



الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي والبحث العلمي
Ministère de l'Enseignement Supérieure et de la Recherche Scientifique
جامعة وهران 2 محمد بن أحمد
Université d'Oran 2 Mohamed Ben Ahmed
معهد الصيانة والأمن الصناعي
Institut de Maintenance et de Sécurité Industrielle

Département de sécurité industrielle et environnement

MÉMOIRE

Pour l'obtention du diplôme de Master

Filière : Génie industrielle

Spécialité : Maintenance des Automatismes et de l'instrumentation Industriels (MAII)

Thème

Automatisation de la régulation de température de PVA

Présenté et soutenu publiquement par :

Nom

HAMDOUCHE

BELKACEM BENOUNANE

Prénom

KHEIRA

CHAHRAZED

Devant le jury composé de :

| Nom et Prénom | Grade | Etablissement | Qualité |
|-----------------|-------|------------------|-----------|
| BELKHODJA Leila | MCD | Université oran2 | Président |
| AISSANI Nassima | MCA | Université oran2 | Encadreur |
| BENFKIR . A. | MCD | Université oran2 | Examineur |

Année 2023/2024

Dédicace

J'ai le grand honneur de dédier ce modeste travail

A ma mère et mon père qui m'ont soutenu et aidé tout le long de mon parcours, sans eux je ne serais certainement pas devenu ce que je suis maintenant.

Mon binôme KHEIRA pour ces quatre années de travail pleines de souvenirs.

MON MEILLEUR AMI AMINE Ça a eu un grand impact sur mon moral.

CHAHRAZED

Dédicace

Je dédie ce modeste travail

A mes très chers parents qui m'ont soutenu dans mes études

Mon binôme CHAHRAZED

KHEIRA

Remerciements

En préambule à ce mémoire nous remercions ALLAH de nous avoir donné la patience, le courage et la force d'accomplir ce travail.

Nous tenons à remercier en premier lieu Madame AISSANI NASSIMA d'avoir accepté d'être mon encadreur durant de ce travail, et pour la confiance qu'elle m'a donnée et ses précieux conseils.

Nous remercions très chaleureusement Mr BEN BELLA WALID, Mr SNOUCI HASNI et Mr Khalil, pour leur aide précieuse qu'il nous a apportée tout au long de ce travail.

Nos vifs remerciements vont également aux membres du jury pour l'intérêt qu'ils ont porté à notre recherche en acceptant d'examiner notre travail et de l'enrichir par leurs propositions.

On remercie nos familles pour leur soutien et leurs encouragements.

Enfin, nous tenons également à remercier toutes les personnes qui ont participé de près ou de loin à la réalisation de ce travail.

SOMMAIRE

Introduction General

Chapitre 01 : Présentation de l'entreprise et la Régulation

| | | |
|----------|---|----|
| I. | Introduction..... | 2 |
| II. | Présentation de l'entreprise | 2 |
| II.1 | L'Entreprise Nationale des Peintures (ENAP)..... | 2 |
| II.2 | Spécialité de l'ENAP..... | 2 |
| II.3 | Organisme de l'unité SIG..... | 3 |
| III. | Peinture et Fabrication polyvinyle acrylique | 3 |
| III.1 | La peinture | 3 |
| III.1.1 | Définition de la peinture..... | 3 |
| III.1.2 | Nature de la peinture | 4 |
| III.1.3 | Composition d'une peinture..... | 4 |
| III.1.4 | Les deux types de peinture principaux..... | 5 |
| III.2 | Fabrication polyvinyle acrylique | 5 |
| III.2.1 | Les matières premières | 5 |
| III.2.2 | Procédé de fabrication du poly acétate de vinyle | 6 |
| III.2.3 | Mode de fonctionnement de processus de fabrication PVA..... | 7 |
| III.3 | Description des instruments du processus..... | 14 |
| III.3.1 | Le régulateur de température PID..... | 15 |
| III.3.2 | Vanne pneumatique | 15 |
| III.3.3 | Les pompes | 15 |
| III.3.4 | Capteur de température..... | 16 |
| III.3.5 | La sonde PT100..... | 16 |
| III.3.6 | Les débitmètres | 16 |
| III.3.7 | Soupape de sûreté..... | 16 |
| III.3.8 | Réacteur..... | 17 |
| III.3.9 | Moteur | 17 |
| III.3.10 | Un manomètre | 17 |
| IV. | La régulation | 18 |
| IV.1 | Introduction | 18 |

| | | |
|----------|--|----|
| IV.2 | Définition de la régulation industrielle | 19 |
| IV.2.1 | Régulation manuelle et régulation automatique | 19 |
| IV.2.2 | La boucle de régulation | 19 |
| IV.3 | Régulateur industriel..... | 20 |
| IV.3.1 | Architecture d'un régulateur | 20 |
| IV.3.2 | Sens d'action d'un régulateur | 21 |
| IV.3.3 | Le sens de régulateur utilisé | 22 |
| IV.3.4 | Choix du sens d'action de régulation | 22 |
| IV.3.5 | Définition de la régulation PID | 22 |
| IV.3.6 | Les actions proportionnelle /intégrale /derivé..... | 23 |
| IV.4 | Les instruments de régulation | 24 |
| IV.4.1 | La vanne..... | 24 |
| IV.4.1.1 | Eléments d'une vanne | 24 |
| IV.4.2 | Les capteurs | 26 |
| V. | Régulation et identification d'un système..... | 28 |
| V.1 | Introduction | 28 |
| V.2 | Les étapes d'identification d'un système..... | 28 |
| 1. | Collecte de données..... | 28 |
| 2. | Prétraitement des données | 28 |
| 3. | Choix du modèle | 28 |
| 4. | Estimation des paramètres | 28 |
| 5. | Validation du modèle..... | 28 |
| V.3 | Simulation Matlab..... | 28 |
| V.4.1 | Identification..... | 29 |
| V.4.2 | Régulation..... | 30 |
| VI. | Conclusion..... | 33 |
| I. | Introduction..... | 35 |
| III. | Définition de système automatisé..... | 35 |
| I.1 | Objectif de l'automatisation..... | 35 |
| I.2 | Avantages et les Inconvénients de l'automatisation | 35 |
| I.3 | Domaines d'application des systèmes automatisés..... | 36 |
| I.4 | Structure d'un système automatisé | 36 |
| VI.1 | | 36 |
| VI.2 | | 36 |

| | |
|--|----|
| VI.3 | 36 |
| VI.4 | 36 |
| II.4.1 Partie opérative | 36 |
| II.4.2 La partie commande P.C..... | 37 |
| II.4.3 La partie pupitre ou la partie supervision | 38 |
| III. Les automates Programmables Industriels..... | 39 |
| III.1 Historique Les Automates Programmables Industriels | 39 |
| III.2 Définition un automate programmable (API) | 39 |
| III.3 Principe générale de fonctionnement d'un API..... | 39 |
| III.4 Architecture d'un API..... | 40 |
| III.4.1 Aspect extérieur | 40 |
| III.4.2 Structure interne | 41 |
| III.5 Présentation de quelques gammes SIMATIC..... | 43 |
| III.6 Description de l'Automate S7-1200 | 45 |
| III.6.1 Choix de la CPU S7-1200..... | 46 |
| III.6.2 Le choix des modules d'Entrées/Sorties | 46 |
| III.7 Nature des informations traitées par l'automate..... | 46 |
| III.8 Les notions de choix d'un automate programmable (API) | 46 |
| III.9 Domaines d'emploi des automates | 47 |
| III.10 La communication et les Réseaux industriels | 47 |
| III.10.1 Le Profibus..... | 47 |
| III.10.2 Profinet..... | 47 |
| III.11 Les avantages et les inconvénients..... | 47 |
| III.11.1 Avantage des API..... | 47 |
| III.11.2 Inconvénients des API | 48 |
| III.12 Les langages de programmation d'un API | 48 |
| III.13 Principaux constructeurs d'API | 49 |
| IV. L'Environnement de developpement « TIA Portal V16 » | 50 |
| VI.1 Présentation du logiciel | 50 |
| VI.2 Les avantages du logiciel..... | 50 |
| VI.3 STEP 7 sur TIA Portal | 50 |
| VI.4 S7-PLCSIM V16 | 50 |
| VI.5 Logiciel de supervision WinCC | 51 |
| VI.5.1 WinCC runtime advanced V16..... | 52 |

| | | |
|---------|---|----|
| VI.6 | La relation entre TIA portal et Wincc..... | 52 |
| VI.7 | La supervision industrielle | 52 |
| VI.7.1 | Rôle de la supervision..... | 53 |
| VI.7.2 | Définition de SCADA | 53 |
| VI.7.3 | Définition IHM | 53 |
| VI.7.4 | Choix de l'interface homme-machine | 54 |
| VI.8 | Conception d'un programme avec TIA PORTAL V13 : | 54 |
| VI.8.1 | Vue du portail et vue du projet | 55 |
| VI.8.2 | Création d'un projet | 57 |
| VI.9 | Configuration et paramétrage du matériel..... | 58 |
| VI.10 | Adressage des E/S | 59 |
| VI.11 | Compilation et chargement de la configuration matérielle | 60 |
| VI.12 | Les blocs de programme..... | 62 |
| VI.12.1 | Les blocs d'organisation – OB..... | 63 |
| ii. | Les fonctions – FC..... | 67 |
| iii. | Les blocs fonctionnels – FB | 67 |
| iv. | Les blocs de données « DB » | 67 |
| V. | Conclusion | 67 |
| I. | Introduction..... | 69 |
| II. | La hiérarchie du programme..... | 69 |
| III. | Editeur de mnémoniques..... | 70 |
| IV. | Simulation de programme de système de contrôle de température | 70 |
| IV.1 | Chargement de programme | 71 |
| IV.2 | Démarrage de WinCC Runtime | 71 |
| IV.3 | Visualisation de l'état du programme..... | 72 |
| V. | Les blocs de programme..... | 72 |
| V.1 | OB Cyclique interrupt (OB30) | 72 |
| V.2 | Programme du Bloc d'Organisation OB1 | 73 |
| V.3 | Le bloc de fonction fc4..... | 74 |
| VI. | Programmations des alarmes | 77 |
| VI.1 | Programmation au niveau de l'automate..... | 77 |
| VI.2 | Configuration au niveau de l'IHM | 78 |
| VII. | La Vue Globale | 79 |
| VII.1 | Animation de la vanne de régulation..... | 79 |

| | | |
|---------|---|----|
| VII.2 | Création de bloc d'affichage (pop-up)..... | 80 |
| VII.3 | Animation de bouton d'état de vanne..... | 80 |
| VII.4 | Animation du bouton arrêt..... | 81 |
| VII.5 | Exemple sur le fonctionnement de PID Compact..... | 82 |
| VII.5.1 | La visualisation des courbes..... | 82 |
| VII.6 | La vue du process..... | 83 |
| VII.6.1 | Exemple vue alarme..... | 83 |
| VII.7 | Sécurité..... | 84 |
| VI. | conclusion..... | 86 |

LISTE ABRÉVIATIONS

ENAP : Entreprise Nationale des Peintures.

PVA : polyvinyle acrylique.

API : Automate programmable industriel.

CP : Processeurs de communication.

CPU : Central Processing Unit.

PC : Partie opérative.

PO : Partie commande.

PI : Partie interface.

E/S : entrées / sorties.

GRAFSET : Graphe Fonctionnel de Commande Etape Transition.

WinCC : Windows Contrôle Center.

HMI : Interface Homme Machine.

PLC : Programmable Logic Controller.

Profibus : process Field bus.

SA : Système Automatisé.

SCADA : Supervisory Control And Data Acquisition.

TIA : Totally Integrated Automation.

TOR : Tout ou rien.

OB : bloc d'organisation.

FB : Bloc de fonction.

FC : Fonction.

DB : Bloc de données.

SCL : Structured Control Language.

LIST : Le langage de liste d'instructions.

LOG : Le langage à base de Logigramme.

IL : Instruction List.

LD : Ladder Diagram.

SFC : Programme de chaîne séquentielle.

Auto : Automatique.

MAN : Manuel.

LISTE DES FIGURES

| | |
|---|----|
| Figure 1: Produits ENAP | 13 |
| Figure 2: Situation Géographique | 13 |
| Figure 3: ENAP SIG | 13 |
| Figure 4: Processus de Fabrication de "PVA" | 18 |
| Figure 5: Le refroidisseur YORK YAES | 19 |
| Figure 6:Le fonctionnement de refroidisseur YORK YAES | 24 |
| Figure 7: Le régulateur de température "PID" | 25 |
| Figure 8: Vanne pneumatique | 25 |
| Figure 9: Pompe | 25 |
| Figure 10: La sonde "PT100" | 26 |
| Figure 11: Débitmètre | 26 |
| Figure 12: Soupape de sûreté | 26 |
| Figure 13: Réacteur | 27 |
| Figure 14: Moteur | 27 |
| Figure 15: Manomètre | 27 |
| Figure 16: Régulation de température d'un échangeur thermique | 28 |
| Figure 17: La boucle de régulation | 29 |
| Figure 18: constitution d'un régulateur | 30 |
| Figure 19: Régulateur sens inverse | 31 |
| Figure 20: Régulateur sens direct | 31 |
| Figure 21: Action Proportionnel | 33 |
| Figure 22: Vanne manuel | 34 |
| Figure 23: Vanne automatique | 35 |
| Figure 24: Exemple des capteurs | 36 |
| Figure 25: La sonde "PT100" | 37 |
| Figure 26: Chargement des données Dans "MATLAB" | 39 |
| Figure 27: Prétraitement des données | 39 |
| Figure 28: Choix De modèle | 39 |
| Figure 29: Estimations des Paramètres | 40 |
| Figure 30: Validation Du modèle | 40 |
| Figure 31: Modélisation du système | 40 |
| Figure 32: Simulation du système | 41 |
| Figure 33: Réglage des Paramètres | 41 |
| Figure 34: Contrôleur "PID" | 42 |
| Figure 35: surveillance et ajustements continus | 42 |
| Figure 36: Facilitation De l'accord du "PID" | 43 |
| Figure 37: Exemple d'un système automatisé | 44 |
| Figure 38: Structure d'un système automatisé | 45 |
| Figure 39: fonctionnement d'un API | 49 |
| Figure 40: Architecture d'un API | 49 |
| Figure 41: API compacte | 50 |
| Figure 42: API modulaire | 50 |

| | |
|---|----|
| Figure 43: Structure interne de l'API | 50 |
| Figure 44: S7-200 | 52 |
| Figure 45: S7-300 | 52 |
| Figure 46: S7-400 | 53 |
| Figure 47: S7-1200 | 53 |
| Figure 48: S7-1500 | 54 |
| Figure 49: Automate S7-1200- Siemens | 55 |
| Figure 50: Langage FBD | 57 |
| Figure 51: Langage GRAFCET | 58 |
| Figure 52: Langage à Contact LD | 58 |
| Figure 53: Fenêtre du travail dans le WinCC | 60 |
| Figure 54: Système SCADA | 62 |
| Figure 55: IHM | 63 |
| Figure 56: La vue de Projet | 64 |
| Figure 57: Vue du portail | 64 |
| Figure 58: Vue du portail | 65 |
| Figure 59: Vue du portail | 65 |
| Figure 60: Création d'un projet | 66 |
| Figure 61: Configuration et paramétrage du matériel | 67 |
| Figure 62: Configuration et paramétrage du matériel | 68 |
| Figure 63: Adressage des E/S | 68 |
| Figure 64: Compilation de la configuration matérielle | 69 |
| Figure 65: chargement de la configuration matérielle | 70 |
| Figure 66: Vérification avant chargement | 70 |
| Figure 67: Chargement est terminé | 71 |
| Figure 68: les blocs de programme | 71 |
| Figure 69: création un bloc de programme | 71 |
| Figure 70: Les OB d'alarmes temporisées | 73 |
| Figure 71: Diagramme de fonctionnement des OB | 75 |
| Figure 72: Hiérarchie du programme | 79 |
| Figure 73: Table de variables | 80 |
| Figure 74: Démarrage de simulation | 80 |
| Figure 75: Table de simulation | 81 |
| Figure 76: Chargement de programme | 81 |
| Figure 77: Visualisation de l'état du programme | 82 |
| Figure 78: Paramètre de PID | 83 |
| Figure 79: OB30 PID-Compact de température | 83 |
| Figure 80: Blocs d'organisation main OB | 84 |
| Figure 81: Réseau pour commander le régulateur | 84 |
| Figure 82: FC vanne de régulation | 85 |
| Figure 83: Les réseaux d'ordre fermeture et ouverture automatique | 86 |
| Figure 84: Commande de vanne en mode local, manuel et automatique | 86 |
| Figure 85: Commande de vanne pour fermeture en mode local, manuel et automatique | 87 |
| Figure 86: Réseau pour blocage alarme d'ouverture | 88 |
| Figure 87: Réseau pour alarme d'ouverture | 88 |

| | |
|--|----|
| Figure 88: Fenêtre des alarmes | 89 |
| Figure 89: La Vue Globale | 89 |
| Figure 90: Animation de la vanne de régulation | 90 |
| Figure 91: Création d'une Faceplate pour la vanne | 90 |
| Figure 92: Animation de bouton d'état de vanne | 91 |
| Figure 93: Animation de bouton arrêt | 91 |
| Figure 94: La visualisation des courbes | 92 |
| Figure 95: Simulation d'HMI | 93 |
| Figure 96: Exemple d'alarme | 93 |
| Figure 97:Création de mot de passe | 94 |
| Figure 98: création de mot de passe | 94 |
| Figure 99: Création de mot de passe | 95 |
| Figure 100: Exemple de création de mot de passe | 95 |

Introduction générale

Introduction générale

L'industrie des peintures et vernis est un modèle d'une de ces industries de transformation où le rôle est devenu très important dans la vie moderne, donc il est nécessaire d'intervenir pour la résolution des problèmes pratiques.

Depuis des décennies, l'automatisation a joué un rôle important dans la production de peinture, permettant d'obtenir des résultats et de mieux rivaliser sur le marché. Cela permet également de fournir de meilleures solutions aux besoins des clients et de livrer des produits de meilleure qualité dans des délais plus courts. L'automatisation permet également de personnaliser les produits et les fonctionnalités selon les besoins de chaque entreprise et client. De plus, grâce à l'automatisation, il est possible d'effectuer des tâches qui ne peuvent pas être réalisées manuellement dans ce domaine.

Notre projet de travail va dans cette direction. Notre objectif est de mettre en œuvre et de contrôler la station de fabrication de PVA tout en atteignant une production élevée avec moins de pertes matérielles tout en améliorant les conditions de travail du personnel.

Pour cela, nous avons utilisé le logiciel TIA Portal de Siemens. L'environnement comprend des simulateurs SIEMENS tels que le S7-1200. Nous avons utilisé l'écran HMI pour le fonctionnement.

Nous exposons dans le présent travail trois grands chapitres décrivant les volets principaux de notre projet de fin d'études :

Dans le premier chapitre, nous présenterons la société de peinture ENAP de sig à Mascara et la définition de peinture, procédé de fabrication de polyvinyle acrylique et le système de refroidissement. Nous présenterons aussi des généralités sur la régulation et leur identification.

Le deuxième chapitre est consacré à la structure et définition de système automatisé, les avantages et les critères de choix d'un API et initialisation sur logiciel de programmation TIA Portal.

Le dernier chapitre englobe tous les travaux d'applications pour l'automatisation d'une station de fabrication de PVA en utilisant le logiciel TIA Portal V16.

Enfin, une conclusion générale clôturera notre mémoire.

Chapitre 01 :

Présentation de

l'entreprise et la

Régulation

I. Introduction

Dans ce chapitre nous présenterons les types de peintures et principalement la peinture à eau, ses composants et son processus de fabrication et généralités sur la régulation et identification.

II. Présentation de l'entreprise

II.1 L'Entreprise Nationale des Peintures (ENAP)

ENAP est une Entreprise Publique Économique qui a pour métier de base la production des revêtements organiques (peintures, vernis, résines, émulsions Siccateurs et colles).

L'ENAP est composée de six (06) unités de production et deux (02) unités commerciales implantées sur le territoire national :

Deux (02) unités de production à la wilaya d'Alger (Oued-Smar et Cheraga).

Une (01) unité de production à la wilaya de Bouira (Lakhdaria).

Une (01) unité de production à la wilaya d'Oran (Oran).

Une (01) unité de production à la wilaya de Mascara (Sig).

Une (01) unité de production à la wilaya de Souk -Ahras (Souk-Ahras).

Une (01) unité commerciale à la wilaya de Sétif (Sétif) [14].

II.2 Spécialité de l'ENAP

Grâce à sa capacité de production l'ENAP couvre les besoins du marché national par ses différents produits qui sont destinés aux divers secteurs d'activités.

Peinture pour bâtiment.

Peinture pour carrosserie.

Peintures et vernis pour bois.

Peintures anticorrosion [14].



Figure 1: Produits ENAP

II.3 Organisme de l'unité SIG

L'ENAP-SIG –S'étale sur 13 hectares dans la zone industrielle a deux kilomètres du centre-ville de sig. Nous nous intéressons à l'unité ENAP de sig MASCARA de peinture qui fait la production des revêtements organiques et plus spécialement les peintures.



Figure 2: Situation Géographique



Figure 3: ENAP SIG

III. Peinture et Fabrication polyvinyle acrylique

III.1 La peinture

III.1.1 Définition de la peinture

Une peinture est une composition liquide ou poudre, colorée et opacifiante qui a la propriété principale de passer de l'état liquide à l'état solide dans un certain laps de temps. La pellicule posée épouse la forme du support, y adhère fortement et contribue à la décoration de ce dernier [37].

III.1.2 Nature de la peinture

La classification des peintures et vernis selon la norme NFT 30-0003 est basée sur la nature chimique du liant et comporte 9 classes :

1. Peintures à l'eau.
2. Peinture aux huiles et vernis gras.
3. Semi-produit broyé pour peinture.
4. Alkydes.
5. Cellulosiques.
6. Polyesters et polyéthers.
7. Vinyliques acryliques et copolymères.
8. Dérivés du caoutchouc et élastomères.
9. Autres liants [37].

III.1.3 Composition d'une peinture

La peinture est composée essentiellement de liants, de pigments, de solvants, de charges et d'additifs [37].

Les liants

Les liants sont des substances macromoléculaires naturelles ou synthétiques solides ou liquides solubles dans de nombreux solvants organiques. Ils permettent l'étalement de la peinture en film continu et favorisent la liaison entre les pigments [37].

Les pigments

Un pigment de peinture est sous forme d'une poudre de fines particules diffuse la lumière du spectre visible ce qui donne l'effet de blanc, ou qui absorbe certaines longueurs d'ondes ce qui produit la couleur, ainsi le mélange des pigments détermine la couleur souhaitée de la peinture [37].

Les solvants

Ce sont des composés organiques liquides volatils qui servent à rendre la peinture assez fluide pour s'étaler facilement. Le solvant s'évapore lors de séchage de la peinture. Les solvants sont classés selon leur degré de volatilité en solvant lourd, léger, moyen [37].

Les additifs

Les additifs sont des substances qu'on introduit à faibles doses dans les peintures pour obtenir certaines propriétés qui sont : La fluidisation, L'épaississement, action conservatrice particulières.

A titre d'exemple nous avons

- Le siccatif.
- L'anti mousse.
- Les épaississants [37].

Les charges

Les charges généralement sont des substances minérales en poudre insolubles dans le milieu de suspension, utilisées pour des raisons économiques et techniques particulières. Elles se distinguent des pigments par leurs pouvoirs colorant et opacifiant faible [37].

III.1.4 Les deux types de peinture principaux

Les 2 types de peinture les plus communs sont

- . La peinture à l'eau .
- . La peinture à l'huile.

Dans les peintures en phase aqueuse, le solvant est l'eau. C'est pour cette raison que ces peintures ont peu d'odeur.

Les types de peinture à l'eau les plus communs sont les vinyliques, les acryliques et les alkydes, en fonction du liant utilisé.

Tandis que dans les peintures à l'huile (ou phase solvant), nous trouvons des solvants issus (en général) de la chimie du pétrole en quantités plus importantes. Ils sont appelés Composés Organiques Volatiles (COV).

La peinture à l'eau :

La peinture acrylique est une peinture à l'eau dont le liant est une résine acrylique (**polyvinyle acrylique PVA**). C'est une émulsion/dispersion de divers composants acryliques [37].

III.2 Fabrication polyvinyle acrylique

III.2.1 Les matières premières

Polyvinyle acétate (AVM)

L'acétate de vinyle à température ambiante est un liquide incolore fortement diélectrique miscible à la plupart des solvants organiques mais pas avec l'eau [36].

L'acrylate d'éthyl-2-hexyle

L'acrylate d'éthyl-2-hexyle c'est un liquide incolore de mauvaise odeur, très toxique, non volatil provoque facilement des réactions d'addition avec de nombreux composés

organiques et inorganiques, avec un fort dégagement thermique ce qui lui permet d'être facilement polymérisable [36].

Persulfate de potassium

C'est des cristaux à blanc, inodores, il est utilisé comme un catalyseur dans la réaction de polymérisation, le persulfate de potassium joue le rôle d'un initiateur de la chaîne par sa décomposition en radicaux libre sous l'effet de la chaleur, Il augmente aussi la vitesse de la réaction [36].

Persulfate d'ammonium

C'est des cristaux blancs, il est utilisé comme un initiateur pour la polymérisation radicalaire des monomères ainsi que pour l'oxydation du grand agent dans plusieurs applications [36].

Hydroxyéthylcellulose

C'est une poudre blanche, facilement soluble dans l'eau froide ou chaude, Hydroxyethylcellulose est utilisée pour modifier la viscosité et la rhéologie [36].

Bicarbonate de sodium

C'est une poudre blanche inodore, est utilisée comme un régulateur de PH dans la préparation de l'émulsion, son rôle principal est de neutraliser le milieu [36].

Nonylphénol Ethoxyle

C'est un liquide transparent visqueux, sans odeur spécifique, c'est un tensioactif non ionique, il est utilisé comme émulsifiant spécifiquement adopté pour la polymérisation de l'acétate de vinyle, son nom commercial est HELMOL [36].

L'eau

L'eau joue le rôle de solvant pour certains composés et de milieux dispersants pour d'autres [36].

III.2.2 Procédé de fabrication du poly acétate de vinyle

Le procédé de fabrication pour la préparation du PVA est semi continu, le monomère est incorporé dans le réacteur de façon contrôlée, afin d'obtenir une meilleure distribution l'émulsion vinylique se fait en deux étapes :

La première étape

Elle consiste à préparer la solution aqueuse. Dans un réacteur semi-continu muni sous agitation moyenne nous versons de l'eau déminéralisée à une température ambiante, lorsque la température de l'eau atteint 45 C° nous ajoutons le colloïde protecteur (NATROSOL), puis à une élévation de température qui est égale à 50 C° nous additionnons le régulateur de PH (bicarbonate de sodium). Nous terminons cette étape par l'ajout des tensioactifs anionique et non ionique quand la température du milieu atteint 55C°.

La deuxième étape

Une fois la température de la solution aqueuse atteint 75 C°, cette dernière a été transférée dans le réacteur de préparation.

Après la préparation des monomères ils seront transférés au réacteur qui contient déjà une solution aqueuse. Nous ajoutons la première quantité de solution de catalyseur (persulfate de potassium ou d'ammonium) pour amorcer la polymérisation, 10% du monomère (AVM et acrylates d'éthyle-2-hexyle) sont ajoutés en petites quantités pour éviter chute de température et l'apparition de mousse. Quand les 10% des monomères sont terminés, les 90% des monomères restants sont versés avec un débit constant pour avoir un temps d'addition d'environ 4h. Pendant cette période la solution de catalyseur est ajoutée en petites quantités toutes les 5 minutes, la température de mélange réactionnel est comprise entre 78 C° et 86C°. A la fin de la réaction, la solution de catalyseur final est ajoutée et le mélange est laissé pour refroidir.

Le temps de préparation de la 2eme étape de fabrication de PVA est assisté par un régulateur PID qui contrôle l'ouverture et fermeture des vannes de refroidissement.

