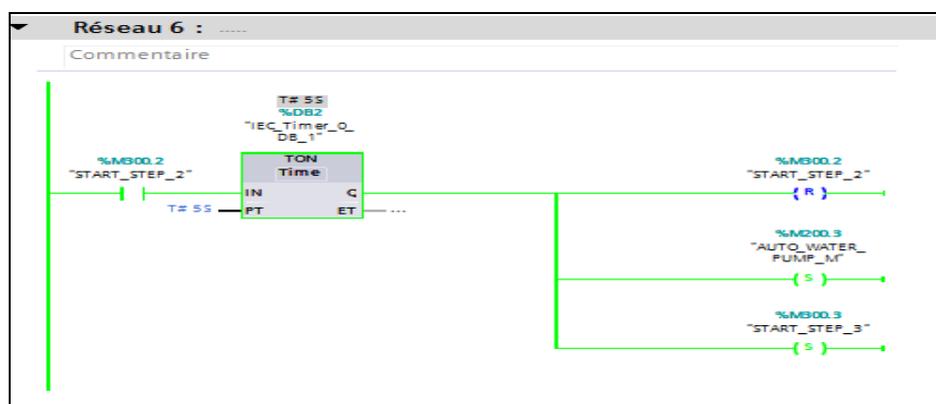
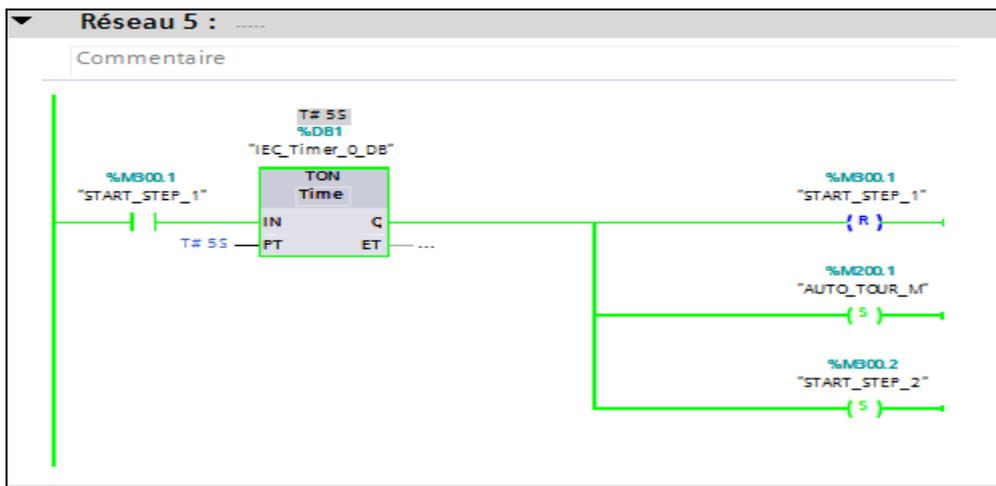
Annexes

Annexe A :

Simulation en Mode Automatique :