III.2.3 Mode de fonctionnement de processus de fabrication PVA

Au départ, le mélange est fait à vide. Ensuite, lorsque l'on ajoute les 10% de produit et que l'on appuie sur le bouton de démarrage du moteur, la température augmente en raison de la réaction exothermique du produit. Une fois que le mélange est homogène, on ajoute les 90% restants de produit tout en surveillant la température pour ne pas dépasser la consigne de 86°C afin que le PVA ne devienne pas mousseux. À ce stade, le système de refroidissement intervient pour refroidir le processus grâce à la sonde PT100 (dont la définition a été vue dans le chapitre précédent) qui mesure la température et fournit une sortie proportionnelle. Le régulateur PID utilise cette mesure pour comparer la température actuelle à la consigne prédéfinie. Le régulateur calcule cette erreur en soustrayant la température mesurée de la consigne. Ensuite, les actions décrites dans le chapitre précédent interviennent pour stabiliser le système.

Action proportionnelle (P)

L'action proportionnelle est proportionnelle à l'erreur. Le régulateur PID multiplie l'erreur par un coefficient proportionnel (K_p) pour obtenir l'action proportionnelle. L'action proportionnelle est ensuite convertie en un signal 4- 20 mA pour contrôler la vanne de refroidissement. Plus l'erreur est grande, plus l'action proportionnelle est importante [5][35].

Action intégrale (I)

L'action intégrale est utilisée pour éliminer l'erreur statique. Le régulateur PID intègre l'erreur au fil du temps et multiplie cette intégrale par un coefficient intégral (K_i). L'action intégrale est ajoutée à l'action proportionnelle pour corriger l'erreur au fil du temps [5][35].

Action dérivative (D)

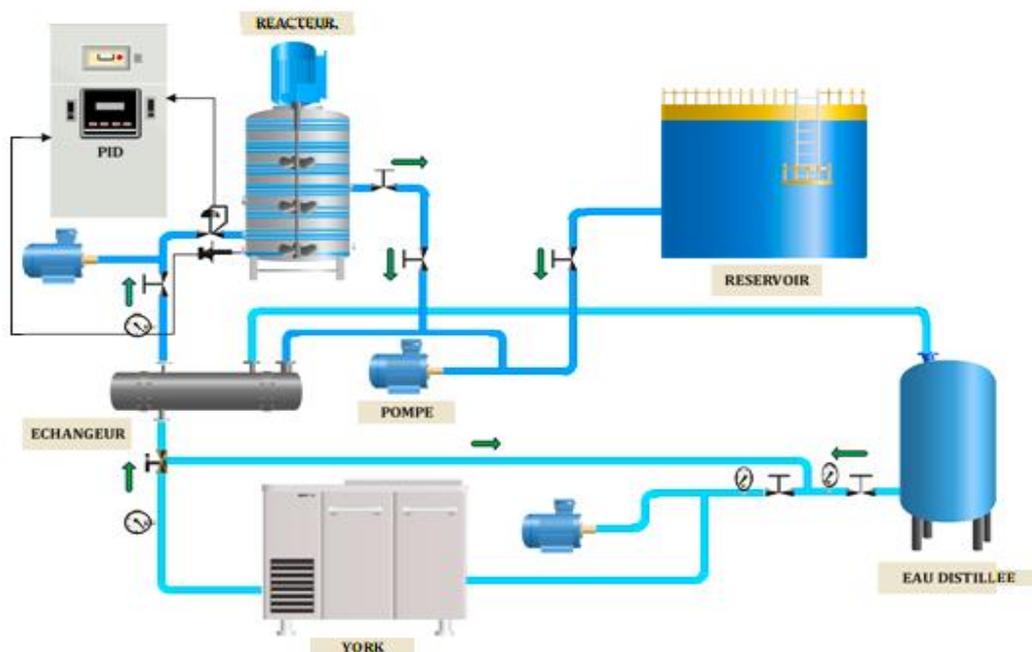
L'action dérivative est utilisée pour anticiper les variations de la variable de processus. Le régulateur PID dérive l'erreur par rapport au temps et multiplie cette dérivée par un coefficient dérivatif (K_d). L'action dérivative est ajoutée à l'action proportionnelle et

intégrale pour ajuster la réponse du système en fonction des changements rapides de la température [5][35].

Contrôle de la vanne de refroidissement

La vanne de refroidissement est une vanne pneumatique, ce qui signifie qu'elle s'ouvre et se ferme par pourcentage, Le régulateur PID peut envoyer un signal pour activer ou désactiver la vanne de refroidissement en fonction de l'erreur de température. Par exemple, si l'erreur est positive, le régulateur peut activer la vanne de refroidissement pour baisser la température.

En utilisant ces actions proportionnelle, intégrale et dérivative, le régulateur PID ajuste en continu la commande de vanne de refroidissement pour maintenir la température à la consigne souhaitée.



Le système de refroidissement dans l'unité ENAP est faite dans le groupe YORK.

Figure 4: Processus de Fabrication de "PVA"

Système de refroidissement

Dans le système de fabrication de PVA bâtiment 109 la ligne 700 (notre cas d'étude) un groupe YORK est utilisé pour refroidir le mélange.

Lors de la préparation de la PVA dans le réacteur, le mélange des diverses matières premières avec agitation déclenche une réaction exothermique. La température générée par cette réaction peut présenter un risque pour la formule si elle dépasse 86°C. Il est donc nécessaire de mettre en place un dispositif de contrôle et de refroidissement afin de maintenir l'efficacité souhaitée.

III.2.3.I.1 Description du système de refroidissement YORK YAES



Figure 5: Le refroidisseur YORK YAES

Les refroidisseurs **YORK YAES** sont conçus pour l'eau ou refroidissement eau-glycol.

Il existe deux familles de produits, standard et haute efficacité. Toutes les unités sont conçues pour être situées à l'extérieur sur le toit d'un immeuble ou au rez-de-chaussée.

L'unité se compose de deux compresseurs à vis, un pour chacun de deux circuits réfrigérants séparés, une seule coque et évaporateur à tubes DX, condenseurs refroidis par air et détendeurs thermostatiques.

Les unités sont entièrement assemblées avec tout interconnexion de la tuyauterie de réfrigérant et du câblage interne, prêt pour l'installation sur le terrain.

Avant la livraison, l'unité est testée sous pression, évacuée, et entièrement chargée de réfrigérant et d'huile dans chacun des circuits frigorifiques indépendants. Après assemblage, le test de fonctionnement est effectué avec de l'eau circulant à travers l'évaporateur pour garantir que chaque circuit réfrigérant fonctionne correctement.

La structure de l'unité est fabriquée à partir de gros calibre, acier galvanisé et recouvert de « Désert Sand » (RAL 1019) peinture en poudre émaillée cuite [11].

Compresseur

Les compresseurs à vis semi-hermétiques à double hélice sont fournis pour garantir une efficacité opérationnelle élevée et des performances fiables. Le contrôle de la capacité est atteint par une seule vanne coulissante. Le compresseur est un type volumétrique caractérisé par deux rotors à gorges hélicoïdales fabriqués à partir de fer forgé.

Le moteur 50 Hz fonctionne à 2950 tr/min pour entraîner directement le rotor mâle qui à son tour entraîne le rotor femelle rotor sur une légère pellicule d'huile.

Chaque compresseur est à entraînement direct, semi-hermétique, rotatif type à double vis et comprend les éléments suivants :

Deux rotors à vis, à profils asymétriques, fabriqués en acier forgé.

Un boîtier de compresseur en fonte usiné avec précision pour fournir un dégagement optimal pour les rotors.

Un clapet anti-retour de décharge interne pour empêcher le rotor backspin pendant l'arrêt.

Un silencieux de décharge interne à réglage acoustique pour minimiser le bruit, tout en optimisant le débit pour une performance maximum.

Vannes de service d'arrêt de refoulement (avec aspiration en option Vannes d'arrêt).

Un gaz d'aspiration fiable refroidi à haut rendement, accessible moteur hermétique avec protection redondante contre les surcharges utilisant à la fois une protection par thermistance et une protection contre les surcharges de courant.

Un écran de gaz d'aspiration et réparable, 0,5 micron plein filtre à huile à débit dans le boîtier du compresseur.

Du gaz réfrigérant est injecté dans le vide créé par le maillage du mâle à cinq lobes et du mâle à sept lobes rotor femelle. Un engrènement supplémentaire des rotors ferme le rotor se visse jusqu'à l'orifice d'aspiration et progressivement comprime le gaz dans une direction axiale vers le port de déchargement. Le gaz est comprimé en volume et augmenté en pression avant de sortir à un niveau conçu volume à l'extrémité de refoulement du carter du rotor. Depuis les cycles d'admission et de décharge se chevauchent, ce qui entraîne un flux régulier de gaz est maintenu.

Les rotors sont logés dans un compresseur en fonte boîtier usiné avec précision pour fournir un dégagement pour les rotors. Le contact entre le mâle et le rotor femelle roule principalement sur une bande de contact sur chacun des cercles primitifs du rotor. Il en résulte pratiquement aucune usure du rotor et fiabilité accrue.

Le compresseur intègre un système anti friction complet Conception de roulement pour une puissance absorbée réduite et une augmentation fiabilité. Quatre roulements à rouleaux cylindriques séparés (durée de vie L10 de 50 000 heures dans les conditions de conception) charges radiales. Roulements à billes à contact oblique poignée axiale charges. Ensemble, ils maintiennent un positionnement précis du rotor à tous les rapports de pression, minimisant ainsi les fuites et maintiennent l'efficacité. Un clapet anti-retour sans ressort est installé dans le boîtier de refoulement du compresseur pour empêcher le retour en arrière du rotor du compresseur dû au système gradients de pression du réfrigérant pendant l'arrêt.

Le refroidissement du moteur est assuré par les gaz aspirés de l'évaporateur circulant à travers le moteur.

Le compresseur est lubrifié en éliminant l'huile du réfrigérant à l'aide d'un séparateur d'huile externe. Le l'huile sous pression est ensuite refroidie dans les serpentins du condenseur et renvoyée au compresseur pour la lubrification. Le la pression de service du compresseur est de 31 bars (450 PSIG).

Un thermoplongeur de 350 watts (230 V 1 Ø 50Hz) est situé dans le compresseur. Le chauffage est la température activée pour éviter la condensation du réfrigérant [11].

- **Démarrage du moteur**

Deux types de démarrage de moteur de compresseur sont disponibles : Démarreur à transition ouverte étoile/triangle et étoile/triangle en option démarreur de transition fermé. Le démarreur étoile/triangle standard utilise 3 moteurs contacteurs et un relais temporisé de transition. L'optionnel Le démarreur étoile/triangle fermé utilise 4 contacteurs de moteur, un ensemble de résistances de transition et un relais à retard de transition. Le démarrage étoile/triangle permet de limiter le courant d'appel à environ 33% LRA avec la transition fermée option réduisant l'étoile transitoire au courant delta.

Lorsque le microprocesseur émet un signal de démarrage pour exécuter un compresseur, il tourne en étoile pendant 4 à 10 secondes en fonction du courant du moteur détecté par le microprocesseur. Normalement, la transition vers le delta prend 10 secondes si le courant est inférieur à 125 % FLA. Si le moteur courant dépasse 125% FLA, la transition se fait vers delta après un temps d'exécution en étoile d'au moins 4 secondes [11].

- **Contrôle de capacité**

Les compresseurs démarreront à la position de charge minimale et fourniront une plage de contrôle de capacité de 20 % à 100 % par compresseur à l'aide d'un tiroir à fonction continue. La pression de sortie contrôlée par microprocesseur la vanne de régulation de la capacité de régulation commandera la capacité du compresseur indépendante de l'entrée de la vanne de régulation pression et équilibrera la capacité du compresseur avec la charge de refroidissement.

Le rappel automatique par ressort de la vanne de régulation de capacité à la position de charge minimale garantira que le compresseur démarrage à charge moteur minimale [11].

Séparateur d'huile

Chaque circuit dispose d'un système de gaz augmenté à haut rendement séparateur d'huile à impact pour maximiser l'huile extraction sans support fragile à décomposer.

Le séparateur d'huile est monté dans la conduite de refoulement du compresseur. Le gaz de décharge haute pression est forcé autour d'un virage à 90 degrés. Le pétrole est expulsé vers l'extérieur le séparateur par action centrifuge et capturé sur un grillage où il s'écoule vers le fond de l'huile séparateur et passe à la section du refroidisseur d'huile du serpent du condenseur où il est refroidi avant de retourner au compresseur.

L'huile s'écoule vers le compresseur par un filtre à huile remplaçable de 0,5 à 3,0 microns et alimentation en huile solénoïde, est à haute pression. Cette "huile" haute pression "injection" force l'huile dans le compresseur où elle se trouve alimentée par gravité aux engrenages et aux roulements pour la lubrification.

Après avoir lubrifié les engrenages et les roulements, il est injecté à travers des orifices sur un filetage fermé près de l'extrémité d'aspiration des rotors. L'huile est automatiquement injectée en raison de la différence de pression entre le refoulement

pression et la pression réduite à l'extrémité d'aspiration des rotors. Cela lubrifie les rotors et fournit un joint d'huile contre les fuites autour des rotors pour assurer compression du fluide frigorigène (efficacité volumétrique)

L'huile assure également le refroidissement en transférant une grande partie de la chaleur de compression du gaz vers l'huile, les températures de décharge baissent et réduisent les risques de panne d'huile.

L'huile injectée dans la cage du rotor s'écoule dans les rotors à une vitesse pointez environ 1,2 x aspirations. Cela garantit qu'un un différentiel minimum d'au moins 1,8 bar (30 PSI) existe entre le refoulement et 1,2 x aspirations, pour forcer l'huile dans carter du rotor, un minimum de 0,6 bar (10 PSI) est tout ce qui est nécessaire pour assurer la protection du compresseur. Huile la pression est mesurée comme la différence entre la pression de refoulement et la pression de l'huile entrant le carter du rotor.

La pression de service maximale du séparateur d'huile est de 31 bars (450 PSIG). Une soupape de décharge est installée dans le séparateur d'huile tuyauterie. Le niveau d'huile doit être au-dessus du milieu du Regard d'huile "inférieur" lorsque le compresseur est en marche.

Le niveau d'huile ne doit jamais dépasser le haut du Regard supérieur [11].

Refroidisseur d'huile

Le refroidissement de l'huile est assuré par un système à ailettes dédié refroidi par air. Échangeur de chaleur à tubes situé dans le condenseur partie de l'unité. L'huile sortant du séparateur d'huile passe à travers cette section de refroidisseur d'huile où elle est refroidie avant retour au compresseur [11].

Circuits réfrigérants

Chaque circuit réfrigérant utilisé un tuyau réfrigérant en cuivre formé sur des machines à cintrer contrôlées par ordinateur pour réduire le nombre de joints brasés, ce qui permet d'obtenir un système résistant aux fuites.

Les composants de la conduite de liquide comprennent : une vanne d'arrêt manuelle avec port de chargement, un noyau amovible à haute absorption filtre déshydrateur, un voyant avec indicateur d'humidité et un détendeur thermostatique.

Les conduites d'aspiration sont recouvertes d'une isolation à cellules fermées [11].

Économiseurs

Certains modèles de la gamme sont équipés d'économiseurs de circuit réfrigérant. C'est un échangeur de chaleur à plaques réfrigérant-réfrigérant pour maximiser la capacité et l'efficacité de l'unité en atteignant sous-refroidissement supplémentaire par réfrigérant liquide. Le contrôle de l'unité Le système contrôle l'économiseur via une électrovanne [11].

Condenseur

Les serpentins du condenseur sont fabriqués à partir de matériaux sans soudure, coefficient de condensation élevé et amélioré en interne, tubes de cuivre résistants à la corrosion disposés en quinconce rangées et mécaniquement dilatées en corrosion

ailerons en alliage d'aluminium résistant avec colliers d'ailerons pleine hauteur. Ils ont une pression de service nominale de 31 bars (450 PSIG).

Les unités standard ont des condenseurs équipés d'une seule vitesse Ventilateurs. En option, des ventilateurs à deux vitesses peuvent être installés. Tous les fans sont équilibrés dynamiquement et statiquement, entraînement direct avec composite renforcé de fibre de verre résistant à la corrosion pales moulées dans une croix à profil aérodynamique complet et à faible bruit section, fournissant une décharge d'air verticale à partir d'une extension Orifices pour plus d'efficacité et un faible niveau sonore. Chaque ventilateur est situé dans un compartiment séparé pour éviter les flux croisés pendant cyclisme des fans. Protections de gros calibre, enduites de PVC l'acier galvanisé est fourni.

Les moteurs de ventilateur sont à haut rendement, à entraînement direct, 8 pôles, 3 Phase, classe « F », protégée contre les surcharges de courant, totalement type fermé (TEAC) avec double étanchéité, en permanence lubrifiés, roulements à billes [11].

Évaporateur

L'évaporateur est une conception à coque et tube avec réfrigérant côté tube et eau côté calandre. Les tubes sont formés en forme de « U » et maintenus dans un tube bundle, qui est libre de s'étendre indépendamment du Shell. Un circuit frigorifique indépendant est prévu pour chaque compresseur. La conception côté eau (coquille) fonctionne la pression est de 10,3 bar (150 psi) et côté réfrigérant (tubes) est de 24,5 bars (355 psi).

Les évaporateurs ont des déflecteurs de passage d'eau fabriqués en composite résistant à la corrosion/non métallique matériau, têtes amovibles pour accéder à l'intérieur tubes de cuivre amélioré et sans soudure. Évent d'eau et des raccords de vidange sont également inclus.

Les buses d'eau sont équipées de brides PN 16 selon ISO 7005 parties 1a. Brides à souder d'accompagnement assorties les brides de l'évaporateur sont fournies en vrac pour le terrain installation par entrepreneur. Tous les écrous, boulons et les joints sont inclus.

Les évaporateurs sont équipés d'un thermostat chauffage contrôlé pour une protection jusqu'à -29°C ambiante et sont isolés avec de la mousse flexible à cellules fermées de 19 mm [11].

III.2.3.1.2 Le fonctionnement de système de refroidissement

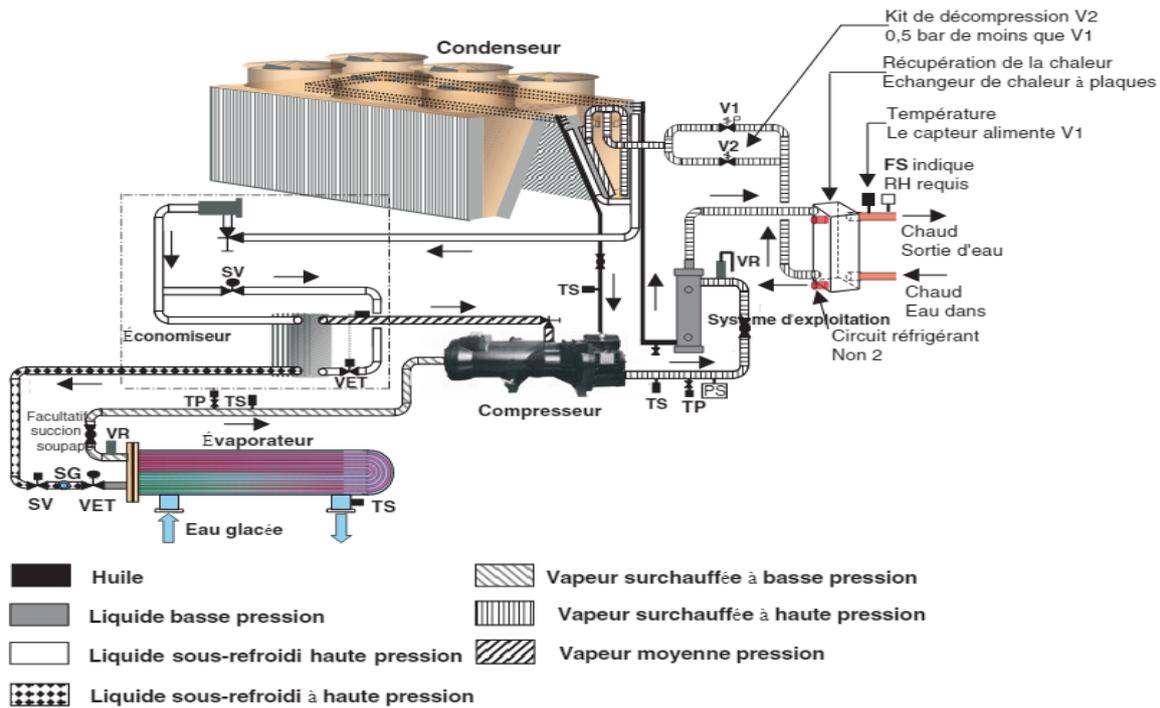


Figure 6: Le fonctionnement de refroidisseur YORK YAES

En bas à gauche de la figure, le fluide frigorigère est détendu et se refroidit dans la vanne d'expansion. Ce fluide frigorigère basse pression entre dans l'évaporateur où il s'évapore, se surchauffe par les calories cédées par l'eau circulant dans la virole en (cette eau froide alimente le bâtiment à climatiser).

La vapeur frigorigère basse pression réchauffée entre dans le compresseur où elle est comprimée à haute pression tout en augmentant encore sa température. A la sortie du compresseur, la pression et la température de la vapeur frigorigère sont augmentées. Cette vapeur HP traverse le séparateur d'huile où un filtre la débarrasse de l'huile absorbée dans le compresseur.

L'huile est refroidie dans le refroidisseur d'huile OC et réinjectée dans le compresseur. Le frigorigère haute température circule dans le condenseur CDR où il est refroidi par le flux d'air créé par les ventilateurs et se liquéfie. La chaleur est rejetée dans l'air ambiant. Le fluide frigorigère condensé et liquide entre dans l'échangeur nommé "économiseur" où une petite quantité du liquide frigorigère est prélevée et détendue à la moyenne pression MP afin de sous-refroidir le frigorigère liquide traversant l'autre circuit de cet échangeur. La petite quantité de frigorigère MP retourne directement au compresseur. Le fluide frigorigère sous-refroidi parvient à la vanne d'expansion et le circuit recommence [11].

III.3 Description des instruments du processus

Dans ce qui suit, nous présenterons les instruments du processus et qui seront utilisés pour assurer le fonctionnement du procédé de fabrication et de refroidissement principalement.

III.3.1 Le régulateur de température PID

Un régulateur de température **PID** permet au régulateur d'allumer ou éteindre le système chauffant jusqu'à la valeur optimale pour le procédé.



Figure 7: Le régulateur de température "PID"

III.3.2 Vanne pneumatique

La vanne de régulation automatique pneumatique 2 voies est adaptée au passage de fluides tels que l'eau froide.



Figure 8: Vanne pneumatique

III.3.3 Les pompes

La pompe est un dispositif qui assure la circulation d'un fluide dans une tuyauterie. La pompe aspire et refoule le fluide engendrant un accroissement de vitesse et de pression du fluide.



Figure 9: Pompe

III.3.4 Capteur de température

Ce sont des capteurs résistifs qui permettent l'effet de réchauffement ou de refroidissement sur leur résistance en signal électrique.

III.3.5 La sonde PT100

Est constituée d'un filament en platine, entourant une tige de verre, dont la caractéristique est de changer en fonction de la température. Sa résistance est de 100Ω pour 0°C . Elle s'élève en fonction de la température. En injectant à la sonde un courant constant et continu, il suffit alors de mesurer la tension qui est proportionnelle à la résistance.



Figure 10: La sonde "PT100"

III.3.6 Les débitmètres

Les débitmètres sont des appareils de mesure du débit d'un fluide ou d'un gaz s'écoulant dans une canalisation. Les débitmètres varient selon le niveau du débit et la nature du fluide.



Figure 11: Débitmètre

III.3.7 Soupape de sûreté

Appareil de robinetterie destiné à limiter la pression d'un fluide à une valeur prédéterminée et fonctionnant par ouverture d'un obturateur lorsque cette valeur prédéterminée est atteinte ; ce qui permet d'empêcher, d'éviter un bouleversement, ce qui sert d'exutoire.



Figure 12: Soupape de sûreté

III.3.8 Réacteur

En chimie, un réacteur est une enceinte ou récipient apte à la réalisation et l'optimisation de réactions chimiques et généralement de procédés de transformation de la matière (génie des procédés). Le but recherché dans un réacteur est l'homogénéité du milieu réactionnel du point de vue de la température et du mélange des réactifs.



Figure 13: Réacteur

III.3.9 Moteur

Un moteur est un dispositif transformant une énergie non-mécanique (éolienne, chimique, électrique, thermique par exemple) en une énergie mécanique.



Figure 14: Moteur

III.3.10 Un manomètre

Est un instrument servant à mesurer une pression. On a développé depuis les débuts de l'époque moderne de nombreux appareils pour mesurer les pressions et le degré de vide : ces instruments de mesure sont appelés tantôt capteurs de pression, tantôt sondes (pour le degré de vide). Le manomètre est un instrument de mesure de pression, destiné en principe à mesurer des pressions voisines de la pression atmosphérique. Le terme « manomètre » renvoie plutôt dans son acception courante à des instruments de mesure à colonne de liquide.



Figure 15: Manomètre

IV. La régulation

IV.1 Introduction

La régulation des procédés industriels regroupe tous les moyens physiques et techniques déployés pour maintenir une grandeur physique qui doit être régulée, par une valeur souhaitée, appelée point de consigne. Lorsqu'une perturbation ou un changement de point de consigne survient, la régulation prend des mesures correctives.

Action sur cette grandeur physique du procédé, appelée grandeur de contrôle.

Dans l'exemple de la figure, La température du liquide est régulée en influençant le débit de vapeur de l'échangeur thermique et ce quelles que soient les perturbations : débit de charge, température d'entrée de charge, etc.

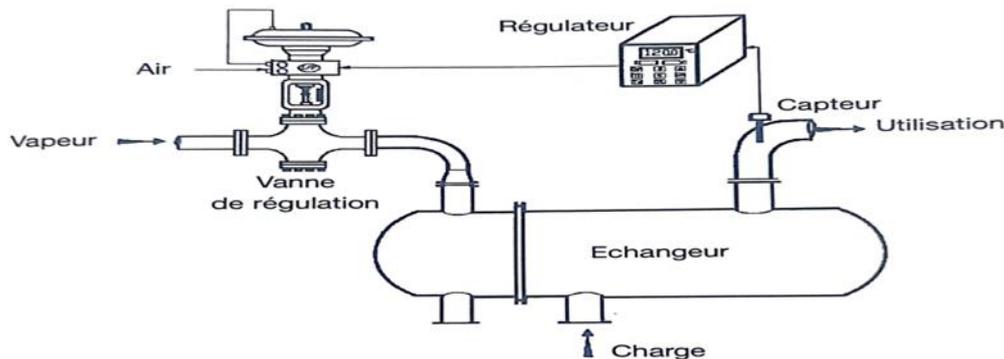


Figure 16: Régulation de température d'un échangeur thermique

Dans ce cas, le capteur de température, le régulateur et la vanne représentent des dispositifs qui permettent de mettre en œuvre la technique de régulation la plus courante qui est la boucle fermée. Selon les processus et les objectifs à atteindre, il existe de nombreux types d'appareils et de techniques [1].

Le terme de régulation est employé lorsqu'on cherche à combattre des perturbations afin de garder une valeur constante par exemple, une température, une pression, un débit ou une hygrométrie... [1].

Parmi les appareils :

- Régulateurs monoblocs analogiques et numériques.
- Systèmes numériques de contrôle commande de procédé.
- Opérateurs de calcul arithmétiques et dynamiques [1].

Parmi les techniques :

- Régulation en boucle fermée.
- Régulation auto-adaptée.
- Régulation discontinue.
- Régulation en cascade.
- Régulation a priori.
- Régulation split-range.
- Régulation de rapport.
- Régulation auto-sélective.
- Régulation par correcteur de SMITH.
- Régulation par modèle de référence.
- Régulation par logique floue.
- Régulation multivariable.

Régulation Batch [1].

IV.2 Définition de la régulation industrielle

La régulation industrielle est une science qui étudie le maintien des grandeurs physiques à des valeurs désirées pour le bon fonctionnement des procédés de fabrication. Elle demande des connaissances de toutes les autres sciences (mathématique, physique, chimie, électronique, électrotechnique, mécanique et toutes les autres sciences). Elle est présente dans tous les domaines de la vie quotidienne. Les techniques de régulation ont évolué beaucoup et rapidement surtout avec l'évolution des techniques numériques et de communication (microprocesseur, automate, robotique etc.) [2].