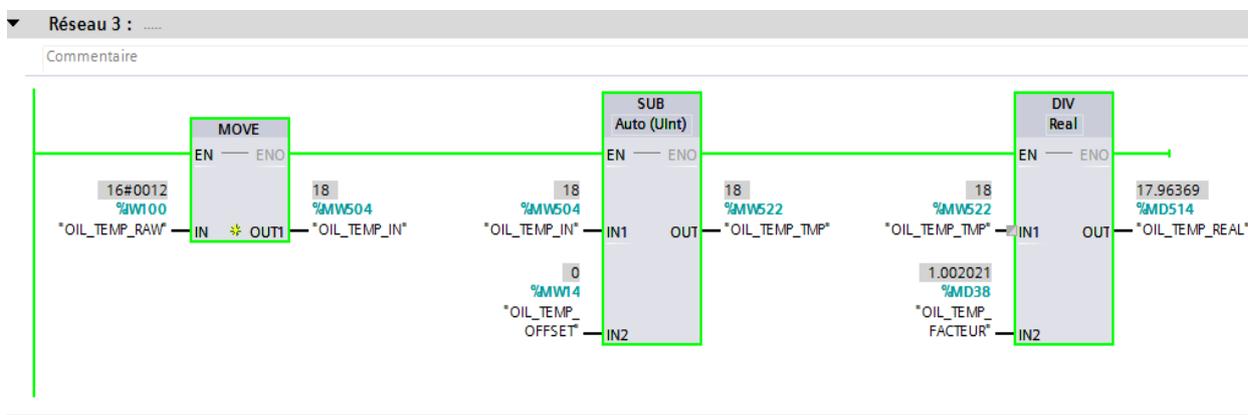
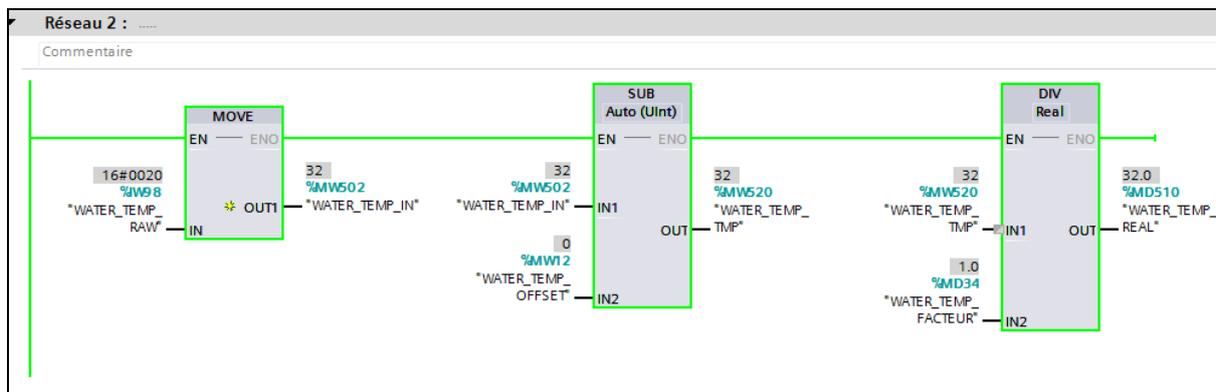






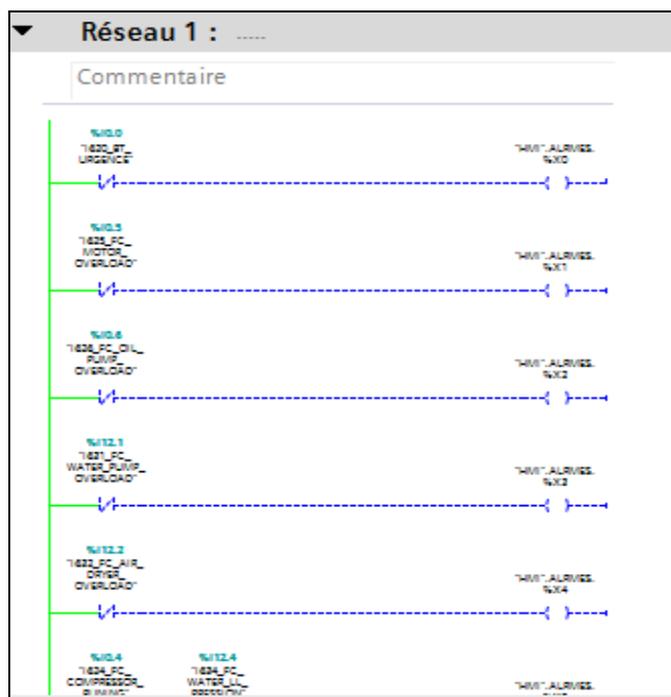
Annexe B :