IV.2.1 Régulation manuelle et régulation automatique

Nous ferons une petite comparaison entre régulation manuelle et régulation automatique. Dans la régulation manuelle, l'opérateur est le facteur clé. Il doit utiliser les moyens qu'il possède naturellement tels que les yeux, le cerveau et les mains pour pouvoir contrôler un processus ; Sa présence avant le processus est essentielle.

D'autre part, la régulation automatique nécessite des instruments de mesure et de contrôle spécialisés ainsi que des dispositifs de réglage automatique pour contrôler le processus de production sans la présence d'un opérateur [2].

IV.2.2 La boucle de régulation

La boucle de régulation dans sa forme la plus simple, est toujours constituée des éléments suivants :

Le capteur fournit un signal proportionnel à la grandeur mesurée, généralement un signal 4-20 mA¹ ;

Un régulateur contrôle via un signal normalisé 4-20 mA ;

Un dispositif de régulation, généralement une vanne de régulation.

Le schéma de la Figure est une structure en boucle fermée : le régulateur agit sur le procédé via l'organe de réglage et contrôle l'évolution de la grandeur réglée représentée par la mesure. la régulation industrielle met en œuvre un algorithme de calcul appelé PID, ces lettres désignant les actions du régulateur dénommées : proportionnelle, intégrale et dérivée [3].

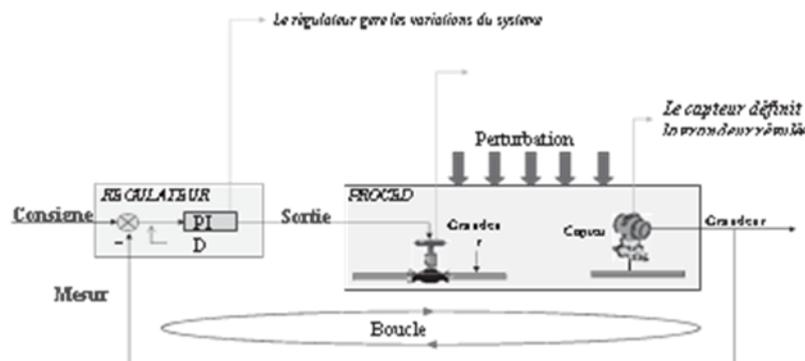


Figure 17: La boucle de régulation

IV.3 Régulateur industriel

Est un dispositif utilisé dans le domaine de l'industrie pour contrôler et réguler divers paramètres tels que la température, la pression, le débit, etc. Il permet de maintenir ces paramètres à des niveaux prédéfinis pour assurer un fonctionnement optimal des équipements industriels. C'est un outil essentiel pour garantir la qualité et la sécurité des processus industriels.

Dans la boucle de régulation, le régulateur joue un rôle stratégique : lors des changements de consigne, ramener la mesure à la consigne le plus rapidement possible, puis agir pour ramener la mesure à la consigne lorsqu'une perturbation survient [4].

IV.3.1 Architecture d'un régulateur

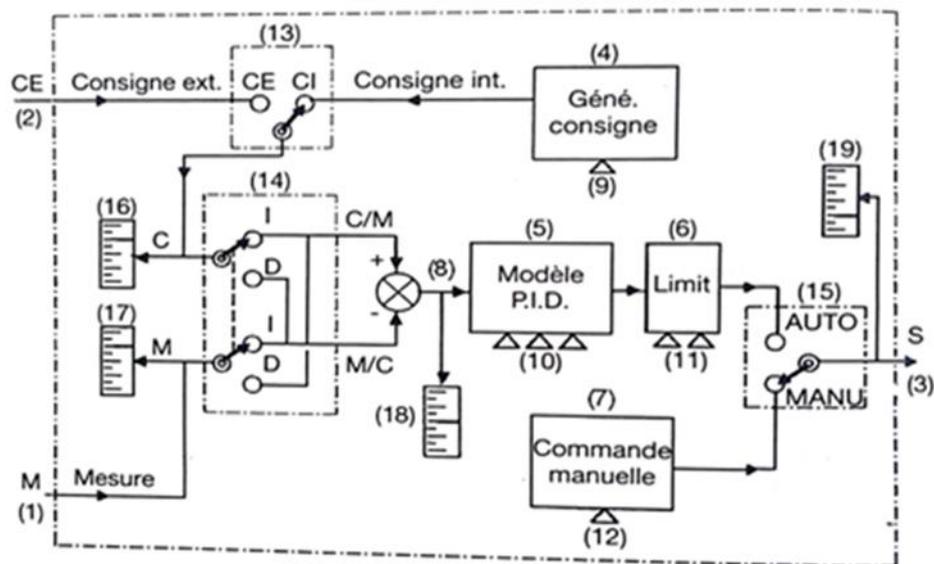


Figure 18: constitution d'un régulateur

Les signaux

- 1 : Entrée mesure : ce signal, issu du transmetteur, représente la grandeur à régler.
- 2 : Consigne externe : provient d'un instrument extérieur.
- 3 : Sortie signal de commande de l'organe de réglage (vanne...) L'échelle standard la plus utilisée est : 4 à 20 mA.

Les blocs

- 4 : Générateur de consigne.
- 5 : Module PID : en automatique, la sortie de ce bloc est celle du régulateur. La position automatique correspond au fonctionnement normal du régulateur.
- 6 : Limiteur de sortie : limite le signal de sortie, en position automatique, à des valeurs haute et basse préfixées.
- 7 : Commande manuelle : Générateur de sortie manuelle.
- 8 : Détecteur d'écart entre la mesure et la consigne.

Les réglages

- 9 : Réglage de consigne interne.

- 10 : Réglage des actions P, I et D.
- 11 : Réglage des limites haute et basse.
- 12 : Réglage de la sortie du régulateur en position manuelle.

Les sélecteurs

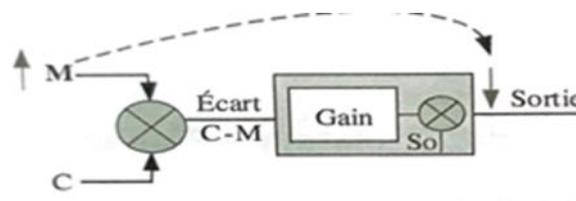
- 13 : Sélecteur de consigne interne ou consigne externe.
- 14 : Sélecteur du sens d'action du régulateur.
- 15 : Sélecteur de fonctionnement automatique ou manuel.

Les indicateurs

- 16 : Indicateur de consigne.
- 17 : Indicateur de mesure.
- 18 : Indicateur d'écart Mesure - Consigne.
- 19 : Indicateur de sortie [2].

IV.3.2 Sens d'action d'un régulateur

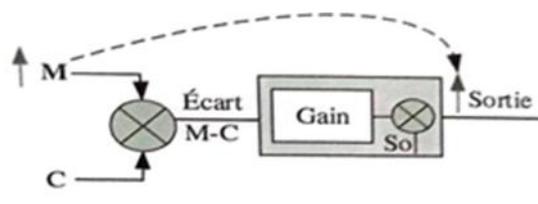
Physiquement, le sens effectif est un simple inverseur qui croise les signaux de mesure et de consigne. La direction de l'action n'est pas l'indifférence : Il n'y a qu'une seule option possible. Dans les contrôleurs plus anciens, la direction valide était sélectionnée via un interrupteur. Si vous souhaitez utiliser le avec votre matériel numérique actuel, cliquez simplement sur la case qui sélectionne les calculs que vous souhaitez effectuer. La sélection s'effectue lors de la mise en service de l'appareil. Ne peut pas être modifié ultérieurement [3].

I. Sens inverse**Figure 19: Régulateur sens inverse**

$$E=(C-M)$$

$$\text{Sortie} = (\text{écart} \times \text{gain}) + \text{sortie initiale} = [(C-M) \times \text{gain}] + \text{sortie initiale}$$

Lorsque le régulateur est en sens inverse, la sortie diminue quand la mesure augmente.

II. Sens direct**Figure 20: Régulateur sens direct**

$$E= (M-C)$$

Sortie = (écart × gain) + sortie initiale = [(M-C) × gain] + sortie initiale

Lorsque le régulateur est en sens direct, la sortie augmente quand la mesure augmente.

Si la sortie diminue si la consigne augmente.

Ces définitions sont principalement utilisées par les fabricants d'équipements réglementés. Mais quelques fabricants considèrent les instructions plutôt que les mesures. Cela renverse l'argument [3].

IV.3.3 Le sens de régulateur utilisé

Tout d'abord, le sens de fonctionnement est spécifié dans les instructions de l'appareil.

Sinon, vérifiez la fonctionnalité avec un test rapide.

Le contrôleur peut être configuré pour un fonctionnement proportionnel uniquement avec un gain de 1 ou une bande proportionnelle de 100 %.

Connectez un simulateur approprié pour générer le signal de mesure (par exemple générateur 4-20 mA).

Mettre une consigne correspondant à 50% de l'étendue de mesure.

Régler le simulateur sur 50° de plage de mesure.

Augmenter le signal de mesure et observer le sens de modification de la sortie [4].

IV.3.4 Choix du sens d'action de régulation

Le sens du régulateur dépend du type de l'actionneur (vanne automatique NF ou NO) et de la position de cette vanne dans le procédé.

Le sens d'action de l'ensemble vanne de régulation est direct si la vanne s'ouvre lorsque le signal de commande U augmente, sinon inversé [3].

IV.3.5 Définition de la régulation PID

Le régulateur standard le plus utilisé dans l'industrie est le régulateur PID (Proportionnel Intégral Dérivé). Il permet de régler à l'aide de ses trois paramètres les performances (amortissement, temps de réponse, ...) d'un processus modélisé par un deuxième ordre. Nombreux sont les systèmes physiques qui, même en étant complexes, ont un comportement voisin de celui d'un deuxième ordre. Par conséquent, le régulateur **PID** est bien adapté à la plupart des processus de type industriel et est relativement robuste par rapport aux variations des paramètres du procédé.

Si la dynamique dominante du système est supérieure à un deuxième ordre, ou si le système contient un retard important ou plusieurs modes oscillants, le régulateur **PID** n'est plus adéquat et un régulateur plus complexe (avec plus de paramètres) doit être utilisé, au dépend de la sensibilité aux variations des paramètres du procédé [4].

La loi de commande du régulateur **PID**

$$u(t) = K_p * e(t) + \frac{1}{T_i} * \int_{-\infty}^t e(r). dr + T_d \cdot \frac{de}{dt}$$

Fonction de transfert du régulateur **PID**

$$G_c(s) = \frac{U(s)}{E(s)} = K_p \frac{1+sT_i+s^2T_iT_d}{sT_i}$$

IV.3.6 Les actions proportionnelle /intégrale /drivé

Les actions du régulateur à une très forte influence sur le comportement de la grandeur régnante ou de la valeur mesurée par rapport à la consigne [5][35].

I. Action proportionnelle (P)

L'augmentation du gain réduit, mais n'élimine pas, les erreurs statiques, rendant le système de contrôle plus rapide mais rendant le système plus instable. Par conséquent, le gain ne peut pas être augmenté sans limite, mais est limité par un gain critique là où le système entre dans la zone d'instabilité.

L'efficacité du contrôleur est proportionnelle à la différence entre la valeur mesurée et la valeur réglée. Cette différence est appelée bande proportionnelle. Par exemple, une tension artérielle de +3 °C signifie que la sortie du régulateur varie entre 0 et 10 volts (vanne 3 voies, registre) dans une plage de variation de 3 °C [5][35].

Plus l'erreur est grande, plus l'action proportionnelle est importante [5][35].

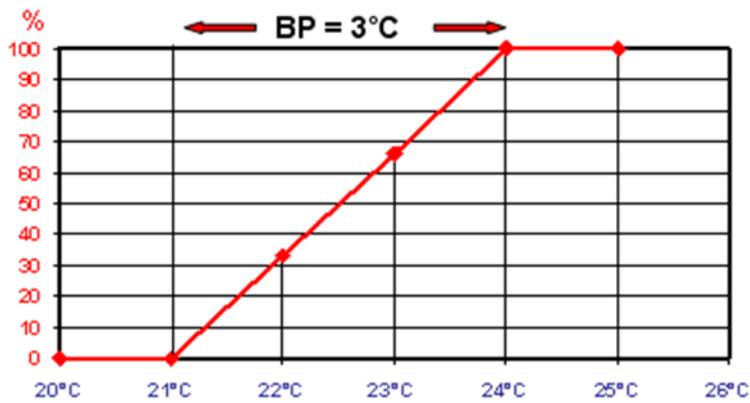


Figure 21: Action Proportionnel

II. Action intégrale (I)

L'intégration permet au contrôleur de prendre en compte le taux de changement entre la valeur mesurée réelle et le point de consigne. Cette notion de temps s'exprime en secondes. La composante intégrale n'est jamais utilisée seule, mais complète l'action proportionnelle (contrôle PI) et permet d'ajuster la réponse du régulateur en fonction du temps pour compenser cette différence [5][35].

III. Action dérivative (D)

L'action dérivative est utilisée pour anticiper les variations de la variable de processus. Le régulateur PID dérive l'erreur par rapport au temps et multiplie cette dérive par un coefficient dérivatif (Kd). L'action dérivative est ajoutée à l'action proportionnelle et

intégrale pour ajuster la réponse du système en fonction des changements rapides de la température [5][35].

IV.4 Les instruments de régulation

IV.4.1 La vanne

Une vanne de régulation est un dispositif à commande mécanique qui modifie le débit ou la surface de passage d'un fluide dans un tuyau. Le dispositif se compose d'une vanne reliée à un actionneur qui peut modifier la position de l'élément de fermeture ou du volet en fonction des signaux du système de contrôle. Les actionneurs peuvent être entraînés par de l'énergie pneumatique, électrique, hydraulique ou toute combinaison de ces énergies [2].

IV.4.1.1 Eléments d'une vanne

Quel que soit le fabricant, le type de vanne ou sa génération, une vanne automatique est toujours décomposable technologiquement en trois parties bien distinctes :

Servomoteur : est un organe qui reçoit un signal d'un contrôleur ou d'une télécommande manuelle et le convertit en une force de valeur capable de déplacer une charge d'une position définie à une autre.

Le corps : est l'élément d'une vanne automatique, dans lequel le liquide peut s'arrêter ou au contraire circuler plus ou moins librement.

Le positionneur : est un relais intégré au circuit d'air modulant du contrôleur. Son rôle est de positionner avec précision la vanne en fonction des signaux émis par le contrôleur, garantissant ainsi un positionnement rapide et précis [2].

IV.4.1.2 Les types de la vanne

Vanne manuelle

En particulier dans les installations industrielles, les vannes manuelles qui contrôlent le débit de liquides sont des équipements simples mais critiques qui nécessitent généralement une utilisation fréquente par le personnel. Il existe deux types avec bras (vanne tout ou rien), avec robinet [2].



Figure 22: Vanne manuel

Vanne automatique

La vanne automatique est l'actionneur le plus utilisé dans l'industrie. Elle sert à faire varier le débit dans une conduite fluide. Elle représente une résistance d'écoulement variable comme un potentiomètre électrique qui représente une résistance variable

dans un circuit électrique. Elle exécute l'ordre provenant du régulateur. On peut aussi l'appeler organe de réglage ou organe d'exécution. Elle relie la grandeur réglante qui est le débit de passage à travers son corps et la commande du régulateur [2].

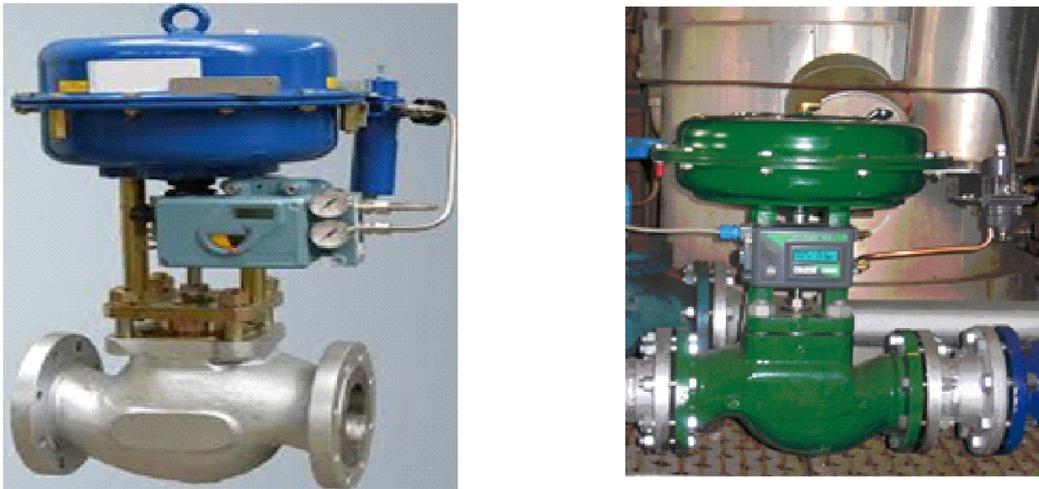


Figure 23: Vanne automatique

IV.4.1.3 Choix de vanne

Pour sélectionner une vanne, vous devez d'abord comprendre le processus d'installation de la vanne.

Le facteur le plus important est d'abord la valeur de la pression différentielle (ΔP) qui doit maintenir la vanne fermée (force sur la vanne = surface de pression de la vanne).

Cela signifie que certains types de vannes qui ne disposent pas de l'entraînement approprié sont déterminés ou exclus en fonction de leur diamètre.

Ensuite, vous devez définir les caractéristiques de température, de pression, de pression différentielle du procédé ainsi que le type de fluide à contrôler (liquide, gaz, vapeur, chargé ou non).

Un autre paramètre important est le coefficient de débit C_v (C_v est proportionnel à la surface de passage entre le siège et la vanne).

Le choix du corps de vanne dépend non seulement de l'application, mais également du savoir-faire du fabricant.

De plus, d'autres critères d'application tels que l'étanchéité et la précision du positionnement peuvent signifier qu'une vanne particulière est plus ou moins adaptée à vos besoins [6].

IV.4.1.4 Certains fabricants de vannes

Bermad Bürkert.

Emerson Process Management Fisher.

Samson.

Flowserve.

IV.4.2 Les capteurs

Définition

Un capteur est un dispositif d'acquisition d'informations qui produit une quantité différente d'un type différent, très souvent de l'électricité, à partir d'une quantité physique. Cette quantité représente la quantité d'échantillonnage et peut être utilisée à des fins de mesure ou de contrôle.

Un transmetteur est un appareil qui convertit les signaux de sortie du capteur en signaux de mesure standard et sert de lien entre le capteur et le système de contrôle [2].

IV.4.2.1 Les types de capteurs

Capteur T.O.R. (Tout ou Rien) qui fournit un signal de sortie logique. 0 ou 1.

Exemple de détecteur de fin de course [7].

Capteur numérique qui fonctionne conjointement avec un compteur pour fournir un signal de sortie numérique.

Exemple : Un capteur ou un codeur incrémental pour mesurer le mouvement d'un coulisseau dans une machine à commande numérique [7].

Capteur analogique ou proportionnel capable de prendre en compte la valeur réelle d'une grandeur physique.

Exemple : Sonde de température [7].



Figure 24: Exemple des capteurs

IV.4.2.2 Choix de capteur

La sélection recommandée s'effectue en deux étapes. L'organigramme suivant illustre cette approche. Celui-ci sélectionne une famille de détecteurs sur la base de critères simples [10].

Phase n°1 : détermine la meilleure famille de détecteurs pour votre application en répondant aux questions suivantes :

Type d'objet à détecter : solide, liquide, gaz, métal.

Possibilité de contact avec des objets.

Vitesse de défilement.

Distance d'objet.

Espace pour intégrer le détecteur dans la machine [10].

Phase n°2 : L'objectif est de déterminer le type et la référence du détecteur que vous recherchez.

Cette deuxième phase considère :

Environnement : température, humidité, poussière, saillies diverses, etc.

Alimentation : AC ou continue.

Signal de sortie électromécanique statique.

Type de connecteur : câble, borné, connecteur [10].

Exemple : La sonde PT 100

Est un capteur de température utilisé dans le secteur industriel (agroalimentaire, chimie, raffineries...).

Ce capteur est constitué d'une résistance en platine, également appelé RTD (détecteur de température à résistance).

La valeur de sortie du Pt100 est de 100 ohms correspondant à une température de 0°C. Ces capteurs disposent d'un capot de protection pour le capteur de température. Le PT100 offre une excellente précision sur une large plage de températures (-200 à 200°C). Jusqu'à +850°C).



Figure 25: La sonde "PT100"

Principe de mesure

Relation la résistance /température.

$$\frac{R_T}{R_0} = 1 + At + Bt^2$$

R_T = résistance du thermomètre à la température T.

R_0 = résistance du thermomètre à 0°C.

t= Température (°C).

$A=3.90838 \cdot 10^{-3}$ et

$B=-5.775 \cdot 10^{-7}$

Cette relation est utilisée pour des températures supérieures à 0°C [8].

V. Régulation et identification d'un système

V.1 Introduction

L'identification et la régulation d'un système sont des aspects fondamentaux de l'ingénierie et du contrôle automatique. L'identification consiste à comprendre et modéliser le comportement dynamique d'un système, permettant ainsi de créer des représentations mathématiques précises. En parallèle, la régulation s'attache à ajuster les paramètres du système en temps réel afin de maintenir ses performances dans des limites spécifiées. Ces techniques sont essentielles dans des domaines variés tels que l'automatisation industrielle, la robotique et les systèmes de contrôle, garantissant ainsi une stabilité et une efficacité optimales des systèmes complexes.

V.2 Les étapes d'identification d'un système

L'identification d'un système comprend généralement plusieurs étapes :

1. Collecte de données

Acquisition de données expérimentales ou observation du système pour obtenir des mesures de ses réponses à différentes entrées.

2. Prétraitement des données

Nettoyage et traitement des données pour éliminer le bruit, les erreurs de mesure et autres anomalies.

3. Choix du modèle

Sélection du type de modèle mathématique (linéaire, non linéaire, etc.) qui représente le mieux le système. Being red, Mars is a Cold place

4. Estimation des paramètres

Utilisation d'algorithmes d'estimation pour ajuster les paramètres du modèle aux données observées...

5. Validation du modèle

Vérification de la qualité du modèle en le testant avec des données non utilisées lors de l'estimation pour s'assurer de sa capacité à prédire correctement le comportement du système.

V.3 Simulation Matlab

Les étapes à suivre

V.4.1 Identification

1. Chargement des données dans MATLAB

Importer les données

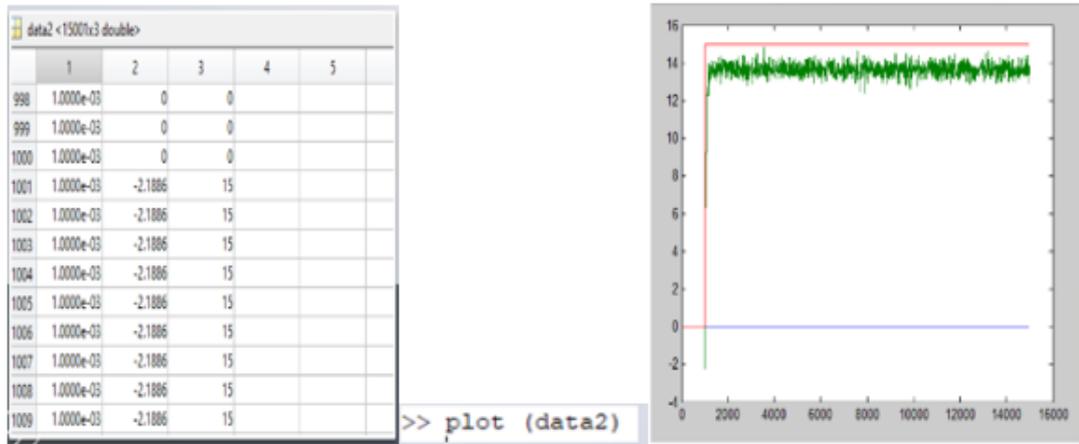


Figure 26: Chargement des données Dans "MATLAB"

2. Prétraitement des données

Nettoyer les données, enlever le bruit, traiter les valeurs manquantes...

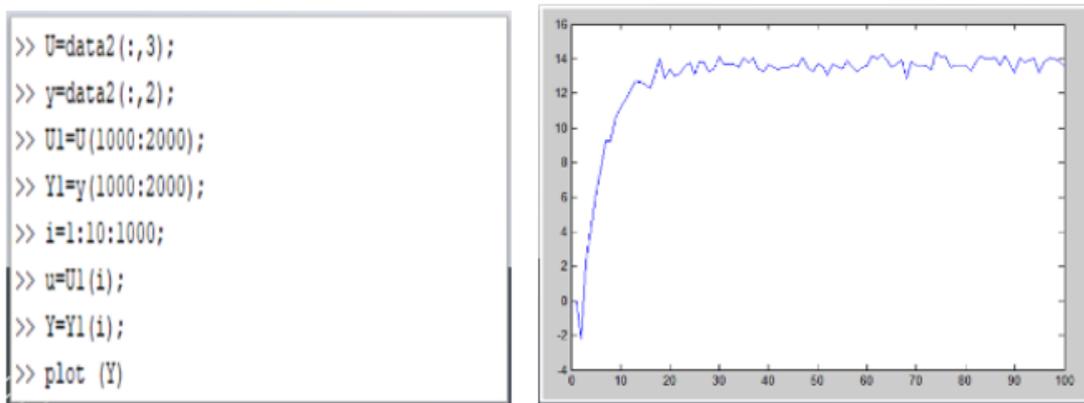


Figure 27: Prétraitement des données

3. Choix du modèle

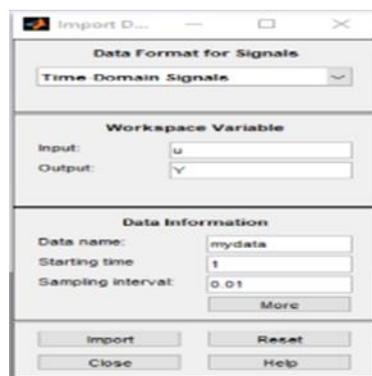


Figure 28: Choix De modèle

4. Estimation des paramètres



Figure 29: Estimations des Paramètres

5. Validation du modèle

Vérifier la qualité du modèle en le testant avec des données non utilisées lors de l'estimation.

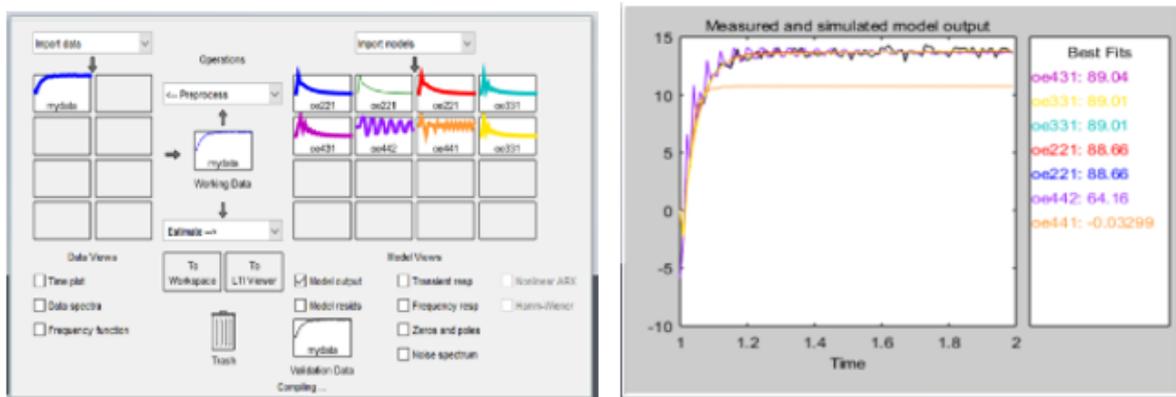


Figure 30: Validation Du modèle

V.4.2 Régulation

1. Modélisation et Simulation du système

- Vérifier la qualité du modèle en le testant avec des données non utilisées lors de l'estimation.

```
>> [num,den]=tfdata(oe321,'v')

num =
    0    0.1860    0.1834   -0.0492

den =
    1.0000    0.0023   -0.6514    0
```

Figure 31: Modélisation du système

- Utilisation MATLAB pour simuler le comportement du système avec le régulateur PID.