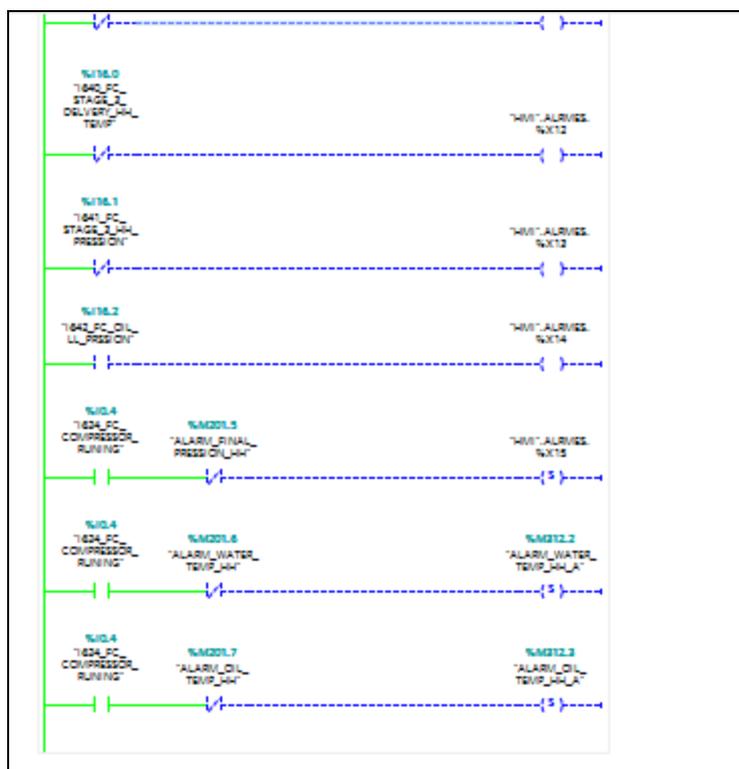
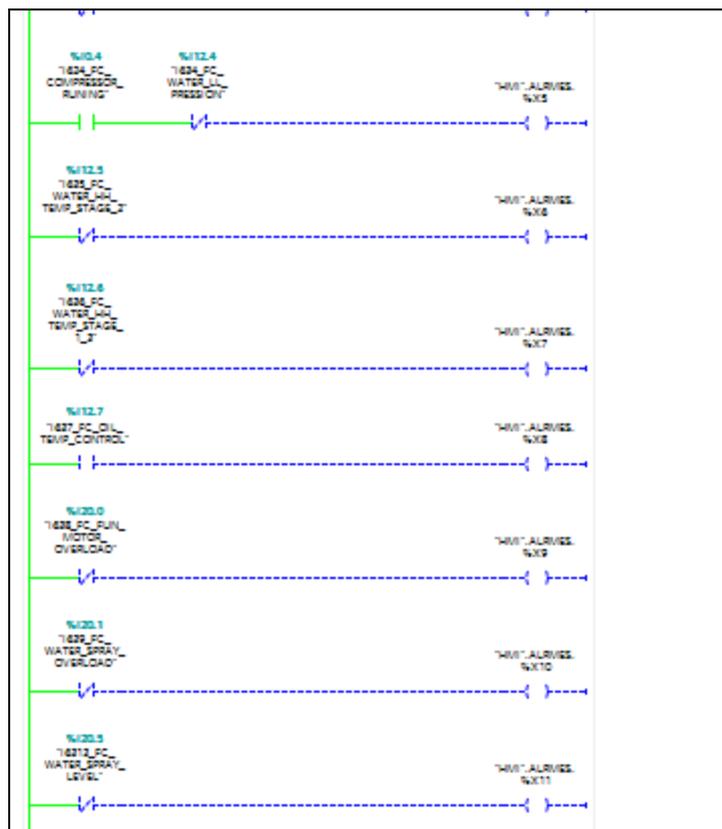
Simulation des blocs analogiques



Annexe C :

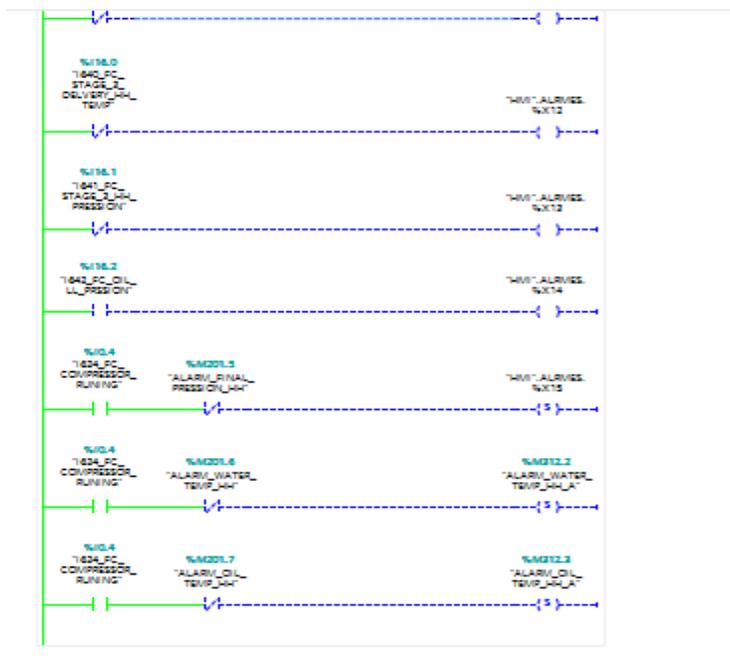
Simulation du bloc des groupes d'alarmes