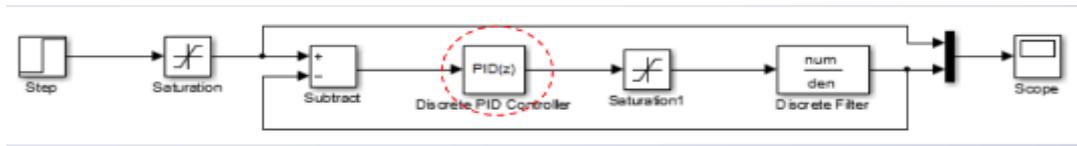


Figure 32: Simulation du système

2. Réglage des paramètres

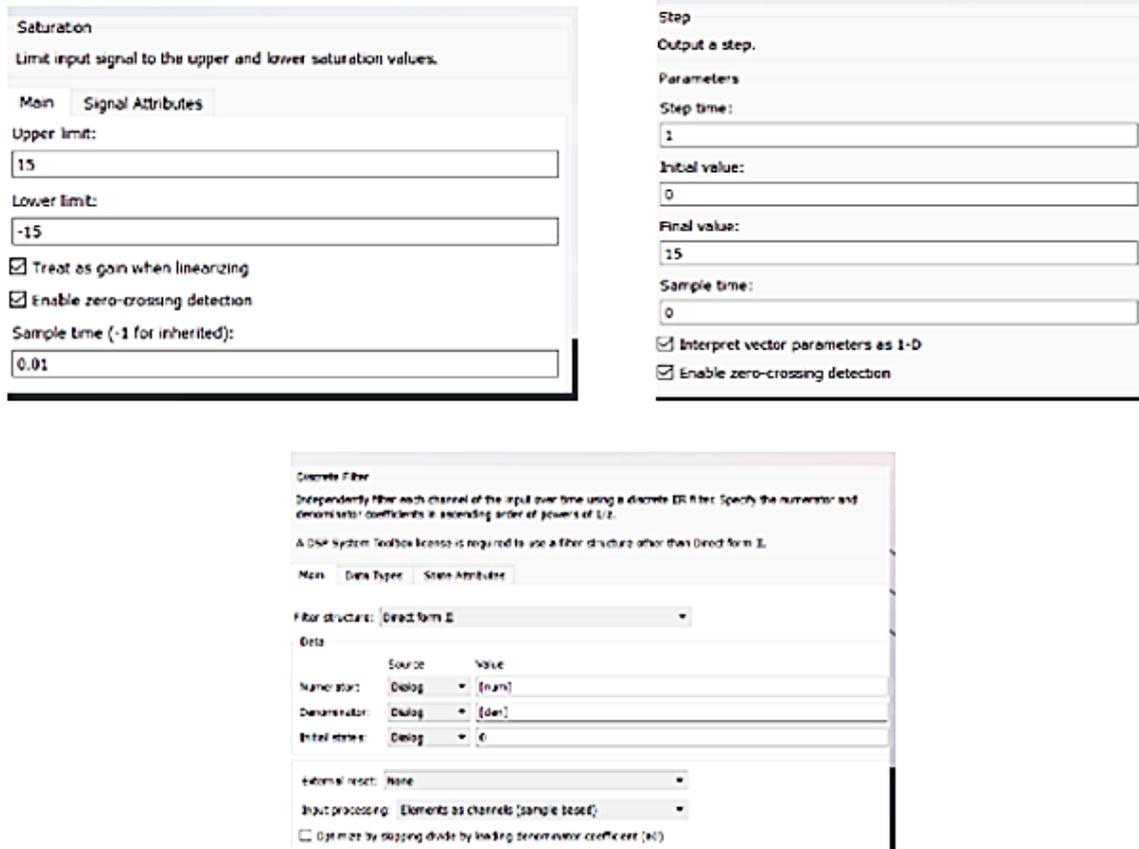


Figure 33: Réglage des Paramètres

Le contrôleur PID

Proportionnel-Intégral-Dérivé, est un mécanisme de régulation utilisé dans les systèmes de contrôle automatique. Il se compose de trois termes principaux :

Proportionnel (P) : Réagit proportionnellement à l'erreur actuelle entre la valeur mesurée et la consigne. Il ajuste la sortie du système en proportion de cette erreur.

Intégral (I) : Accumule les erreurs passées au fil du temps. Cela permet de corriger les erreurs persistantes, même si le système atteint un état stable.

Dérivé (D) : Anticipe les variations futures en réagissant à la vitesse à laquelle l'erreur change. Il aide à prévenir les oscillations excessives et à stabiliser le système.

L'utilité du contrôleur PID réside dans sa capacité à maintenir la sortie d'un système aussi proche que possible de la consigne désirée en ajustant les actions

proportionnelles, intégrale et dérivée. Il est largement utilisé dans divers domaines tels que l'automatisation industrielle, la régulation de température, la robotique, et d'autres applications où la précision du contrôle est cruciale. Le PID offre une approche équilibrée entre réactivité rapide et stabilité du système.

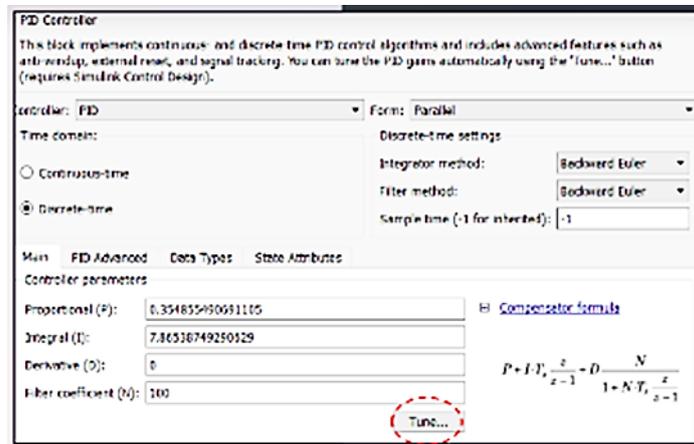


Figure 34: Contrôleur "PID"

3. Surveillance et ajustements continus

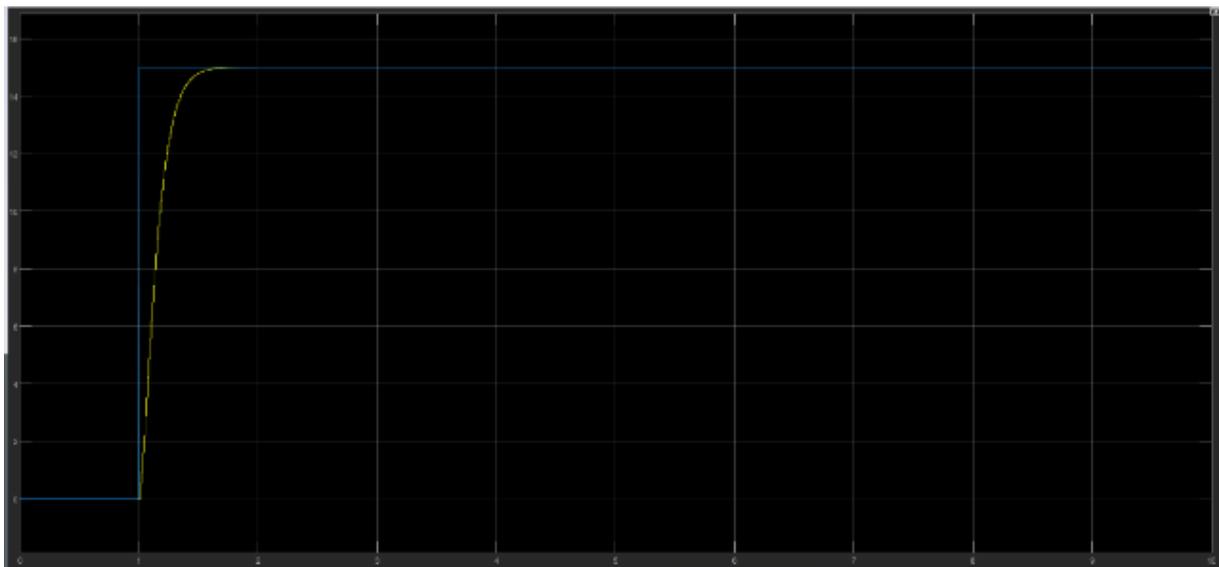


Figure 35: surveillance et ajustements continus

L'utilisation d'un filtre dans le contrôleur PID

Souvent appelé "filtre dérivé", est une technique visant à améliorer la stabilité et la robustesse du contrôle. Ce filtre est appliqué au terme dérivé du PID, Voici quelques utilités de l'ajout de ce filtre :

Réduction du bruit : Le filtre dérivé atténue le bruit élevé fréquent qui peut être amplifié par la dérivée du signal de commande. Cela rend le contrôleur moins sensible aux fluctuations de mesure et aux perturbations.

Élimination des instabilités : Le terme dérivé pur peut conduire à des instabilités, notamment en présence de bruit. Le filtre aide à éliminer les effets indésirables et à éviter les oscillations excessives dans la réponse du système.

Amélioration de la robustesse : En filtrant la dérivée, le contrôleur devient moins sensible aux variations rapides du signal d'erreur, ce qui améliore la robustesse face aux changements brusques ou aux perturbations.

Facilitation de l'accord du PID : L'utilisation du filtre dérivé peut faciliter le processus d'accord du PID en rendant le système moins sensible aux ajustements précis des paramètres.

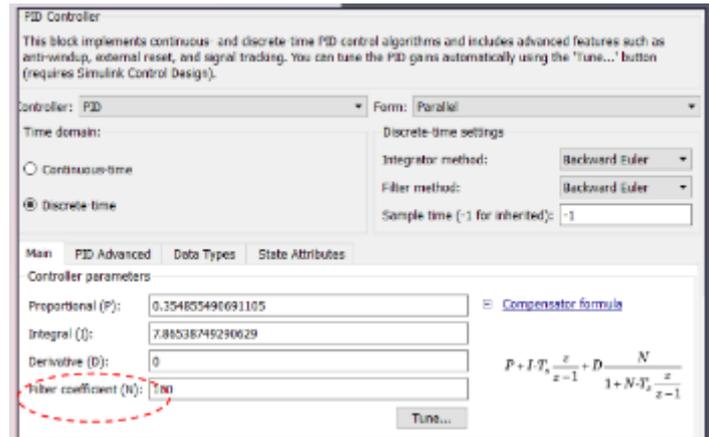


Figure 36: Facilitation De l'accord du "PID"

Ces étapes ont été réalisées par les ingénieurs de l'entreprise et les paramètres PID nous ont été donnés. Ces paramètres seront présentés dans le chapitre simulation.

VI. Conclusion

Dans ce chapitre nous avons présenté l'entreprise de peinture et le processus de fabrication de PVA et les principes de régulations et ses outils ainsi que les étapes d'identification d'un système.

Chapitre 02 :

L'automatisation

I. Introduction

Dans ce chapitre nous présenterons les outils de l'automatisation et principalement les automates programmables en mettant le point sur l'automate S7 1200 qui sera utilisé pour notre application. Nous présenterons aussi l'environnement TIAPORTAL ou le projet sera réalisé et simulé.

III. Définition de système automatisé

Un système est considéré comme "automatisé" lorsqu'il effectue constamment le même cycle de travail pour lequel il a été conçu. Il s'agit d'une combinaison organisée de techniques interconnectées avec des moyens de commande et de contrôle qui lui garantissent un fonctionnement reproductible, plus ou moins autonome de l'intervention humaine [12].

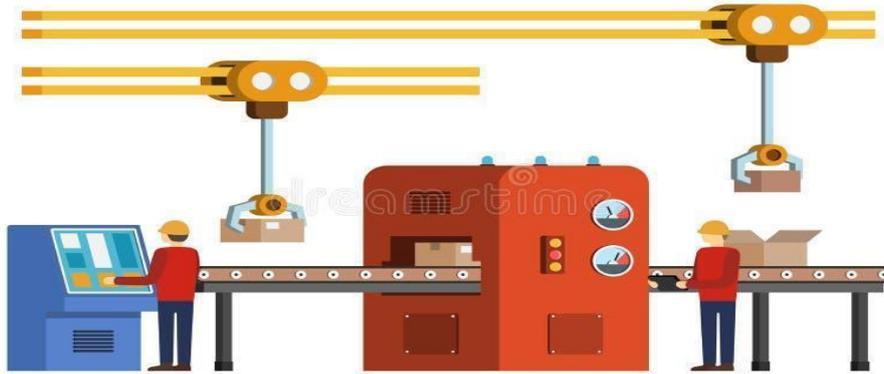


Figure 37: Exemple d'un système automatisé

I.1 Objectif de l'automatisation

- **Dans le but d'améliorer les conditions de travail du personnel** en éliminant les tâches les plus difficiles et en renforçant la sécurité.
- **Dans le but d'améliorer le produit** : Améliorer sa faisabilité, sa conformité aux exigences et sa fiabilité dans le temps.
- **Dans le but d'améliorer la compétitivité de l'entreprise** en réduisant les coûts de production, en augmentant sa productivité, sa qualité de production, ainsi que sa capacité de contrôle, de gestion et de planification [12].

I.2 Avantages et les Inconvénients de l'automatisation

Les avantages de l'automatisation des processus sont multiples mais on peut résumer quelques avantages dans les points suivants :

- La capacité de production accélérée.
- L'aptitude à convenir à tous les milieux de production.
- La souplesse d'utilisation.
- Flexibilité (une machine peut s'adapter à plusieurs productions).

- Confidentialité (une machine ne peut pas parler).
- Un S.A. peut travailler 24h sur 24h.
- Pas de grèves.

Par contre, l'automatisation des processus possède certains inconvénients :

- Le coût élevé du matériel, principalement avec les systèmes hydrauliques.
- La maintenance doit être structurée.
- La suppression d'emplois.
- Incidence sur l'emploi (licenciement –chômage : la mise en place d'une machine se substituant à 10 salariés n'aboutit pas à la création de 10 emplois).
- Investissement pour l'achat de machines.
- Panne.
- Consommation d'énergie.
- Formation d'un personnel plus qualifié (technicien de maintenance, de contrôle... [13]).

I.3 Domaines d'application des systèmes automatisés

De nos jours, la conception d'un système de production est complexe sans l'utilisation des diverses technologies et composants qui constituent les systèmes automatisés, notamment dans les secteurs de l'automobile, de l'aviation, de l'industrie, médical et de transport [13].

I.4 Structure d'un système automatisé

Dès sa conception, un système à automatiser doit être décomposé en trois parties :

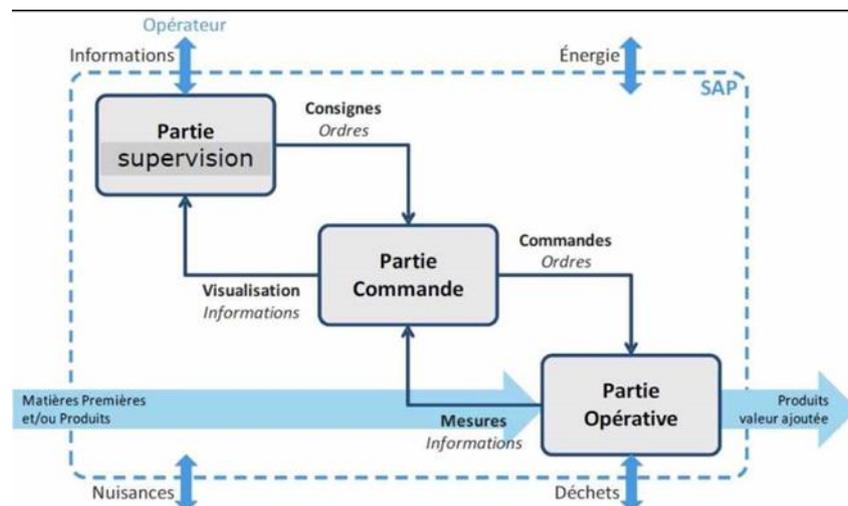


Figure 38: Structure d'un système automatisé

II.4.1 Partie opérative

Appelée parfois « partie puissance », la partie opérative d'un automatisme assure la transformation de la matière d'œuvre

- **La partie mécanique** : chariots, glissières, engrenages, poulies, broches...

- **Les actionneurs** : convertissent l'énergie d'entrée disponible sous une certaine forme (électrique, pneumatique, hydraulique) en une énergie utilisable sous une autre forme, par exemple :
 - Energie thermique destinée à chauffer un four (l'actionneur étant alors une résistance électrique).
 - Energie mécanique destinée à provoquer une translation de chariot (l'actionneur pouvant être un vérin hydraulique ou pneumatique).
 - Energie mécanique destinée à provoquer une rotation de broche (l'actionneur pouvant être alors un moteur électrique).
- **Les préactionneurs** : reçoivent les signaux de commande et réalisent la commutation de puissance avec les actionneurs.

Les préactionneurs des moteurs électriques sont appelés contacteurs.

Les préactionneurs des vérins et des moteurs hydrauliques et pneumatiques sont appelés distributeurs (à commande électrique ou pneumatique).

- **Les capteurs** : qui communiquent à la partie commande des informations sur la position d'un mobile, une vitesse, la présence d'une pièce, une pression...

Les capteurs T.O.R (tout ou rien), qui délivrent un signal de sortie logique, c'est-à-dire 0 ou 1. Exemple : détecteur de fin de course.

Les capteurs numériques ou « incrémentaux », qui associés à un compteur, délivrent des signaux de sortie numérique. Exemple : capteur ou codeur incrémental utilisé pour la mesure des déplacements des chariots de machine à commande numérique.

Les capteurs analogiques ou proportionnels » qui permettent de prendre en compte la valeur réelle d'une grandeur physique. Exemple : Sonde de température.

- **Les appareils de ligne** : Ceux-ci représentent l'ensemble des composants indispensables à la mise en œuvre et à la bonne marche de l'automatisme [12].

II.4.2 La partie commande P.C

Appelée également « partie traitement des informations », elle regroupe tous les composants de traitement des informations nécessaires à la bonne marche de la partie opérative.

La partie commande communique avec l'opérateur par l'intermédiaire d'un pupitre.

Les informations entre la partie commande et la partie opérative passent souvent par l'intermédiaire d'interfaces.

Trois technologies sont actuellement utilisées

- Électromécanique.
- Pneumatique.
- Électronique.

La troisième se présente sous cette forme :

Logique programmée : l'enchaînement des mouvements du système automatisé est programmé sous forme d'instructions (programme), traitées et gérées par l'unité centrale de la partie commande :

- **Les automates programmables industriels (A.P.I.)** : Ils sont munis de bornes d'entrées et sorties.

- **Les micro et mini-ordinateur** : Leur utilisation demande des connaissances en informatique.... Ils ne possèdent pas en général de bornes d'entrées et sorties.
- **Les micros systèmes** : Idem ci-dessus mais possèdent des bornes d'entrées et sorties.

II.4.3 La partie pupitre ou la partie supervision

Le pupitre permet à l'opérateur de dialoguer et de commander la partie opérative. Il comporte :

- **Des capteurs de commande** (marche, arrêt, arrêt d'urgence...).
- **Des voyants de signalisation** (mise sous tension, fonctionnement anormal, buzzer...).
- **Des appareils de mesure** de pression (manomètre), de tension (voltmètre), d'intensité (ampèremètre).

Les interfaces

Elles assurent une compatibilité entre les signaux qui circulent entre la partie commande et la partie opérative. On en distingue deux types :

- Celles qui permettent un changement de niveau d'énergie : relais instantanés, contacteurs auxiliaires...
- Celles qui permettent un changement de type d'énergie : interfaces électropneumatiques, contacts à pression [12].

III. Les automates Programmables Industriels

III.1 Historique Les Automates Programmables Industriels

Les automates programmables industriels sont apparus aux Etats Unis vers les années 1969, essentiellement dans l'industrie automobile américaine (General Motors en leader). Ce n'est qu'en 1971 qu'ils firent leur première apparition en France. Ils sont de plus en plus employés dans l'industrie.

III.2 Définition un automate programmable (API)

API est une machine électronique, programmable par un personnel et destinée à piloter en ambiance industrielle et en temps réel des procédés automatiques.

D'après la norme NFC 63-850 un API est un "Appareil électronique qui comporte une mémoire programmable par un utilisateur automatique (et non informaticien) à l'aide d'un langage adapté, pour le stockage interne des instructions composant les fonctions d'automatisme comme par exemple :

- Logique séquentielle et combinatoire.
- Temporisation, comptage, décomptage, comparaison.
- Calcul arithmétique.
- Réglage, asservissement, régulation, etc., pour commander, mesurer et contrôler au moyen d'entrées et de sorties (logiques, numériques ou analogiques) différentes sortes de machines ou de processus, en environnement industriel [15].

III.3 Principe générale de fonctionnement d'un API

Tous les automates fonctionnent selon le même mode opératoire :

- **Traitement interne**

L'automate effectue des opérations de contrôles et met à jour certains paramètres systèmes (détection des passages en RUN/STOP, mises à jour des valeurs de l'horodateur,).

- **Lecture des entrées**

L'automate lit les entrées (de façon synchrone) et les recopies dans la mémoire image des entrées.

- **Exécution du programme**

L'automate exécute le programme instruction par instruction et écrit les sorties dans la mémoire image des sorties.

- **Ecriture des sorties**

L'automate bascule les différentes sorties (de façon synchrone) aux positions définies dans la mémoire image des sorties. Ces quatre opérations sont effectuées continuellement par l'automate (fonctionnement cyclique) [13].



Figure 39: fonctionnement d'un API

III.4 Architecture d'un API

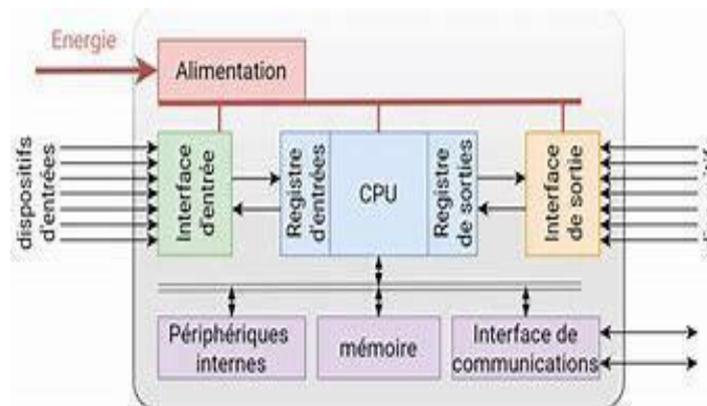


Figure 40: Architecture d'un API

III.4.1 Aspect extérieur

Les automates peuvent être de type compact ou modulaire.

III.4.1.1 Compacte

Pour la configuration compacte, on distingue les modules de programmation (LOGO de Siemens, ZELIO de Schneider, MILLENIUM de Crouzet...etc.) des micro-automates. Il intègre le processeur, l'alimentation, les entrées et les sorties. Selon les modèles et les fabricants, il pourra réaliser certaines fonctions supplémentaires (comptage rapide, E/S analogiques ...) et recevoir des extensions en nombre limité. Ces automates, de fonctionnement simple, sont généralement destinés à la commande de petits automatismes [13].



Figure 41: API compacte

III.4.1.2 Modulaire

Dans ce modèle le processeur, l'alimentation et les interfaces entrées/sorties résident dans des unités séparées (modules). Ces automates sont intégrés dans les automatismes complexes de grande puissance et de capacité de traitement [13].

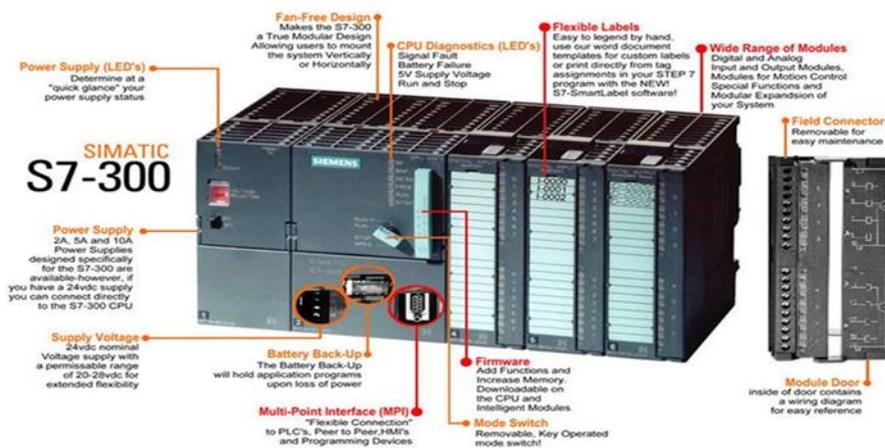


Figure 42: API modulaire

III.4.2 Structure interne

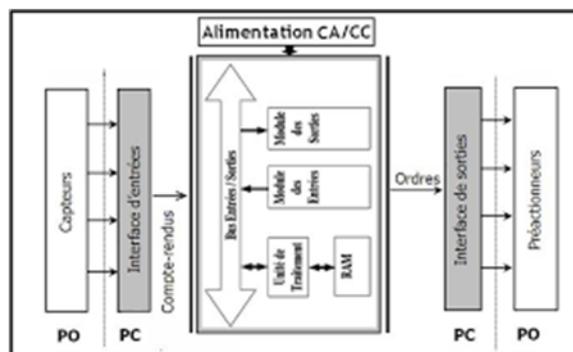


Figure 43: Structure interne de l'API

Un automate programmable industriel est donc constitué des éléments suivants :

III.4.2.1 Un module d'alimentation

Le module d'alimentation transforme l'énergie externe provenant du réseau en la mettant en forme afin de fournir aux différents modules de l'API les niveaux de tension nécessaires à leur bon fonctionnement. Plusieurs niveaux de tension peuvent être utilisés par les circuits internes (3V, 5V, 12V, 24V...). Il sera dimensionné en fonction des consommations des différentes parties.

III.4.2.2 Unité de traitement ou processeur

Le processeur gère l'ensemble des échanges informationnels en assurant :

- La lecture des informations d'entrée.
- L'exécution des instructions du programme mis en mémoire.
- La commande ou l'écriture des sorties.

III.4.2.3 Mémoire programme

La mémoire programme de type RAM contient les instructions à exécuter par le processeur afin de déterminer les ordres à envoyer aux pré actionneurs reliés à l'interface de sortie en fonction des informations recueillies par les capteurs reliés à l'interface d'entrée.

III.4.2.4 Mémoire de données

La mémoire de donnée permet le stockage de :

- L'image des entrées reliées à l'interface d'entrée.
- L'état des sorties élaborées par le processeur.
- Les valeurs internes utilisées par le programme (résultats de calculs, états intermédiaires...).
- Les états forcés ou non des E/S [13].

III.4.2.5 Interface d'entrée

L'interface d'entrée permet la connexion à l'API d'un multiple de capteurs pouvant être de type :

- TOR (logiques ou Tout Ou Rien).
- Numériques.
- Analogiques.

Ces différentes entrées sont mises en forme par l'interface 'entrée avant d'être stockées dans la mémoire de données [13].

III.4.2.6 Une interface de sortie

L'interface de sortie permet la connexion de l'API aux de pré actionneurs, qui peuvent être de type :

- TOR (logiques ou Tout Ou Rien).
- Numériques.
- Analogiques [13].

III.5 Présentation de quelques gammes SIMATIC

Il existe plusieurs gammes de SIMATIC. On trouve le SIMATIC S7 et le SIMATIC M7 :

SIMATIC S7

Dans la gamme S7 on distingue cinq grandes familles d'automates programmables industriels décrites dans ces paragraphes qui suivent.

SIMATIC S7-200

La famille S7-200 est constituée de micro-automates programmables utilisables dans des applications d'automatisations variées. La Figure ci-dessous présente un micro automate S7 -200. Son design compact, ses possibilités d'expansion, son faible prix et son important jeu d'opérations en font une solution idéale pour la commande de petites applications. En outre, le large choix de tailles et de tensions de CPU offre la souplesse nécessaire pour résoudre un problème d'automatisation. Un automate programmable S7-200 consiste en un CPU S7-200 seul ou complété de divers modules d'extension facultatifs connectés à cette dernière à l'aide d'un connecteur de bus fourni avec ce module d'extension [16].



Figure 44: S7-200

SIMATIC S7-300

La famille S7-300 est constituée d'automates programmables de conception modulaire utilisés pour des automatismes de gamme, et peuvent être connectés entre eux au moyen d'un câble-bus PROFIBUS. Un automate S7-300 consiste en un CPU, un module d'alimentation PS, un module de signaux FM, un module de signaux SM et un processeur de communication, comme indiqué dans la figure suivante [16].



Figure 45: S7-300

SIMATIC S7-400

La famille S7-400 est aussi constituée d'automates programmables de conception modulaire. Pratiquement chaque tâche d'automatisation peut être résolue par un choix approprié des constituants de S7-400 et avec la possibilité d'expansion de plusieurs modules. Les modules se présentent sous forme de boîtiers que l'on adapte sur un châssis [16].



Figure 46: S7-400

SIMATIC S7-1200

L'automate SIMATIC S7-1200 est modulaire et compact, polyvalent et constitue une solution parfaitement adaptée à une grande variété d'applications. Une conception modulaire et flexible, une interface de communication répondant aux exigences les plus sévères dans l'industrie et une large gamme de fonctions technologiques performantes et intégrées faisant de lui une solution d'automatisation complète [17].

Cet automate sera utilisé dans notre projet car il est disponible à notre laboratoire d'Informatique Industrielle SIEMENS.



Figure 47: S7-1200

SIMATIC S7-1500

Le SIMATIC S7-1500 est le système de commande modulaire pour une multitude d'applications dans le secteur de l'automatisation discrète. Il dispose d'une vaste gamme de modules combinables individuellement [18].



Figure 48: S7-1500

III.6 Description de l'Automate S7-1200

Dans le laboratoire de notre institut les étudiants se familiarisent avec l'automate S7-1200.

Le contrôleur S7-1200 offre la souplesse et la puissance nécessaires pour commander une large gamme d'appareils afin de répondre aux besoins en matière d'automatisation. Sa forme compacte, sa configuration souple et son important jeu d'instructions en font une solution idéale pour la commande d'applications très variées [19].

Le CPU combine un microprocesseur, une alimentation intégrée, des circuits d'entrée et de sortie, un PROFINET intégré, des E/S rapides de commande de mouvement, ainsi que des entrées analogiques intégrées dans un boîtier compact en vue de créer un contrôleur puissant.

Une fois le programme chargé, le CPU contient la logique nécessaire au contrôle et à la commande des appareils dans votre application. Le CPU surveille les entrées et modifie les sorties conformément à la logique de votre programme utilisateur, qui peut contenir des instructions booléennes, des instructions de comptage, des instructions de temporisation, des instructions mathématiques complexes ainsi que des commandes pour communiquer avec d'autres appareils intelligents [20].

Le CPU fournit un port PROFINET permettant de communiquer par le biais d'un réseau PROFINET. Des modules supplémentaires sont disponibles pour communiquer via les Réseaux PROFIBUS, GPRS, RS485 ou RS232. Pour ce projet, le S7-1200 est programmé en l'aide du logiciel TIA PORTAL sous Windows.

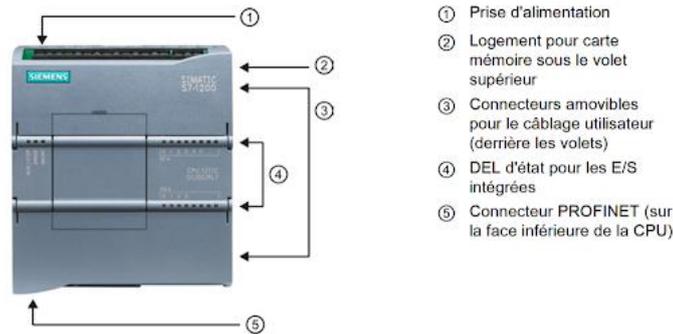


Figure 49: Automate S7-1200- Siemens

III.6.1 Choix de la CPU S7-1200

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 E/S, TOR ou analogiques supplémentaires sans modification de l'encombrement de l'automate. Des modules d'E/S supplémentaires peuvent être ajoutés du côté droit de la CPU pour étendre la capacité d'E/S, TOR ou analogiques [21].

III.6.2 Le choix des modules d'Entrées/Sorties

Le choix des modules E/ S est basé sur les critères suivants :

- Le type et la valeur de la tension d'entrée ou de sortie.
- Le nombre de voies.
- Le type d'entrée ou de sortie (sonde, thermocouple, électrovanne...).

Possibilités d'extension du CPU La gamme S7-1200

Offre divers modules et cartes enfichables pour accroître les capacités du CPU avec des E/S supplémentaires ou d'autres protocoles de communication [21].

III.7 Nature des informations traitées par l'automate

Toutes les informations peuvent être de trois types :

- I. **Tout ou rien (T.O.R.)** : l'information ne peut prendre que deux états (vrai/faux, 0 ou 1).
C'est le type d'information délivrée par un détecteur, un bouton poussoir ...
- II. **Analogique** : l'information est continue et peut prendre une valeur comprise dans une Plage bien déterminée. C'est le type d'information délivrée par un capteur (pression, température ...)
- III. **Logique** : l'information est contenue dans des mots codés sous forme binaire ou bien hexadécimale. C'est le type d'information délivrée par un ordinateur ou un module intelligent [23].

III.8 Les notions de choix d'un automate programmable (API)

- Le nombre d'entrées/sorties.
- La nature des entrées/sorties (numérique, analogique, etc....).

- La nature du traitement (temporisation, comptage, etc....).
- Le dialogue (la console détermine le langage de programmation).
- La communication avec les autres systèmes.
- Les moyens de sauvegarde du programme (Disquette, carte mémoire, etc....).
- La fiabilité et la robustesse [23].

III.9 Domaines d'emploi des automates

On utilise les API dans tous les secteurs industriels pour la commande des machines (convoyage, emballage ...etc.) ou des chaînes de production (automobile, agroalimentaire ...etc.) où il peut également assurer des fonctions de régulation de processus (métallurgie, chimie ...etc.). Il est de plus en plus utilisé dans le domaine du bâtiment (tertiaire et industriel) pour le contrôle du chauffage, de l'éclairage, de la sécurité ou des alarmes, aussi on les trouve dans les systèmes de transport [24].

III.10 La communication et les Réseaux industriels

L'automate programmable industriel doit être connecté à d'autres matériels à processeur, et afin de pouvoir dialoguer avec les agents d'exploitation il faut donc se pencher sur ses liens avec son environnement et les fonctions qu'il doit assurer. Il existe plusieurs bus de terrain industriel qui sont universels ou limités (le bus est propre à l'entreprise), on note quelques protocoles de communication : Modbus, BusCAN, uni-Telway, CANopen, Profibus, Profinet, MPI, ...ect [25].

III.10.1 Le Profibus

Profibus (Processus Field Bus) est le bus de terrain ouvert qui connaît un grand succès dans un large domaine d'application pour une production efficace et économique. Profibus est un réseau en bus pour la communication de processus et de terrain dans les réseaux de cellules mettant en œuvre un petit nombre de stations et des appareils de terrain ainsi que pour la communication de données conforme à CEI 61158/EN 50 170 [25].

III.10.2 Profinet

Profinet est le standard Industriel Ethernet ouvert de PROFIBUS International pour l'automatisation industrielle. PROFINET permet de coupler des appareils du niveau terrain jusqu'au niveau conduite. Il garantit une communication fluide, autorise une ingénierie à l'échelle de l'installation et utilise les standards IT jusqu'au niveau terrain.

III.11 Les avantages et les inconvénients

III.11.1 Avantage des API

- Améliorer les conditions de travail en éliminant les travaux répétitifs.
- Améliorer la productivité en augmentant la production.
- Programmés facilement (logique programmé) alors la Modification du programme facile par rapport à la logique câblée.
- Simplification du câblage.
- Puissance et rapidité.

- Facilité de maintenance (l'API par lui-même est relativement fiable et peut aider l'homme dans sa recherche de défauts).
- Augmenter la sécurité.
- Possibilités de communication avec l'extérieur (ordinateur, autre API)
- Plus économique.

III.11.2 Inconvénients des API

- Plantage.
- Il y a trop de travail requis dans les fils de connexion.
- Besoin de formation.

III.12 Les langages de programmation d'un API

Les langages destinés à la programmation des API sont pour objet d'être facilement mis en œuvre par tout technicien après une courte formation. L'écriture d'un programme consiste à créer une liste d'instructions permettant l'exécution des opérations nécessaires au fonctionnement du système. Actuellement les API disposent en tout ou partie des langages de programmation suivants :

- **Langage List IL (Instruction List)** : Le langage List est très proche de l'assembleur, il utilise la totalité des fonctions de l'API. Tous les réseaux de programmation, définis dans un autre langage, sont compilés dans ce langage [6][3].
- **Langage FBD (Function Bloc Diagram)** : C'est un langage qui permet de programmer graphiquement à l'aide de blocs programmes ou programmables, reliés entre eux et réalisant tout type de fonctions [6][3].

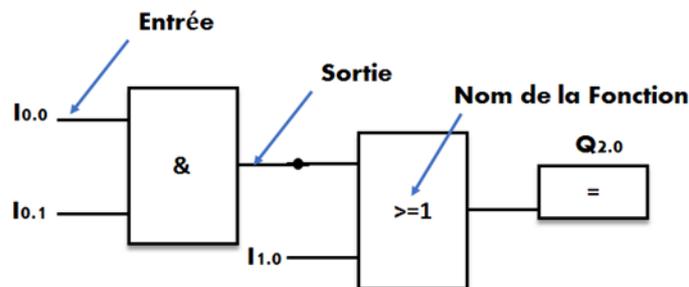


Figure 50: Langage FBD

- **Langage littéral structuré ST (Structured Text)** : Appelé aussi SCF (Structured Control Language), ressemble beaucoup au langage C. C'est un langage structuré qui convient bien pour les applications faisant appel à des calculs compliqués et au traitement des chaînes de caractère, Il permet la programmation de tout type d'algorithme plus ou moins complexe [27][26].
- **Langage GRAFCET** : Graphe Fonctionnel de Commande des Etapes et Transitions ou Séquentiel Fonction hart (SFC), est un outil graphique qui décrit les différents comportements de l'évolution d'un automatisme. C'est un mode de représentation

et d'analyse d'un automatisme, particulièrement bien adapté aux systèmes à évolution séquentielle, c'est-à-dire décomposable en étapes [27][28].

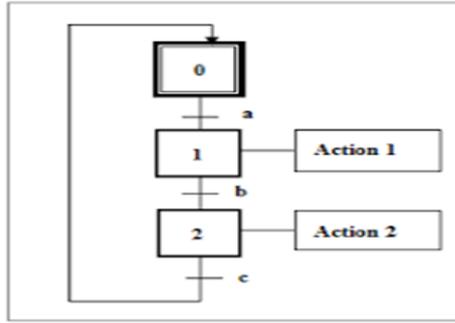


Figure 51: Langage GRAFCET

- **Langage à Contact LD (Ladder Diagram) :** Le langage LD est le plus utilisé, il ressemble aux schémas électriques à contacts, c'est un langage graphique avec une approche aisée et visuelle du problème dédié à la programmation d'équations booléennes (true/false) [27][26].

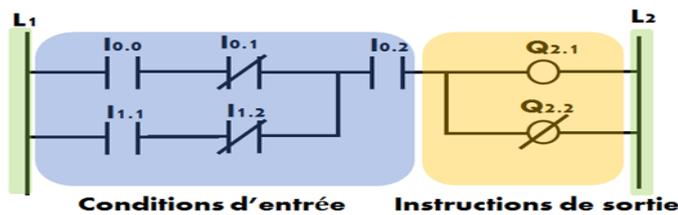


Figure 52: Langage à Contact LD

III.13 Principaux constructeurs d'API

Les principaux constructeurs d'API sont allemands, américain, français, japonais et suédois. A titre Exemple: SIEMENS, SCHNEIDER, ABB, ALLEN BRADLEY, BOSHRXROTH, FESTO, GEFANUC, HONEWELL, VIPA, OMRON, ELECTRIC, MITSUBISHI.

IV. L'Environnement de developpement « TIA Portal V16 »

VI.1 Présentation du logiciel

La plate-forme **TIA** (Totally Integrated Automation) Portal est la dernière évolution des logiciels de travail Siemens qui permet de mettre en œuvre des solutions d'automatisation avec un système d'ingénierie intégré, dans un seul logiciel cette plateforme regroupe la programmation des différents dispositifs d'une installation. On peut donc programmer et configurer, en plus de l'automate, les dispositifs HMI, les variateurs... etc [30].

VI.2 Les avantages du logiciel

- La programmation est simple et rapide grâce aux nouveaux éditeurs de programmation : SCL, CONT, LOG, LIST et GRAPH.
- Augmentation d'efficacité grâce aux innovations linguistiques de TEP7 : programmation symbolique unifiée, arithmétique de boîtes, ajout de blocs pendant l'exécution, et bien plus encore.
- Des performances accrues grâce à des fonctions intégrées : simulation avec PLCSIM, télémaintenance avec Téléservice et système de diagnostic cohérent.
- Technologie flexible : La fonction de commande de mouvement évolutive et efficace des PLC S7-1500 et S7-1200.
- Avec Security Integrated, vous bénéficiez d'une sécurité complète : protection de la propriété intellectuelle, protection contre la copie, protection de l'accès et protection contre la falsification [30].

VI.3 STEP 7 sur TIA Portal

STEP 7, qui est intégré au TIA Portal, est le logiciel utilisé pour configurer, programmer, tester et diagnostiquer tous les contrôleurs STEP 7 est doté d'une multitude de fonctionnalités conviviales qui permettent d'accroître considérablement la productivité de toutes les opérations d'automatisation, notamment la programmation, la simulation, la mise en service et la maintenance [29].

VI.4 S7-PLCSIM V16

S7-PLCSIM prend en charge le débogage et la validation d'un programme automate unique sans nécessiter de matériel réel.

S7-PLCSIM vous permet d'utiliser tous les outils de débogage de STEP 7, y compris la table de surveillance, l'état du programme, les fonctions en ligne et de diagnostic, ainsi que d'autres outils. S7-PLCSIM fournit également des outils qui lui sont propres, notamment une table SIM, un éditeur de séquence, un éditeur d'événements et un contrôle de balayage.

S7-PLCSIM fonctionne en conjonction avec la programmation STEP 7 dans le TIA PORTAL.

Vous utilisez STEP 7 pour effectuer les tâches suivantes :

- Configurez votre automate et tous les modules associés dans STEP 7.
- Programmez votre logique d'application.
- Téléchargez la configuration matérielle et le programme dans S7-PLCSIM en mode Compact ou Project [29].

VI.5 Logiciel de supervision WinCC

WinCC (Windows Control Center), est le logiciel qui permet de créer une Interface Homme Machine (IHM) graphique, qui assure la visualisation et le diagnostic du procédé. Il permet la saisie, l'affichage et l'archivage des données, tout en facilitant les tâches de conduite et de surveillance aux exploitants. Il offre une bonne solution de supervision, car il met à la disposition de l'opérateur des fonctionnalités adaptées aux exigences d'une installation industrielle [14].

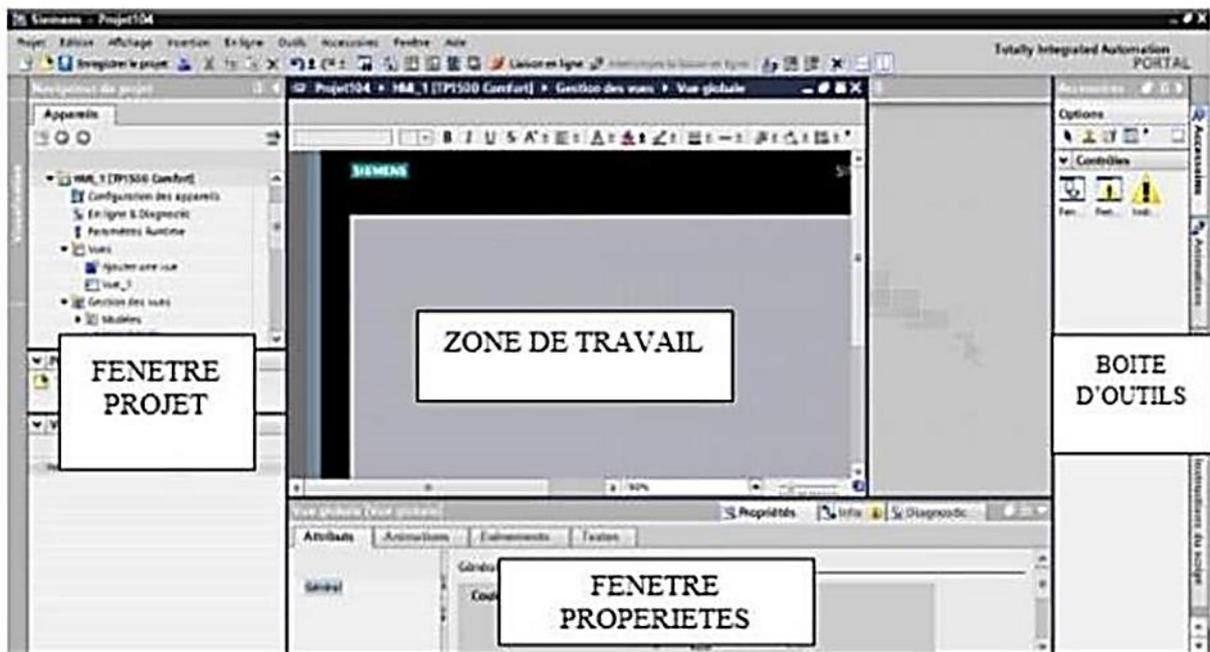


Figure 53: Fenêtre du travail dans le WinCC

Nous distinguons sur cette figure :

- La zone de travail
- La boîte d'outils
- La fenêtre de projet
- La fenêtre des propriétés
- **La zone de travail** : C'est dans cette zone où se fait la construction des différentes vues du projet.
- **La boîte d'outils** : Cette zone nous offre la possibilité d'importer les éléments de base nécessaires pour la création des vues (bouton, champ graphique, champ de texte, ...etc.).
- **La fenêtre de projet** : Elle affiche la structure du projet, on peut à partir de cette zone créer des vues, des variables configurées et des alarmes.
- **La fenêtre des propriétés** : Elle permet de charger ou de modifier les propriétés d'un objet sélectionné dans la zone de travail.

VI.5.1 WinCC runtime advanced V16

WinCC runtime advanced V16 est une interface Homme-Machine (IHM) pour les applications de machines et de processus dans la construction d'installations. Il nous permet de visualiser le processus, facilitant la surveillance par des graphiques d'écran, par exemple, l'utilisateur peut démarrer ou arrêter le moteur, et l'écran affiche un message s'il y a des alarmes ou des erreurs [29].

Application Runtime

L'application Runtime permet à l'opérateur de s'assurer que le processus est exécuté et surveillé en temps réel, et les tâches requises lors de l'exécution sont :

- Lecteur de données et vue d'affichage.
- Communication avec le PLC.
- Contrôle de processus, démarrage/arrêt.
- Surveillance des processus à distance.
- Détection d'alarme [29].

VI.6 La relation entre TIA portal et Wincc

Wincc est intégré dans TIA Portal afin d'offrir le même environnement de travail pour la figuration des postes IHM et programmation des postes automates cela signifie des symboles et une gestion commune. L'interface d'ingénierie a la même apparence pour les stations IHM et API et il n'y a plus de distinction entre projets IHM et PLC. Ces stations sont créées dans le même projet et peuvent figurer dans le même réseau avec les moyens de liaisons et de communication. Les liaisons suivantes permettent la communication entre WinCC et les différents automates programmables :

- Interface MPI (Protocole S7).
- PROFIBUS.
- Industrial Ethernet.
- TCP/IP.
- SLOT PLC.
- S7-PMC PROFIBUS [32].

VI.7 La supervision industrielle

La supervision est une technique industrielle de suivi et de pilotage informatique de procédés de fabrication automatisés. Elle concerne l'acquisition de données (mesures, alarmes, retour d'état de fonctionnement) et des paramètres de commande des processus généralement confiés à des automates programmables.

Dans l'informatique, la supervision est la surveillance du bon fonctionnement d'un système ou d'une activité.

En informatique industrielle, la supervision des procédés est un pupitre de commande évolué. Elle permet de surveiller ou de contrôler l'exécution de tâches du procédé.

VI.7.1 Rôle de la supervision

Deux phases sont importantes pour que les administrateurs soient capables d'atteindre l'objectif voulu par la supervision, à savoir, - surveiller le système et garantir sa disponibilité même en cas d'anomalie.

- Tenter de prévenir en cas de problème (défaillances matérielles ou interruption des services) et garantir une remontée d'information rapide.
- Automatiser les tâches de récupération des applications et des services en assurant des mécanismes de redondance en une durée d'intervention minimale (par exemple : le redémarrage des services interrompus, l'arrêt de la machine en cas de la surcharge du CPU, la sauvegarde des données en cas du risque de perte d'un disque dur en miroir, etc.).

VI.7.2 Définition de SCADA

SCADA (Supervisory Control And Data Acquisition) est synonyme de contrôle de surveillance et d'acquisition de données. Comme son nom l'indique, désigne une catégorie de logiciels destinés au contrôle de processus et à la collecte de données en temps réel [33].



Figure 54: Système SCADA

VI.7.3 Définition IHM

Interface **Homme Machine IHM** est un dispositif qui permet le dialogue entre l'opérateur et la machine. Une IHM réalise deux fonctionnalités : présenter des informations à l'opérateur et permettre à l'opérateur d'agir sur le système en introduisant des commandes ou des consignes. Les règles de conception d'une IHM de point de vue du respect des caractéristiques physiques et psychophysiologiques sont :

- Lisibilité suffisante, pas d'informations fournies uniquement sur la base d'un changement de couleur.
- Signaux sonores de fréquence suffisamment basse (inférieure à 800 Hz),
- Les informations le plus fréquemment consultées doivent se trouver dans les zones le plus souvent parcourues par l'œil [33].



Figure 55: IHM

VI.7.4 Choix de l'interface homme-machine

Il s'agit d'un pupitre écran tactile SIEMENS de référence 6AV2-124-0MC01-0AX0 SIMATIC TP 1200 COMFORT pupitre 12 TOUCH screen panel.

Les unités multi TP1200 COMFORT appartiennent à la catégorie de produits <<< Plateforme multifonctionnelle ». Les **IHM** offrent des options de communication améliorées pour le monde du bureau. Le TP1200 COMFORT offre une grande variété d'applications, de hautes performances et un rapport coût/performance favorable.

Les appareils sont équipés de :

- Un port PROFIBUS
- Un port ETHERNET pour la connexion à PROFINET
- 2 ports USB
- Un écran TFT avec jusqu'à 64K couleurs.

VI.8 Conception d'un programme avec TIA PORTAL V13 :

La stratégie à suivre pour faire la conception d'un programme en utilisant la plateforme TIA PORTAL V13 est :

- La création d'un nouveau projet.
- La configuration matérielle.
- La compilation et le chargement de la configuration.
- La création de la table des mnémoniques.
- L'élaboration du programme.
- La simulation avec le logiciel.
- La visualisation d'état du programme (le test) [33].

La conception d'une solution d'automatisation se fait par deux alternatives, soit on commence par la programmation ou par la configuration matérielle, dans notre cas on a commencé par la configuration [33].

VI.8.1 Vue du portail et vue du projet

Lorsque l'on lance TIA Portal, l'environnement de travail se décompose en deux types de vue :

- **La vue du portail** : elle est axée sur les tâches à exécuter et sa prise en main est très rapide.
- **La vue du projet** : elle comporte une arborescence avec les différents éléments du projet. Les éditeurs requis s'ouvrent en fonction des tâches à réaliser. Données, paramètres et éditeurs peuvent être visualisés dans une seule et même vue.

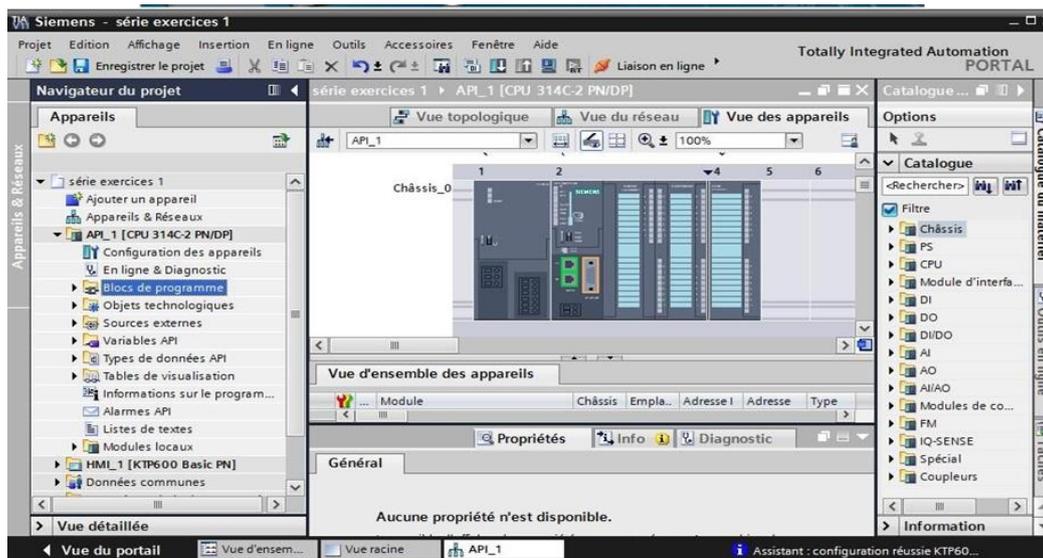


Figure 56: La vue de Projet

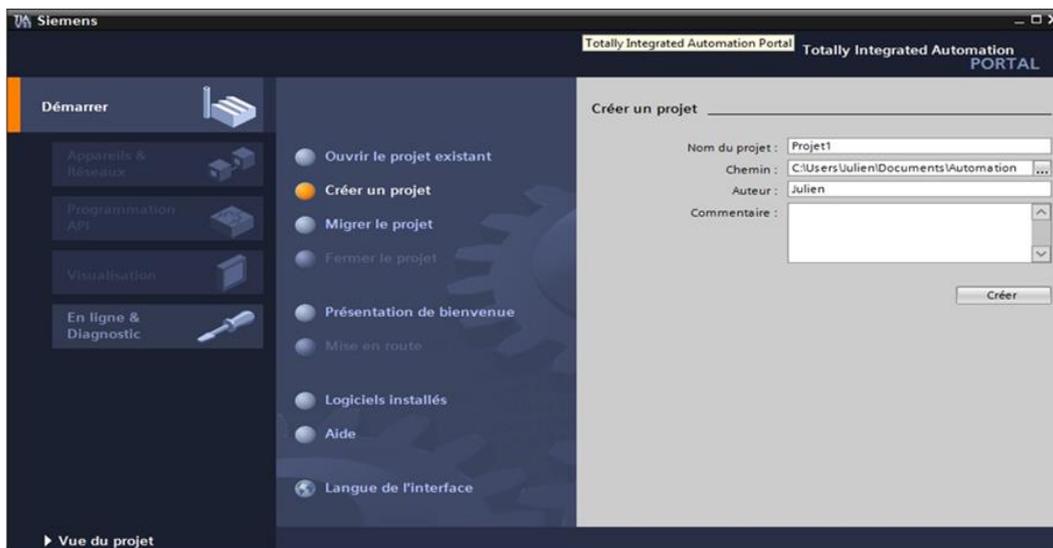


Figure 57: Vue du portail

I. Vue du portail

Chaque portail permet de traiter une catégorie de tâche (actions). La fenêtre affiche la liste des actions pouvant être réalisées pour la tâche sélectionnée.