Annexe D :

Simulation du bloc ack alarme



Annexe E :

Table des variables HMI

Alarmes de bit							
ID	Nom	Texte d'alarme	Classe d'alar...	Variable de d...	Bit de ..	Adresse de dé..	
1	Alarme de bit_1	Arret d'urgence enclenché	Errors	HMI_ALR...	0	HMI.ALRMES.x0	
2	Alarme de bit_2	ALARME COMPRESSOR	Errors	HMI_ALRMES	1	HMI.ALRMES.x1	
3	Alarme de bit_3	ALARME_QF_OIL_PUMP	Errors	HMI_ALRMES	2	HMI.ALRMES.x2	
4	Alarme de bit_4	ALARME_QF_WATER_PUMP	Errors	HMI_ALRMES	3	HMI.ALRMES.x3	
5	Alarme de bit_5	ALARME_AIR_DRYER	Errors	HMI_ALRMES	4	HMI.ALRMES.x4	
6	Alarme de bit_6	ALARME_WATER_PRESSION_LL	Errors	HMI_ALRMES	5	HMI.ALRMES.x5	
7	Alarme de bit_7	ALARME_HH_TEMP_STAGE_2	Errors	HMI_ALRMES	6	HMI.ALRMES.x6	
8	Alarme de bit_8	ALARME_HH_TEMP_STAGE_1_3	Errors	HMI_ALRMES	7	HMI.ALRMES.x7	
9	Alarme de bit_9	ALARME_OIL_TEMP_CONTROLLER	Errors	HMI_ALRMES	8	HMI.ALRMES.x8	
10	Alarme de bit_10	ALARME_FC_FUN_CONTROL_OVERLO.	Errors	HMI_ALRMES	9	HMI.ALRMES.x9	
11	Alarme de bit_11	ALARME_WATER_SPRAY_OVERLOAD	Errors	HMI_ALRMES	10	HMI.ALRMES....	
12	Alarme de bit_12	ALARME_WATER_LEVEL_LL	Errors	HMI_ALRMES	11	HMI.ALRMES....	
13	Alarme de bit_13	ALARME_TEMP_HH_STAGE_3	Errors	HMI_ALRMES	12	HMI.ALRMES....	
14	Alarme de bit_14	ALARME_PRESSION_STAGE_3	Errors	HMI_ALRMES	13	HMI.ALRMES....	
15	Alarme de bit_15	ALARME_FC_OIL_LL_PRESSION	Errors	HMI_ALRMES	14	HMI.ALRMES....	

Étude et automatisation du compresseur d'air SIAD 5lpet

Résumé : Ce manuscrit présente une approche générale pour automatiser un système industriel, en mettant l'accent sur l'étude détaillée d'un compresseur SIAD utilisé pour le soufflage de bouteilles en PET de 5L. La modélisation du fonctionnement du compresseur a été effectuée à l'aide du logiciel STEP7, permettant ainsi à l'automate S7-1200 de gérer automatiquement le processus. Le manuscrit comprend également une description détaillée des automates programmables industriels, en mettant particulièrement l'accent sur le S7-1200 de la société SIEMENS. Une part importante du travail est consacrée à la présentation du logiciel STEP7, décrivant les étapes essentielles pour la création d'un projet d'automatisation, la configuration matérielle, le développement du programme et sa simulation. De plus, une interface de supervision a été développée, permettant le contrôle à distance de l'installation à l'aide de WinCC Flexible.

Mots-clés : Automatisation, compresseur SIAD, automate programmable industriel, corps gras de Bejaia LABELLE, supervision.

Study and Automation of the SIAD 5lpet Air Compressor

Abstract : This manuscript presents a general approach to automate an industrial system, focusing on the detailed study of an SIAD compressor used for blowing 5L PET bottles. The operation of the compressor was modeled using the STEP7 software, enabling the S7-1200 programmable logic controller to automatically manage the process. The manuscript also includes a detailed description of industrial programmable logic controllers, with special emphasis on the SIEMENS S7-1200. A significant portion of the work is dedicated to the presentation of the STEP7 software, outlining the essential steps for creating an automation project, hardware configuration, program development, and simulation. Additionally, a supervision interface was developed to enable remote control of the installation using WinCC Flexible.

Keywords: Automation, SIAD compressor, industrial programmable logic controller, Bejaia LABELLE edible oils, supervision.