Figure 58: Vue du portail

II. Vue du projet

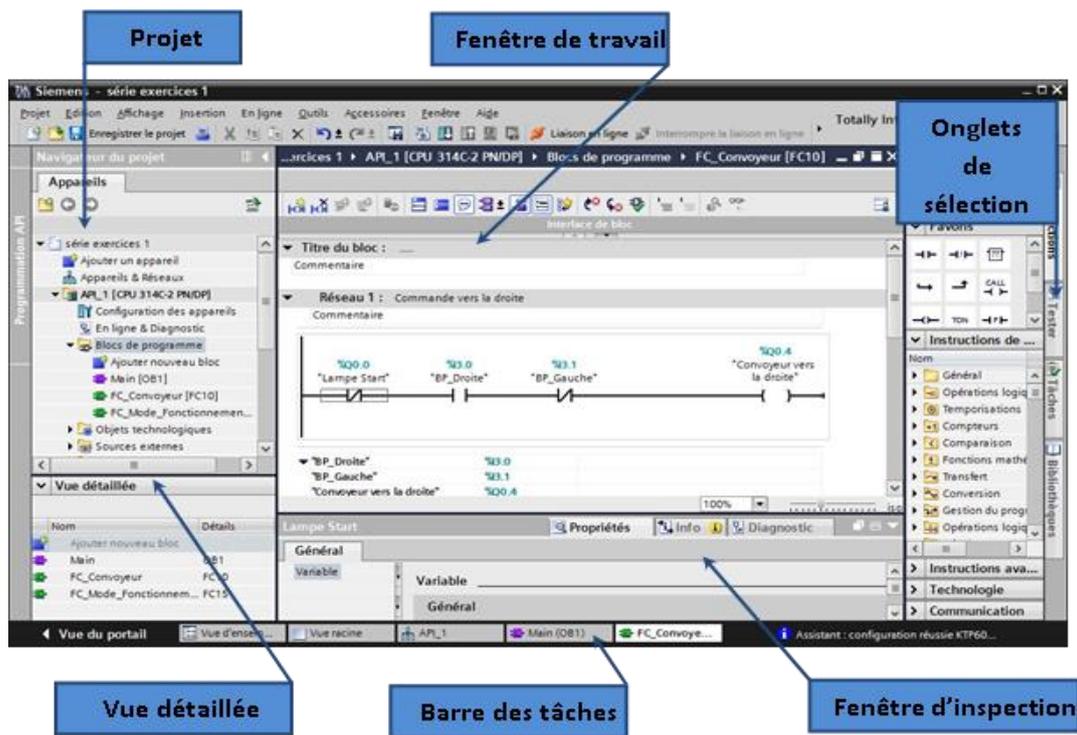


Figure 59: Vue du portail

La **fenêtre de travail** permet de visualiser les objets sélectionnés dans le projet pour être traités. Il peut s'agir des composants matériels, des blocs de programme, des tables des variables, des HMI, ...

La **fenêtre d'inspection** permet de visualiser des informations complémentaires sur un objet sélectionné ou sur les actions en cours d'exécution (propriété du matériel sélectionné, messages d'erreurs lors de la compilation des blocs de programme...). Les **onglets de sélection de tâches** ont un contenu qui varie en fonction de l'objet sélectionné (configuration matérielle, bibliothèques des composants, bloc de programme, instructions de programmation).

Cet environnement de travail contient énormément de données. Il est possible de masquer ou réduire certaines de ces fenêtres lorsque l'on ne les utilise pas. Il est également possible de redimensionner, réorganiser, désancrer les différentes fenêtres [33].

VI.8.2 Création d'un projet

Pour créer un projet dans la vue du portail, il faut sélectionner l'action « **Créer un projet** ».

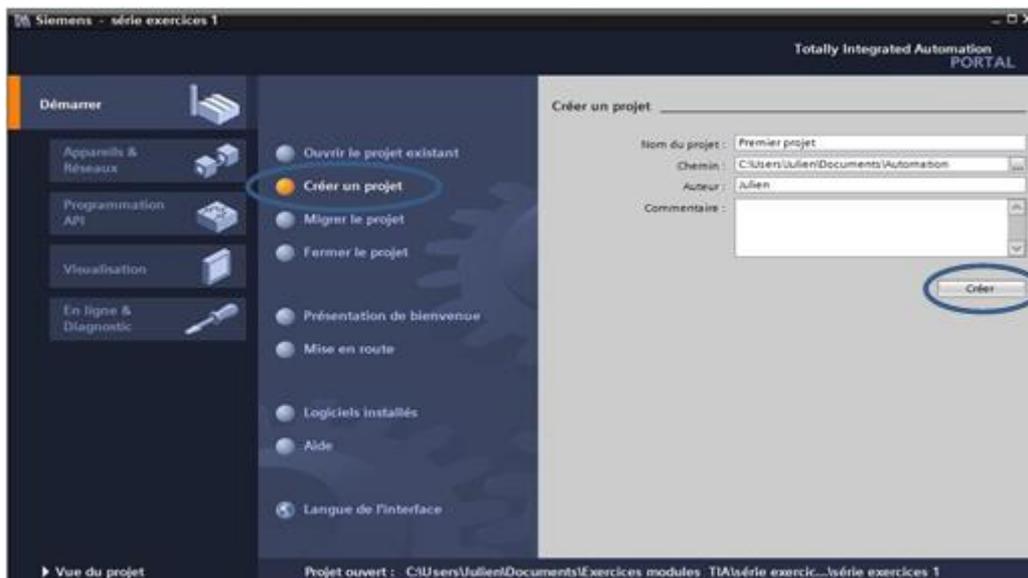


Figure 60: Création d'un projet

On peut donner un nom au projet, choisir un chemin où il sera enregistré, indiquer un commentaire ou encore définir l'auteur du projet.

Une fois que ces informations sont entrées, il suffit de cliquer sur le bouton « **créer** »

VI.9 Configuration et paramétrage du matériel

Une fois votre projet créé, on peut configurer la station de travail.

La première étape consiste à définir le matériel existant. Pour cela, on peut passer par la **vue du projet** et cliquer sur « **ajouter un appareil** » dans le navigateur du projet.

La liste des éléments que l'on peut ajouter apparaît (API, HMI, système PC). On commencera par faire le choix de notre CPU pour ensuite venir ajouter les modules complémentaires (alimentation, E/S TOR ou analogiques, module de communication AS-i, ...).

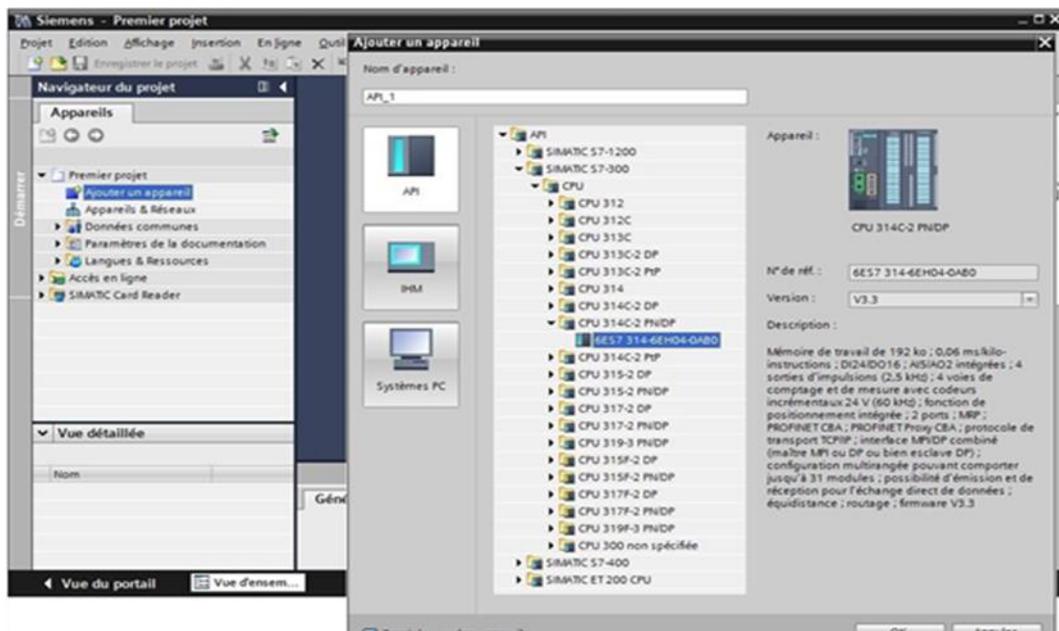


Figure 61: Configuration et paramétrage du matériel

Les modules complémentaires de l'API peuvent être ajoutés en utilisant le catalogue. Si l'on veut ajouter un écran ou un autre API, il faut repasser par la commande « ajouter un appareil » dans le navigateur du projet.

Lorsque l'on sélectionne un élément à insérer dans le projet, une description est proposée dans l'onglet information.

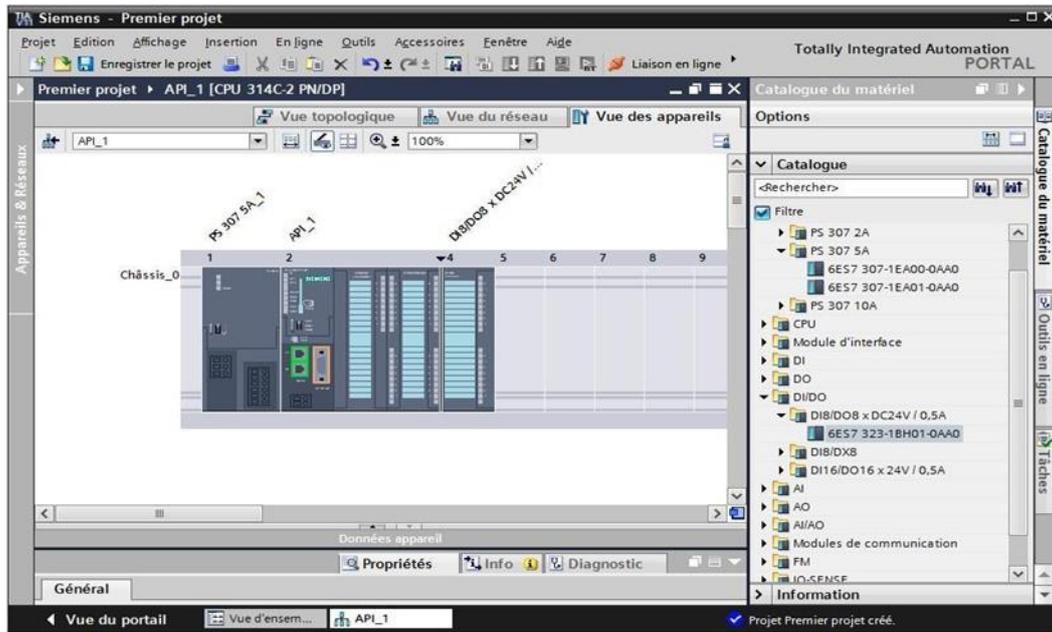


Figure 62: Configuration et paramétrage du matériel

VI.10 Adressage des E/S

Pour connaître l'adressage des entrées et sorties présentes dans la configuration matérielle, il faut aller dans « appareil et réseau » dans le navigateur du projet.

Dans la fenêtre de travail, on doit s'assurer d'être dans l'onglet « Vue des appareils » et de sélectionner l'appareil voulu.

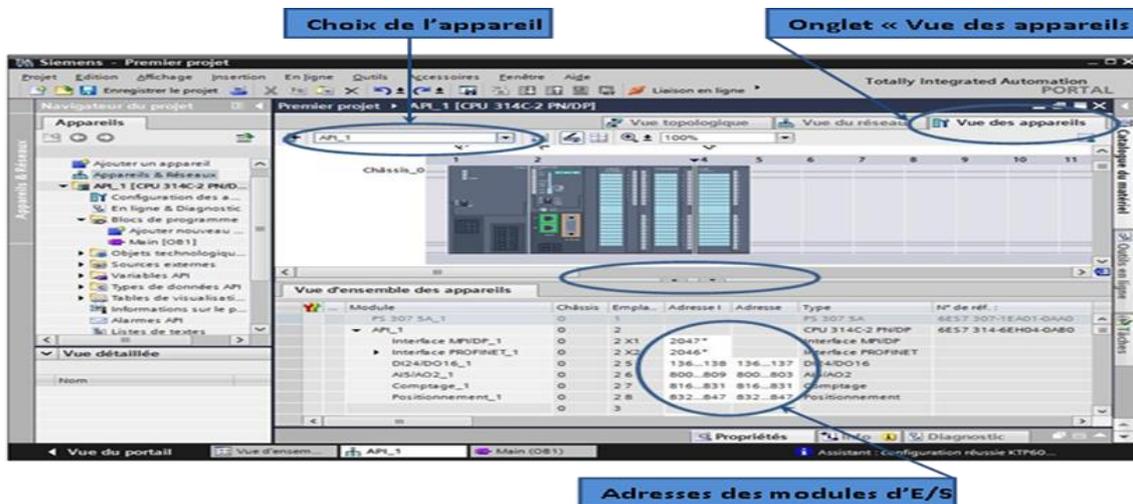


Figure 63: Adressage des E/S

On sélectionne la CPU puis à l'aide des deux petites flèches (voir figure), on fait apparaître l'onglet

« **Vue d'ensemble des appareils** » (n'hésitez pas à masquer certaines fenêtres et à en réorganiser d'autres). Les adresses des entrées et sorties apparaissent. Vous pouvez les modifier en entrant une nouvelle valeur dans la case correspondante.

VI.11 Compilation et chargement de la configuration matérielle

Une fois la configuration matérielle réalisée, il faut la compiler et la charger dans l'automate.

La compilation se fait à l'aide de l'icône « **compiler** » de la barre de tâche. On sélectionne l'API dans le projet puis cliquer sur l'icône « **compiler** ».

En utilisant cette manière, on effectue une compilation matérielle et logicielle.

Une autre solution pour compiler est de faire un clic droit sur l'API dans la fenêtre du projet et de choisir l'option « Compiler, Configuration matérielle » [33].

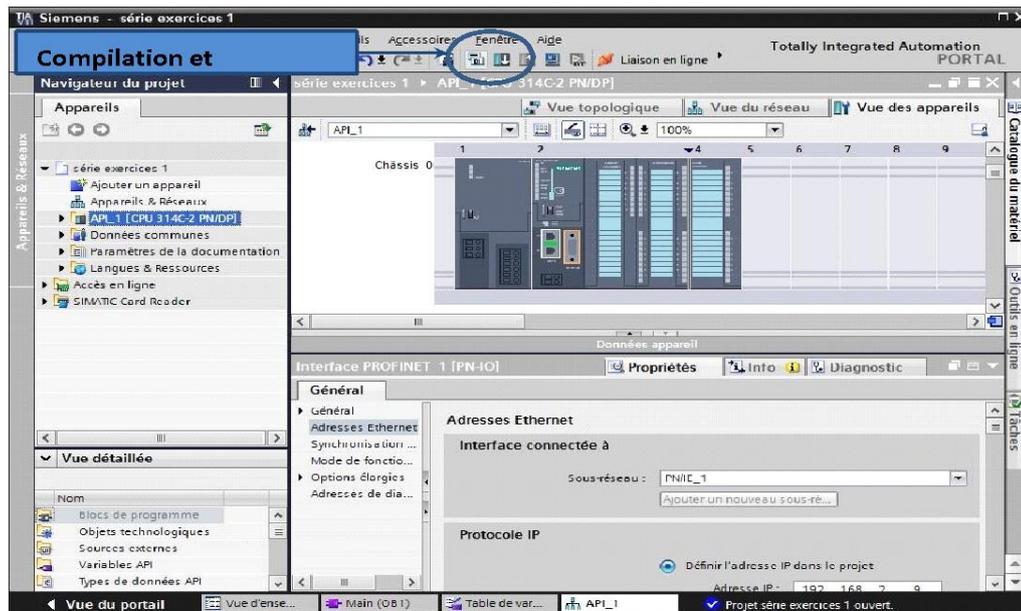


Figure 64: Compilation de la configuration matérielle

Pour charger la configuration dans l'automate, on effectue un clic sur l'icône « **charger dans l'appareil** ». La fenêtre ci-dessous s'ouvre et vous devez faire le choix du mode de connexion (PN/IE, Profibus, MPI). Si vous choisissez le mode PN/IE, l'API doit posséder une adresse IP.

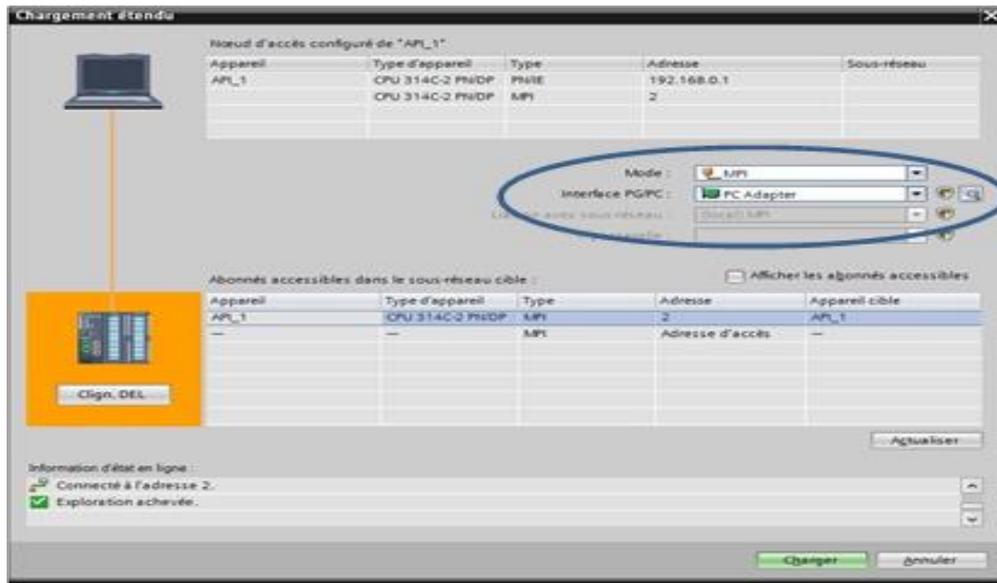


Figure 65: chargement de la configuration matérielle

Pour une première connexion ou pour charger l'adresse IP désirée dans la CPU, il est plus facile de choisir le mode de connexion MPI et de relier le PC à la CPU via le « **PC Adapter** ».

Si le programme trouve un appareil, il figure dans la liste en bas de la fenêtre. La touche « **Clign". DEL** » permet de faire clignoter une LED sur la face avant de l'appareil afin de s'assurer que l'on est connecté à l'appareil désiré.

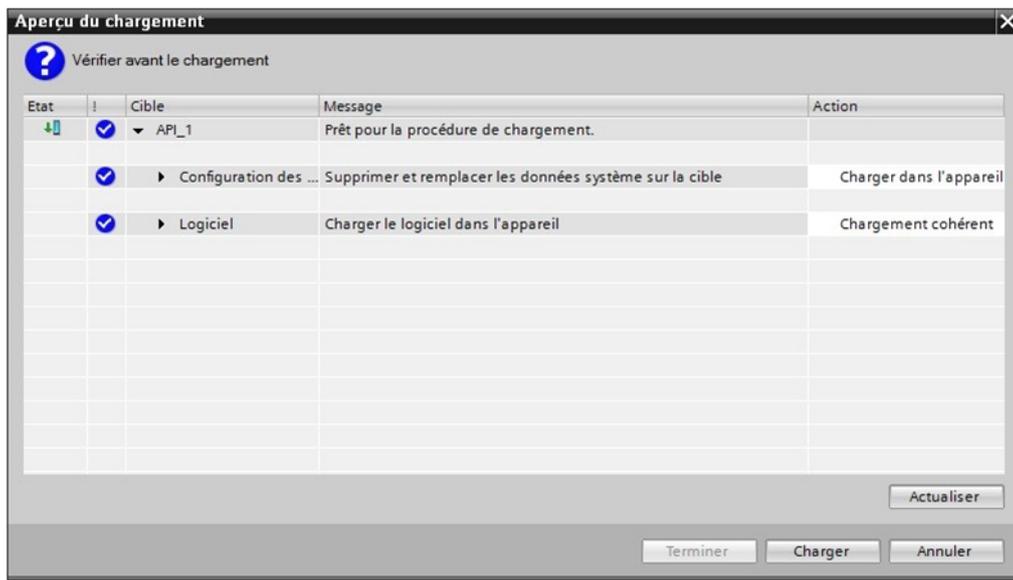


Figure 66 : Vérification avant chargement

Une fois la configuration terminée, on peut charger le tout dans l'appareil. Des avertissements / confirmations peuvent être demandés lors de cette opération. Si des erreurs sont détectées, elles seront visibles via cette fenêtre. Le programme ne pourra pas être chargé tant que les erreurs persistent.

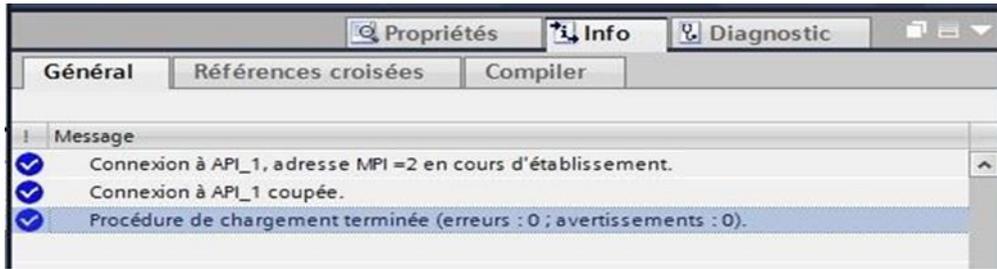


Figure 67 : Chargement est terminé

VI.12 Les blocs de programme.

L'automate met à disposition différents types de blocs qui contiennent le programme et les données correspondantes. Selon les exigences et la complexité du processus, il est possible de structurer le programme en différents blocs : OB, FB et FC [33].

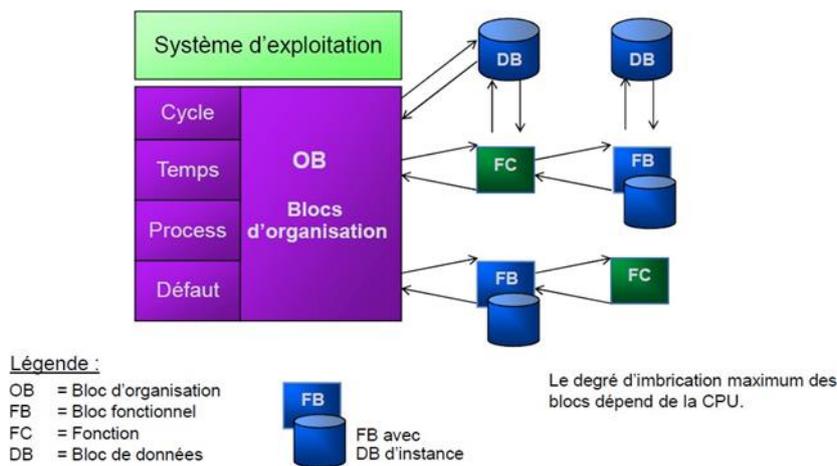


Figure 68: les blocs de programme

Pour crée un bloc de programme clique sur « bloc de programme » ensuite « ajoute nouveau bloc » choisi votre bloc et renomme enfin clique sur « ok ».

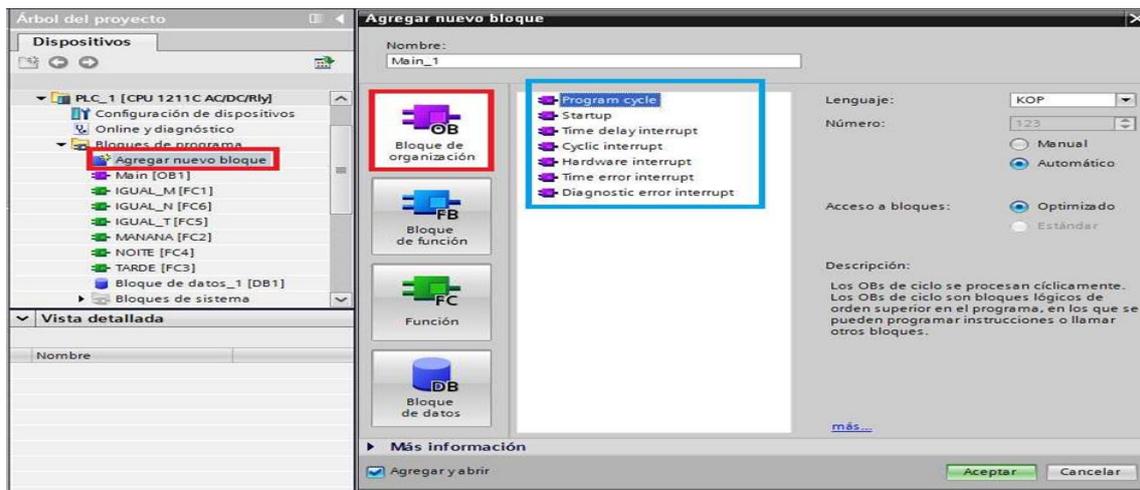


Figure 69: création un bloc de programme

VI.12.1 Les blocs d'organisation – OB

- Ils constituent l'interface entre le système d'exploitation et le programme utilisateur.
- Les OB sont programmables par l'utilisateur, ce qui permet de déterminer le comportement de la CPU.

Les OB sont appelés par le système d'exploitation en liaison avec les événements suivants :

- Comportement au démarrage
- Exécution cyclique du programme
- Exécution du programme déclenchée par des alarmes (cyclique, processus, diagnostic...)
- Traitement des erreurs

Pour que le traitement du programme démarre, le projet doit posséder au moins un OB cyclique (par exemple l'OB 1).

1. Les OB de démarrage (ex : OB 100)

Ils sont traités une seule fois, lorsque le mode de fonctionnement passe de STOP à RUN. Après le traitement de l'OB de démarrage, c'est le traitement de l'OB cyclique qui démarre (voir « type de démarrage dans S7 »).

2. Les OB cycliques (ex : OB 1)

Ils sont traités cycliquement. Les OB cycliques sont des blocs de code de niveau supérieur dans le programme, dans lesquels on peut appeler d'autres blocs.

3. Les OB de traitement périodique

Le traitement cyclique du programme peut être interrompu par des OB de priorités supérieur.

4. Les OB d'alarmes horaires (OB 10)

Les alarmes horaires sont utilisées pour exécuter un programme donné, appelé dans l'OB 10, une seule fois à un moment précis ou périodiquement à partir de ce moment (toutes les minutes, toutes les heures, tous les jours, toutes les semaines, tous les mois, déclenchement annuel).

En fonction de la CPU, l'utilisateur dispose au maximum de huit alarmes horaires différentes.

5. Les OB d'alarmes temporisées (OB 20, 21)

S7 met à disposition jusqu'à quatre OB (en fonction de la CPU) qui sont traités à chaque fois à la suite d'une temporisation paramétrable.

Le traitement du programme d'un OB d'alarme temporisée (OB20) est lancé avec retard après l'apparition d'un événement déterminé.

L'OB20 peut être activé uniquement par un appel de l'instruction « **SRT_DINT** ». Cette instruction est également utilisée pour préciser la durée de la temporisation.





L'action peut être annulée par l'instruction « CAN_DINT ».

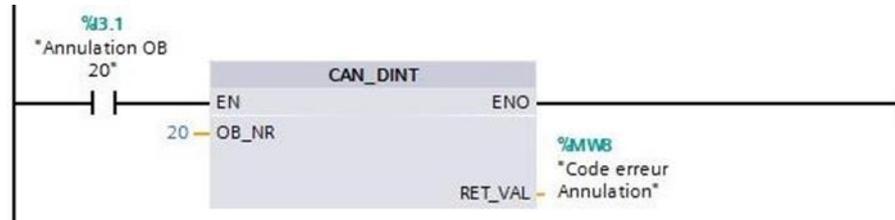


Figure 70: Les OB d'alarmes temporisées

6. Les OB d'alarmes cycliques (OB 30...37)

Une alarme cyclique permet de lancer le traitement d'un bloc à intervalles réguliers. Le S7-300 dispose de l'OB d'alarme cyclique OB35. Par défaut, l'intervalle de temps pour l'appel de l'OB est de 100 ms, la plage de réglage allant de 1 ms à 1 min.

En fonction de la CPU, l'utilisateur dispose au maximum de huit alarmes cycliques différentes.

Il faut veiller à ce que l'intervalle défini soit supérieur au temps nécessaire à l'exécution du contenu de l'OB 35. Si l'OB 35 est encore actif au moment où il est appelé, le système appelle l'OB 80 (erreur d'alarme cyclique).

7. Les OB d'alarmes processus

Le traitement du programme d'un OB d'alarme de processus (OB40) est lancé dès qu'un événement déterminé survient dans le processus.

Les alarmes de processus peuvent être déclenchées par différents signaux provenant des modules :

- **Sur les modules de signaux paramétrables** (DI, DO, AI, AO), le signal qui doit déclencher l'alarme de processus est défini avec l'outil de configuration matérielle.
- **Sur les modules CP** (processeur de communication) et FM (module de fonction), le comportement de l'alarme est défini à l'aide du logiciel de configuration du module correspondant.
- En fonction de la CPU, l'utilisateur dispose au maximum de huit alarmes de processus différentes. Il n'y a pas d'alarme processus disponible sur les automates du CTA [33].

8. Les OB de traitement des erreurs asynchrones

Par définition, les erreurs asynchrones surviennent de manière asynchrone par rapport au traitement du programme et ne peuvent donc pas être imputées à un endroit précis du programme.

Le tableau ci-dessous nous montre les différents types d'erreurs asynchrones.

Tableau 1: les différents types d'erreurs asynchrones

| Type d'erreur | Exemple | OB d'erreur | Priorité |
|---------------------------------|--|-----------------|---|
| Alarme esclave DPV1 | Alarme spéciale des esclave Profibus DPV1 | OB 55 / 56 / 57 | 26 / 28 (également paramétrable selon la CPU) |
| Perte de redondance | Coupure de la CPU (uniquement S7-400 à disponibilité élevée) | OB 70 / 72 | |
| Erreur de temps | Dépassement du temps de cycle maxi. | OB 80 | |
| Défaut d'alimentation | Défaillance de la pile de sauvegarde | OB 81* | |
| Alarme de diagnostic | Rupture de fil sur module avec fonction de diagnostic | OB 82 | |
| Alarme débrogage/enfichage | Retrait / Montage d'un module | OB 83 | |
| Défaut matériel CPU | Niveau de signal erroné sur l'interface MPI | OB 84* | |
| Erreur d'exécution du programme | Erreur d'actualisation de la mémoire image | OB 85 | |
| Défaut du châssis | Défaillance d'un esclave DP | OB 86 | |
| Erreur de communication | Détection erronée d'un télégramme | OB 87 | |

Diagramme de fonctionnement des OB

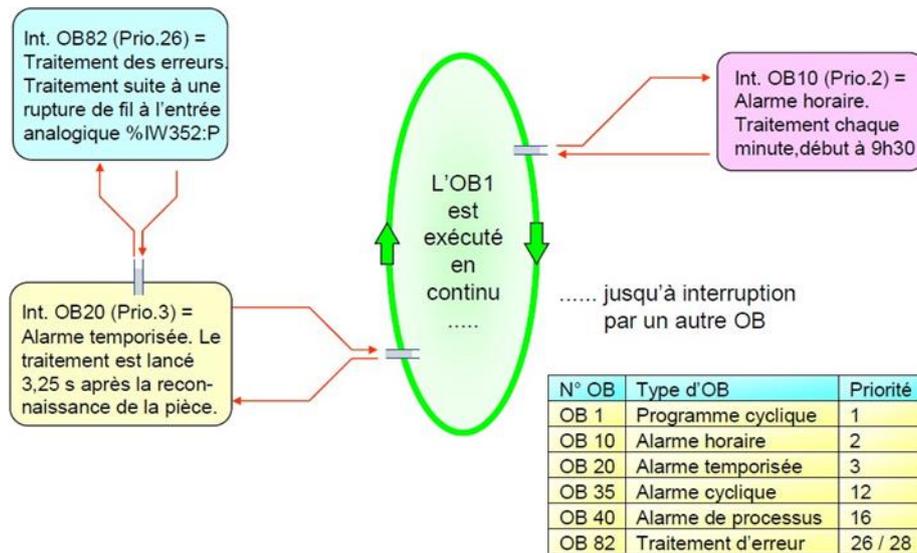


Figure 71: Diagramme de fonctionnement des OB

Les OB constituent l'interface entre le système d'exploitation de la CPU et le programme utilisateur. Ils sont appelés exclusivement par le système d'exploitation. Certains événements (alarmes horaires et de processus,) entraînent le lancement d'un OB approprié [33].

9. Interruption des OB

- Lorsque le système d'exploitation appelle un OB autre que l'OB1, il interrompt le traitement cyclique du programme car l'OB1 est **celui qui a la priorité la plus faible**. Tout OB peut donc interrompre le programme principal et lancer l'exécution de son propre programme, le traitement de l'OB1 reprenant ensuite au point d'interruption.
- Lorsque le système appelle un OB de priorité supérieure à celui déjà en cours d'exécution, l'interruption intervient après l'opération en cours de traitement.

Le système d'exploitation sauvegarde alors la pile complète des registres du bloc interrompu. Les informations contenues dans ces registres sont restaurées lorsque le système d'exploitation reprend le traitement du bloc interrompu [33].

10. Priorité des OB

- Le traitement d'un OB peut être interrompu aux limites d'une instruction par un événement (OB) de priorité supérieure.
- Les priorités s'échelonnent de 0 à 28, 0 étant la priorité la plus faible et 28 la priorité la plus forte.
- L'OB 82 possède soit la priorité 26 lorsqu'il survient au cours du traitement de l'OB 1, soit la priorité 28 lorsqu'il survient au cours d'un démarrage.
- Les OB de même priorité ne s'interrompent pas mutuellement, mais sont traités selon leur ordre d'occurrence [33].

ii. Les fonctions – FC

- Ce sont des blocs de code sans mémoire.
- Les données des variables temporaires sont perdues après l'exécution de la fonction. Si on veut mémoriser ces données, il faut utiliser des opérandes globaux.
- Elles sont utilisées pour la programmation de fonctions utilisées plusieurs fois. On simplifie de ce fait la programmation.

iii. Les blocs fonctionnels – FB

Ce sont des blocs de code qui mémorisent durablement leurs paramètres d'entrée, de sortie et d'entrée/sortie dans des blocs de données d'instance afin qu'il soit possible d'y accéder même après le traitement de blocs.

- L'utilisation de ces différents blocs permet plusieurs structures du programme.

iv. Les blocs de données « DB »

- Servent à mémoriser les données du programme.
Les blocs de données contiennent des données variables qui sont utilisées dans le programme utilisateur.
- Un bloc de données « DB » se définit par l'ensemble des variables qui le structure.
- Il existe deux types de blocs de données :

Bloc de donnée global : << DB >>

- Il est entièrement personnalisable : c'est l'utilisateur qui déclare les variables contenues dans le « DB ».
- Il ne contient que des variables statiques.
- Il n'est affecté à aucun bloc de code spécifique.

Bloc de donnée d'instance : « DB_I »

- Il est affecté directement à un bloc fonctionnel FB instancié.
- La structure d'un << DB >> d'instance « DB_I » n'est pas définissable librement, elle est définie par la déclaration d'interface du modèle bloc fonctionnel « FB » [33].

V. Conclusion

Dans ce chapitre nous avons présenté les bases de l'automatisation, les automates programmables La gamme Siemens et son environnement de développement. Et enfin les différentes étapes pour créer un projet sur TIAPORTAL.

Chapitre 03 : Partie pratique

I. Introduction

Après la création d'un projet TIA portal (revoir chapitre 2), on ajoute un appareil API, la configuration des matériels, les mnémoniques, ainsi que l'écriture du programme.

Nous commencerons par la présentation de l'architecture globale du projet les différentes fonctions.

II. La hiérarchie du programme

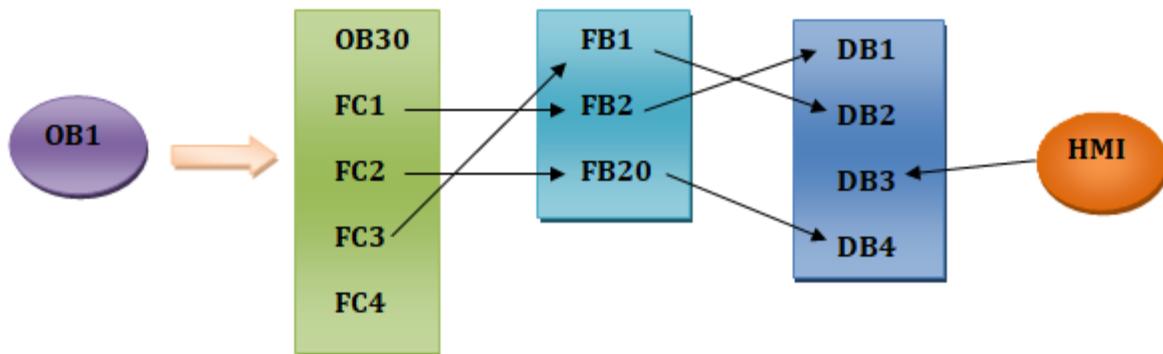


Figure 72: Hiérarchie du programme

Le Programme global est subdivisé en plusieurs sous-systèmes, afin de mieux gérer le système, le diagnostic et sa maintenance. Pour ce faire, des blocs des fonctions (FC) ont été créés pour organiser notre travail, ces derniers contiennent des blocs fonctionnels (FB) où chacun possède des données contenues dans le bloc de données (DB) associé. Ceci permet de mieux tester et déboguer les programmes.

Notre système se base sur 3 blocs fonctionnels et 1 bloc de fonction essentiels (VANNE TOR « FB2 », VANNE REGULATION « FB20 », MOTEUR « FB1 » et ANALOGIQUE « FC4 »). Ainsi que deux blocs Organisationnels OB1 et OB30.

FB2 VANNE TOR a pour rôle de confirmation que l'eau de refroidissement est à température 12°C Elle est composée des fins de course de fermeture et ouverture et la commande d'ouverture et fermeture et le mode distant. Elle contient 32 réseaux. Le réseau d'ouverture c'est réseau 1 et le réseau de fermeture c'est réseau 2.

Pour la programmation on utilise le langage de programmation graphique « CONT ».

Leurs contenus seront détaillés dans la partie simulation et visualisation

III. Editeur de mnémoniques

Dans tous les programmes il faut définir la liste des variables qui vont être utilisées lors de la programmation. Pour cela la table des variables est créée, le tableau des variables API nous permet de définir la liste des variables qui seront utilisées par cet API.

Dans cette table on a identifié toutes les variables et leurs adresses utilisées dans notre Programme.

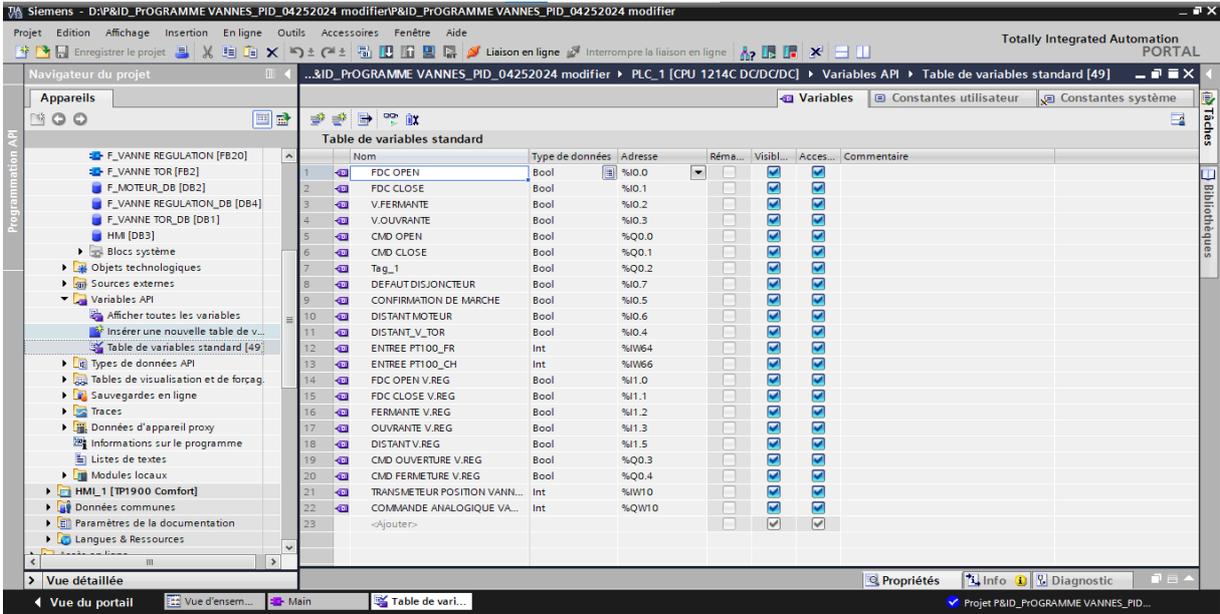


Figure 73: Table de variables

IV. Simulation de programme de système de contrôle de température

Lancer l'automate virtuel par module de simulation PLCSIM en cliquant sur l'icône démarrer la simulation

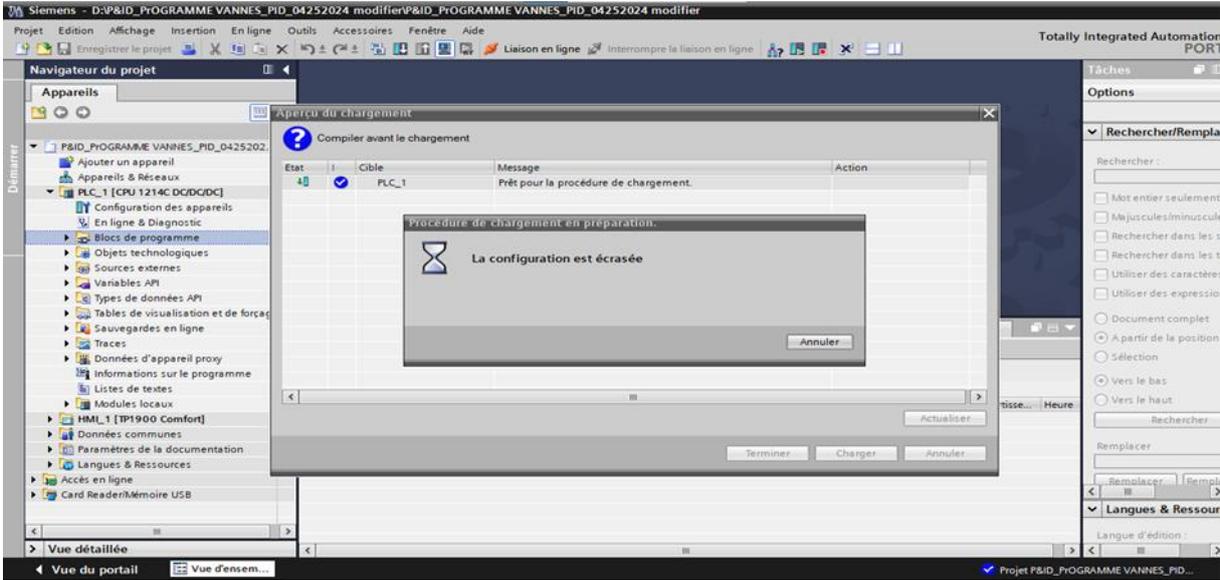


Figure 74: Démarrage de simulation

| Nom | Adresse | Format d'affichage | Valeur visualiséede forçage | Bits | Forçage par lot | Comme. |
|--|---------|--------------------|-----------------------------|------|--------------------------------|--------------------------|
| *FDC OPEN* | %I0.0 | Bool | FALSE | | <input type="checkbox"/> FALSE | <input type="checkbox"/> |
| *FDC CLOSE* | %I0.1 | Bool | FALSE | | <input type="checkbox"/> FALSE | <input type="checkbox"/> |
| *V.FERMANTE* | %I0.2 | Bool | FALSE | | <input type="checkbox"/> FALSE | <input type="checkbox"/> |
| *V.OUVRANTE* | %I0.3 | Bool | FALSE | | <input type="checkbox"/> FALSE | <input type="checkbox"/> |
| *CMD OPEN* | %Q0.0 | Bool | FALSE | | <input type="checkbox"/> FALSE | <input type="checkbox"/> |
| *CMD CLOSE* | %Q0.1 | Bool | FALSE | | <input type="checkbox"/> FALSE | <input type="checkbox"/> |
| *Tag_1* | %Q0.2 | Bool | FALSE | | <input type="checkbox"/> FALSE | <input type="checkbox"/> |
| *DEFAULT DISJONCTEUR* | %I0.7 | Bool | FALSE | | <input type="checkbox"/> FALSE | <input type="checkbox"/> |
| *CONFIRMATION DE MARCHÉ* | %I0.5 | Bool | FALSE | | <input type="checkbox"/> FALSE | <input type="checkbox"/> |
| *DISTANT MOTEUR* | %I0.6 | Bool | FALSE | | <input type="checkbox"/> FALSE | <input type="checkbox"/> |
| *DISTANT_V_TOR* | %I0.4 | Bool | FALSE | | <input type="checkbox"/> FALSE | <input type="checkbox"/> |
| *ENTREE PT100_FR* | %IW64 | DEC+/- | 0 | | 0 | <input type="checkbox"/> |
| *ENTREE PT100_CH* | %IW66 | DEC+/- | 0 | | 0 | <input type="checkbox"/> |
| *FDC OPEN V.REG* | %I1.0 | Bool | FALSE | | <input type="checkbox"/> FALSE | <input type="checkbox"/> |
| *FDC CLOSE V.REG* | %I1.1 | Bool | FALSE | | <input type="checkbox"/> FALSE | <input type="checkbox"/> |
| *FERMANTE V.REG* | %I1.2 | Bool | FALSE | | <input type="checkbox"/> FALSE | <input type="checkbox"/> |
| *OUVRANTE V.REG* | %I1.3 | Bool | FALSE | | <input type="checkbox"/> FALSE | <input type="checkbox"/> |
| *DISTANT V.REG* | %I1.5 | Bool | FALSE | | <input type="checkbox"/> FALSE | <input type="checkbox"/> |
| *CMD OUVERTURE V.REG* | %Q0.3 | Bool | FALSE | | <input type="checkbox"/> FALSE | <input type="checkbox"/> |
| *CMD FERMETURE V.REG* | %Q0.4 | Bool | FALSE | | <input type="checkbox"/> FALSE | <input type="checkbox"/> |
| *TRANSMETTEUR POSITION VANNE REG* | %IW10 | DEC+/- | 0 | | 0 | <input type="checkbox"/> |
| *COMMANDE ANALOGIQUE VANNE REGULATION* | %QW10 | DEC+/- | 0 | | 0 | <input type="checkbox"/> |

Figure 75: Table de simulation

IV.1 Chargement de programme

Charger le programme entier dans la CPU en sélectionnant d'abord l'API et en cliquant sur le symbole charger dans appareil

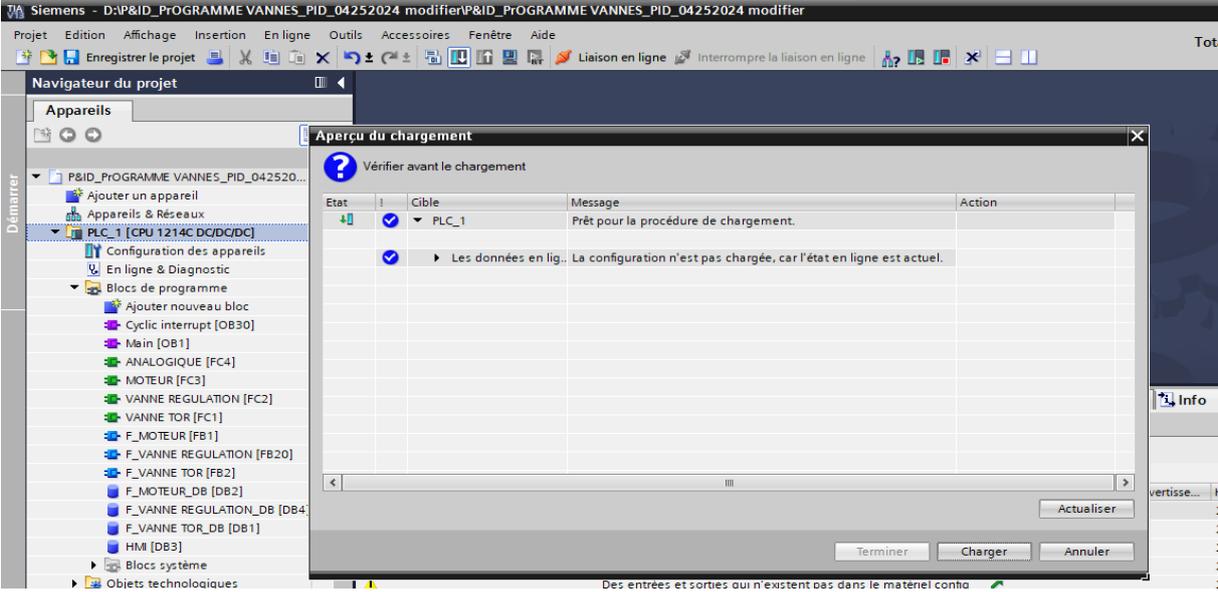


Figure 76: Chargement de programme

IV.2 Démarrage de WinCC Runtime

Démarrer WinCC Runtime sur le pc, activer la liaison en ligne , activer la visualisation dynamique

IV.3 Visualisation de l'état du programme

Après le chargement du programme dans la CPU du simulateur et la mise de cette dernière en mode « RUN » le SIMATIC STEP 7 nous permet de visualiser l'état du programme soit en cliquant sur l'icône 

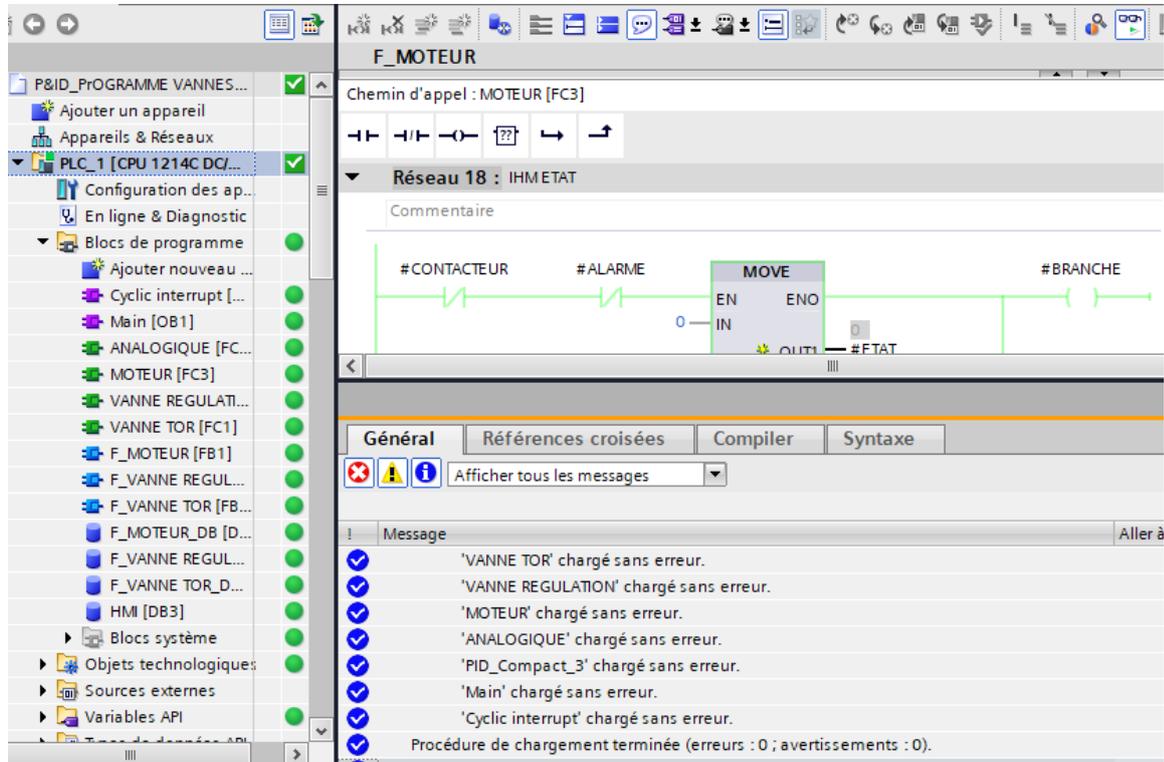


Figure 77: Visualisation de l'état du programme

V. Les blocs de programme

V.1 OB Cyclique interrupt (OB30)

Nous avons créé un bloc PID compact dans le but de mettre en œuvre un régulateur PID (Proportionnel, Intégral, Dérivé) pour le contrôle température de PVA de manière simplifiée et efficace. Ce bloc PID compact nous permettra d'automatiser le processus de régulations de température en ajustant automatiquement les entrées et les sorties du système afin de maintenir une température prédéfinie. Grâce à cette approche simplifiée, nous pourrions configurer facilement les paramètres du PID, tels que les constantes de réglage (K_p , K_i , K_d), pour optimiser le contrôle et assurer la stabilité et la précision du processus.

Par identification faite par ingénieur qui donne ces paramètres dans la période de stage et vu en chapitre 1 section régulation :

$K_p=0.5$

$K_i=772$ ms

$K_d=480$ ms

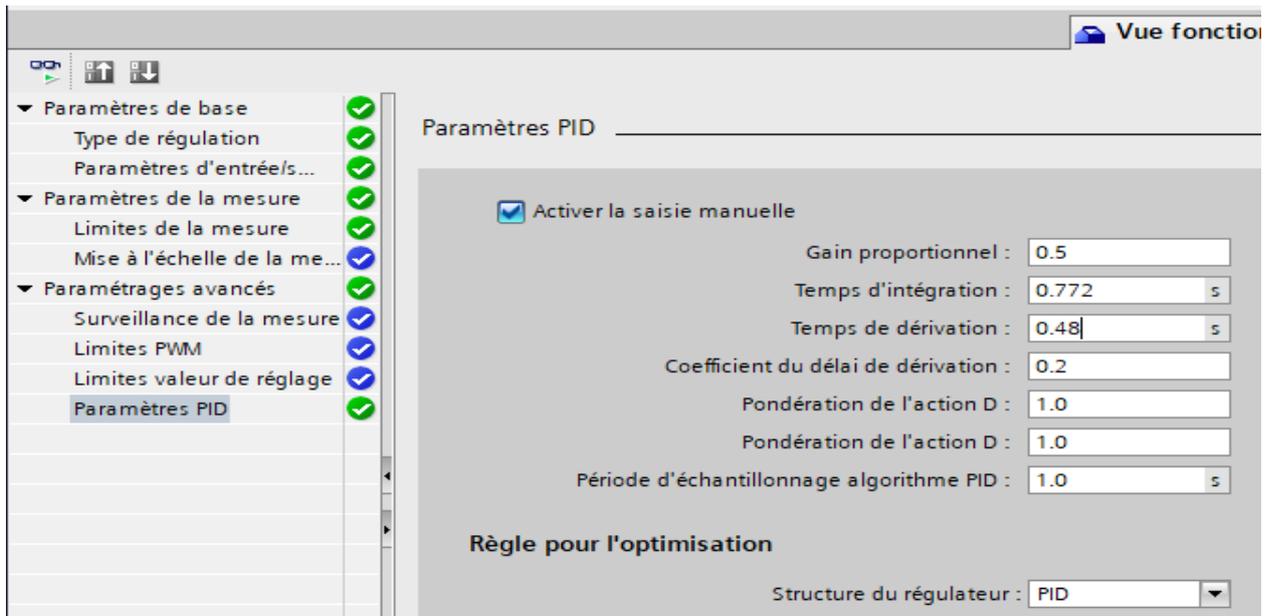


Figure 78: Paramètre de PID

Les conditions de démarrage de PID_Compact (DB5) est le commutateur de la vanne de régulation (DB4.DBX14.0) en mode automatique, la vanne en mode distant (I1.5) et aussi l'entrée de PID (BD3.DBD12).

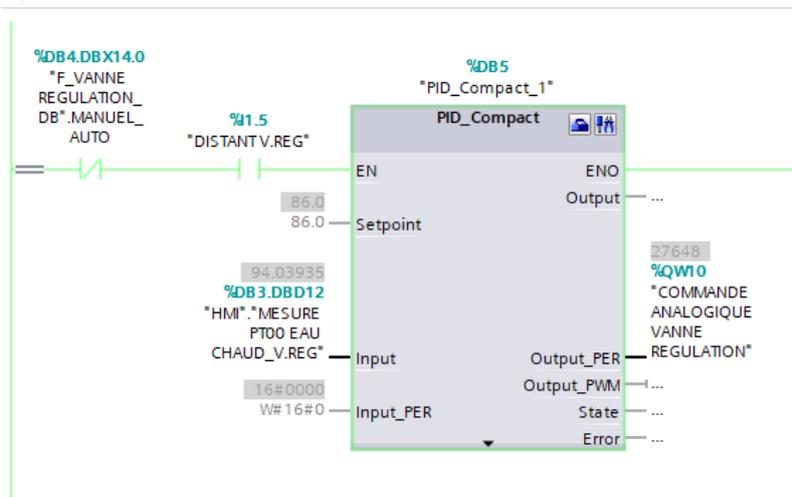


Figure 79: OB30 PID-Compact de température

V.2 Programme du Bloc d'Organisation OB1

Le système d'exploitation de la CPU exécute l'OB1 de manière cyclique. Il fait appel aux blocs fonctionnels (FB) et aux fonctions (FC). Dans l'ordre appel le bloc OB1 appelé le bloc d'interruption cyclique OB30 et aussi les blocs de fonction FC1, FC2, FC3, FC4 et ensuite les FC appelle les bloc fonctionnelles FB1, FB2, FB20 chacun contient un data bloc DB1, DB2, DB3, DB4.

Quand un bloc en appelle un autre, on traite les instructions du bloc appelé. Une fois le traitement du bloc appelé terminer, celui du bloc appelant reprend. Le traitement reprend à l'instruction qui suit l'appel de bloc.

La figure ci-dessous représente les réseaux de FC 1 vanne tout ou rien, FC 3 moteur de, FC 2 vanne de régulation et FC 4 analogique.

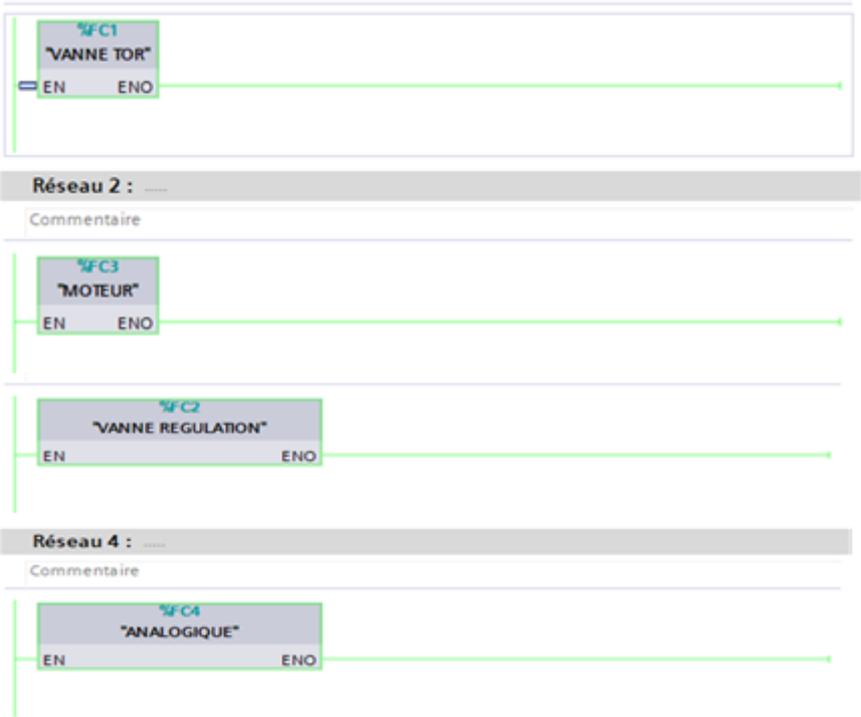


Figure 80: Blocs d'organisation main OB

V.3 Le bloc de fonction fc4

La figure ci-dessous représente les blocs NORM_X et SCALE_X pour entrer analogique de capteur PT100 de température de réacteur.

Dans cette fonction FC4, l'utilisation de l'instruction SCALE X et NORM X existant dans Instructions de base : Conversion avait pour but de traiter des valeurs analogiques (0 .. 27648), c'est-à-dire de convertir la valeur analogique à l'entrée en une valeur réelle à la sortie, qui elle-même est transférée directement à la vanne, selon une échelle (valeur limite inférieure c'est 0 et valeur limite supérieure 100.

Par équation suivante :

$$R = \frac{H - L}{27648} * (IW + L)$$

Dans notre cas signal analogique est reçu à l'entrée IW 66 à partir d'un capteur pt100 et la valeur de sortie réel DB 3.DB D12 sera l'entrée de régulateur DB5.

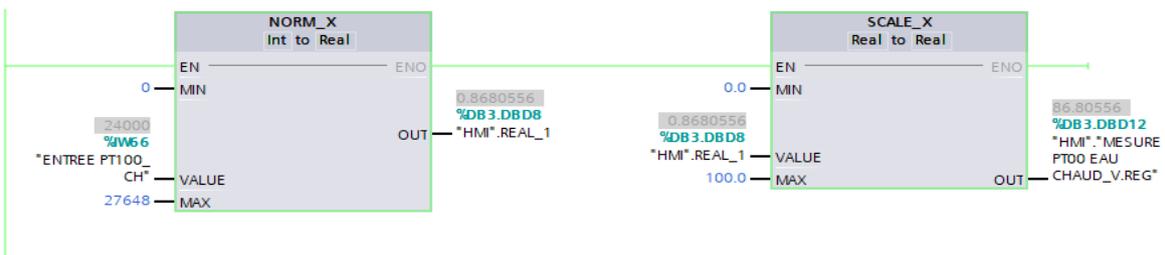


Figure 81: Réseau pour commander le régulateur

La figure ci- dessous représente le bloc fonction de vanne de régulation. Dans cette fonction FC2 on fait appel au bloc de fonction FB20 (Vanne de régulation). Les entrées de ce bloc de fonction sont :

- En mode automatique : mode distant (I1.5) Les fin de courses d'ouverture/ fermeture I1.0 et I1.1 respectivement
- En mode manuel : mode distant Les fin de courses d'ouverture/ fermeture I1.0 et I1.1 respectivement CMD OPEN_HMI (DB3.DBX 16.0) et CMD CLOSE_HMI (DB3.DBD16.1)

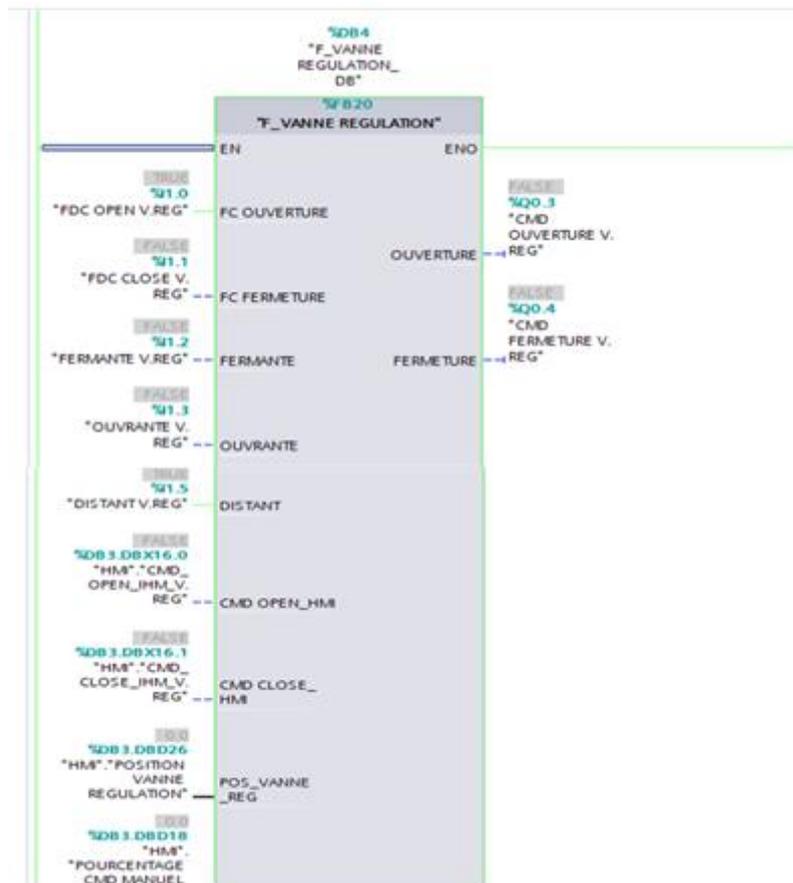


Figure 82: FC vanne de régulation

La figure ci-dessous représente les réseaux des ordres de fermeture et ordres ouverture automatique en mode simulation.

Si la mesure PT100 eau chaude est supérieure ou égale à 86° l'ordre d'ouverture automatique est actif.

Si la mesure PT100 eau chaude inférieure à 86° l'ordre de fermeture automatique est actif.

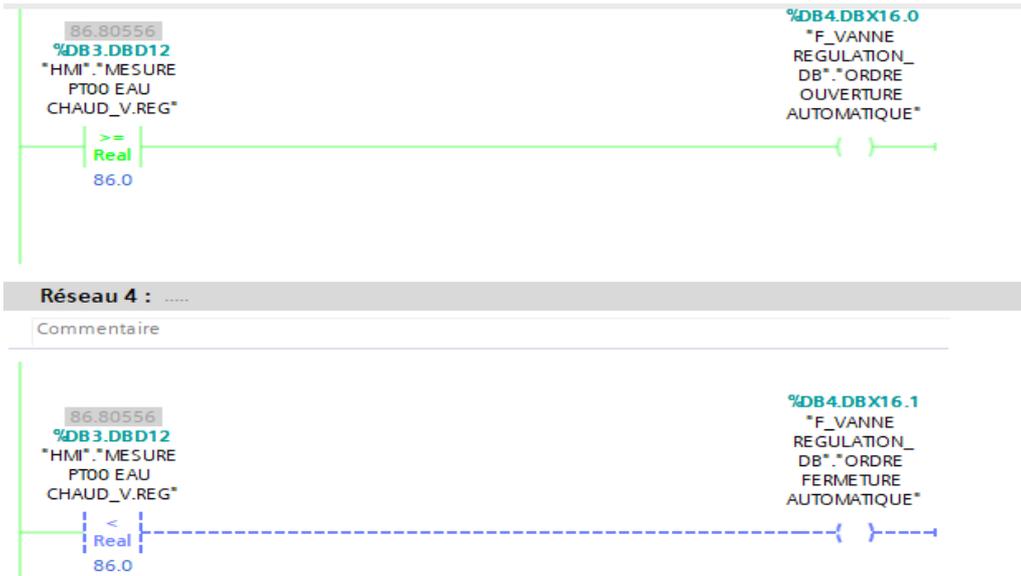


Figure 83: Les réseaux d'ordre fermeture et ouverture automatique

La figure ci-dessous représente le bloc fonctionnel de la vanne de régulation, le réseau de l'ouverture et la commande de la vanne régulatrice.

Pour activer la commande d'ouverture il faut que
 Si en mode automatique : ordre ouverture automatique, commutateur en mode auto, alarme désactivée, mode distant, fc ouverture désactivée et fermeture IHM ferme.

Si en mode manuel : ordre ouverture manuel stabilise, commutateur en mode manuel, alarme désactivée, mode distant, fc ouverture désactivé et fermeture IHM ferme.

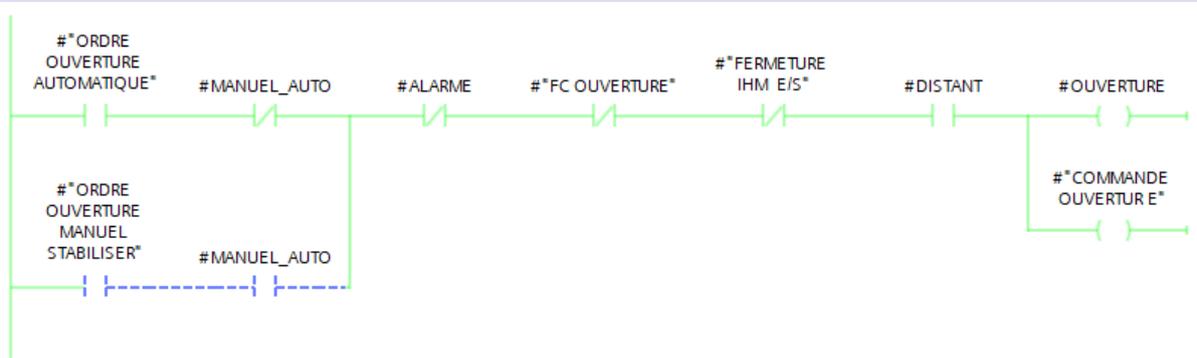


Figure 84: Commande de vanne en mode local, manuel et automatique

La figure ci-dessous représente le bloc fonctionnel de la vanne de régulation le réseau de fermeture et la commande de la vanne régulatrice

Pour activer la commande d'ouverture il faut qu'active :
 Si en mode automatique :
 Ordre fermeture automatique, commutateur en mode auto, alarme désactivée, mode distant, fc fermeture désactivée et ouverture IHM ferme.

Si en mode manuel :

Ordre fermeture manuel stabilise, commutateur en mode manuel, alarme désactivée, mode distant, fc fermeture désactivée et ouverture IHM ferme.

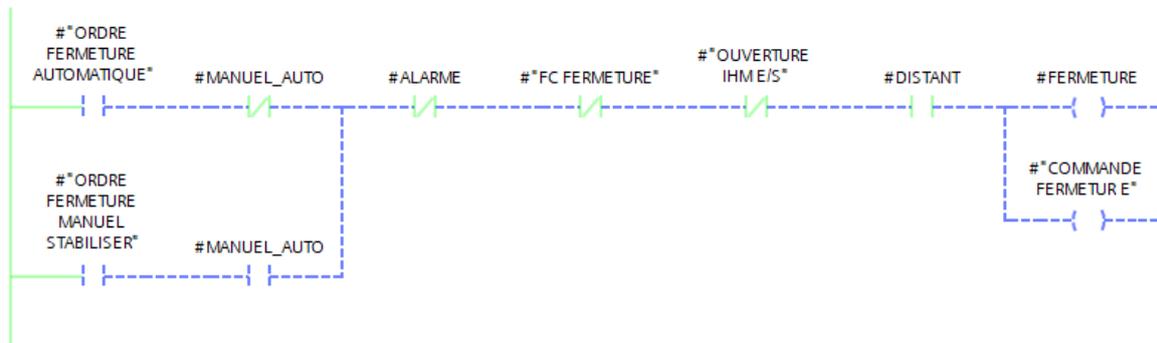


Figure 85: Commande de vanne pour fermeture en mode local, manuel et automatique

VI. Programmations des alarmes

La programmation d'une alarme nécessite deux étapes

VI.1 Programmation au niveau de l'automate.

Si la vanne est bloquée à l'ouverture, un message doit être affiché pour signaler le problème et protéger la vanne. Nous utilisons des variables de type Word pour afficher l'alarme et nous avons travaillé avec les instructions suivantes :

AND : "Opération logique ET", permet de relier la valeur à l'entrée IN1 avec la valeur à l'entrée IN2 bit par bit par une opération logique ET le résultat est transmis à la sortie OUT.

Lors de l'exécution de l'instruction, le bit 0 de la valeur à l'entrée IN1 est relié au bit 0 de la valeur à l'entrée IN2 par un ET logique. Le résultat est stocké dans le bit 0 de la sortie OUT.

OU : "Opération logique OU", permet de relier la valeur à l'entrée IN1 avec la valeur à l'entrée IN2 bit par bit par une opération logique OU et vous interrogez le résultat à la sortie OUT.

Lors de l'exécution de l'instruction, le bit 0 de la valeur à l'entrée IN1 est relié au bit 0 de la valeur à l'entrée IN2 par un OU logique. Le résultat est stocké dans le bit 0 de la sortie OUT.

INV : "Former le complément à 1", permet d'inverser l'état logique des bits à l'entrée IN ET fourni à la sortie OUT

SHL : "Décaler à gauche", permet de décaler le contenu de l'opérande à l'entrée IN bit par bit vers la gauche et le résultat à la sortie OUT. Le paramètre N, permet de déterminer le nombre de positions de bit dont il faut décaler la valeur indiquée. Le nombre généré permet de déclencher l'alarme quand elle est inversée par instruction INV.

En HMI. Message alarme est une variable à chaque fois va prendre une valeur mais alarme message est une variable de type Word qui contient 0001 toujours.

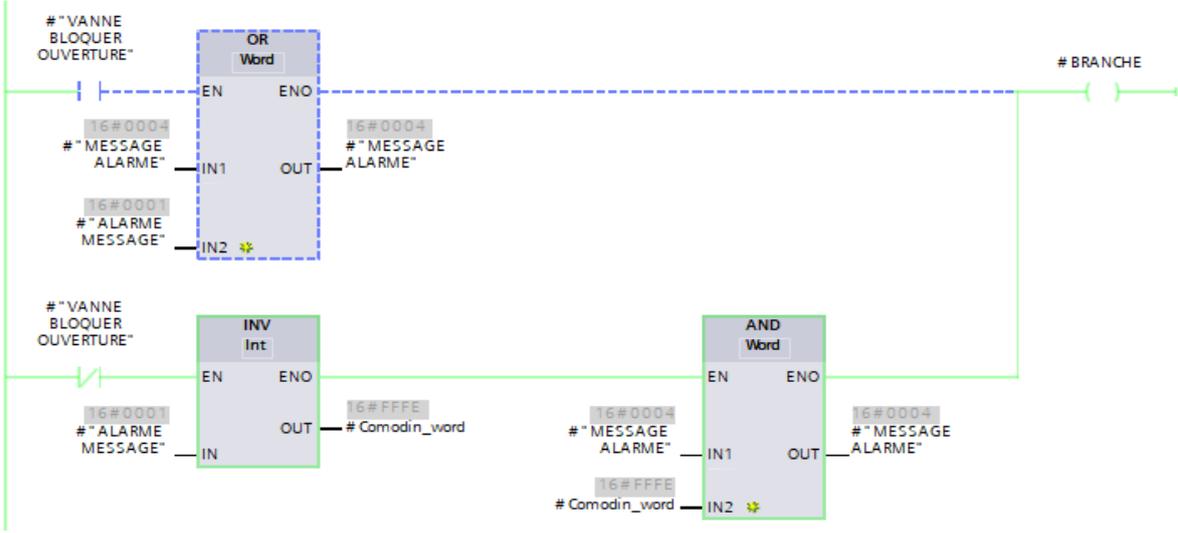


Figure 86: Réseau pour blocage alarme d'ouverture

Message alarme d'ouverture c'est pour opérateur voir que la vanne et ouvert

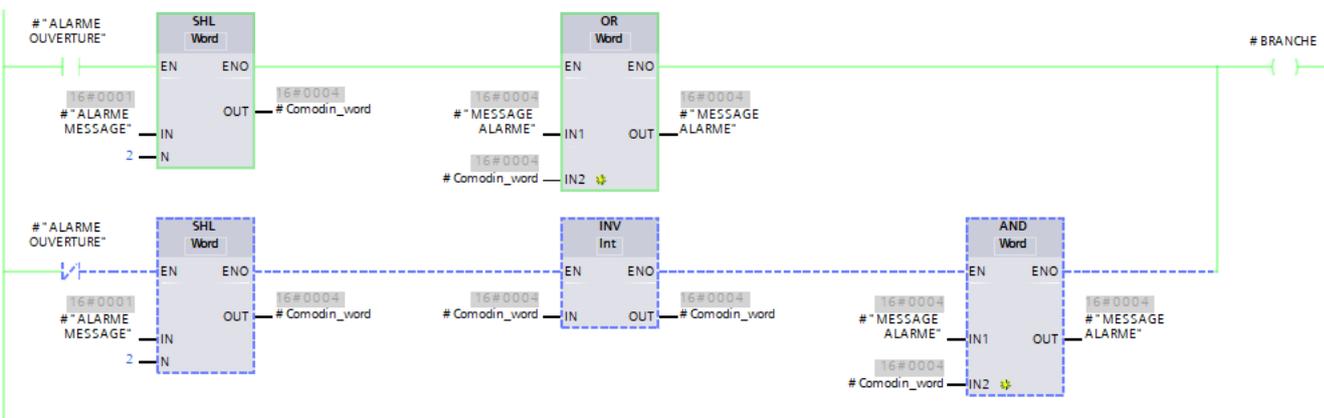


Figure 87: Réseau pour alarme d'ouverture

VI.2 Configuration au niveau de l'IHM

Les alarmes sont affichées sur le pupitre opérateur dans la fenêtre des alarmes ou la vue des alarmes, qui sont similaires en apparence et se contrôlent de la même manière.

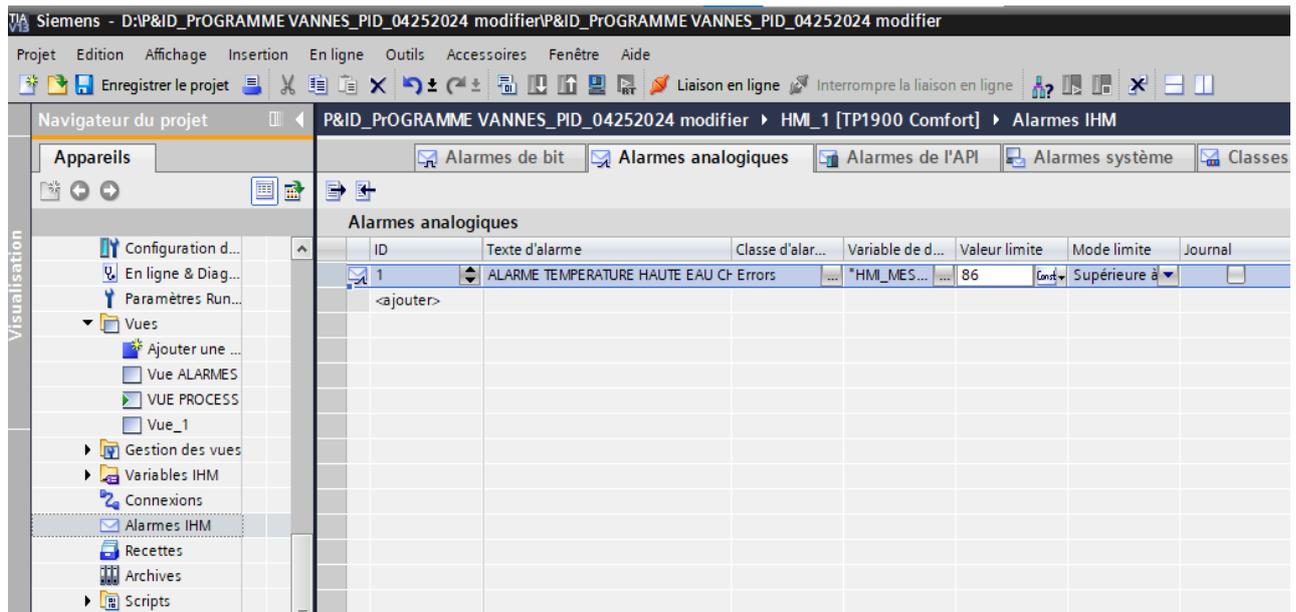


Figure 88: Fenêtre des alarmes

VII. La Vue Globale

Cette vue montre le schéma de visualisation de toute la station, la figure ci-dessous Représente la vue du Globale.

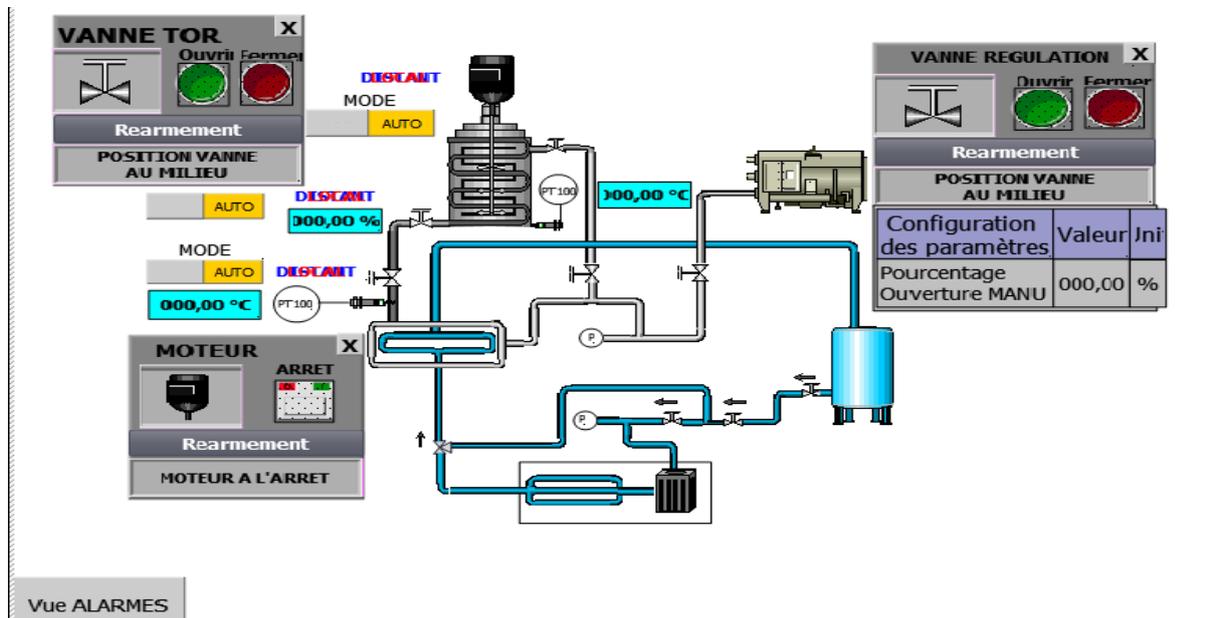


Figure 89: La Vue Globale

VII.1 Animation de la vanne de régulation

Afin d'animer la vanne et relier à une variable de l'API on mettre la représentation de la vanne :

- Cliquez sur " animation " > "Affichage " > "représentation "

- Puis ajoute "plage" et modifier la couleur

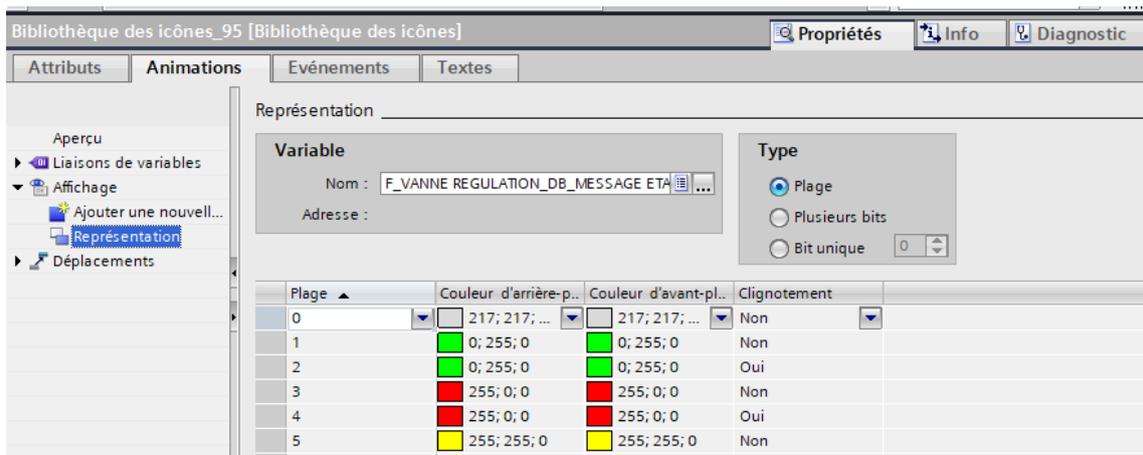


Figure 90: Animation de la vanne de régulation

VII.2 Création de bloc d'affichage (pop-up)

Après avoir créé la vue de l'IHM, la première étape est de créer des Faceplates pour le moteur et les deux vannes. Les Faceplates, ou blocs d'affichage standard, sont des ensembles personnalisés d'objets d'affichage et de commande, enregistrés, gérés et édités de manière centralisée dans le projet.

Pour créer un Faceplate, commencez par créer un bloc d'éléments en ajoutant des champs E/S et des champs de texte. Sélectionnez ensuite l'ensemble des éléments et cliquez sur "Créer un bloc d'affichage".

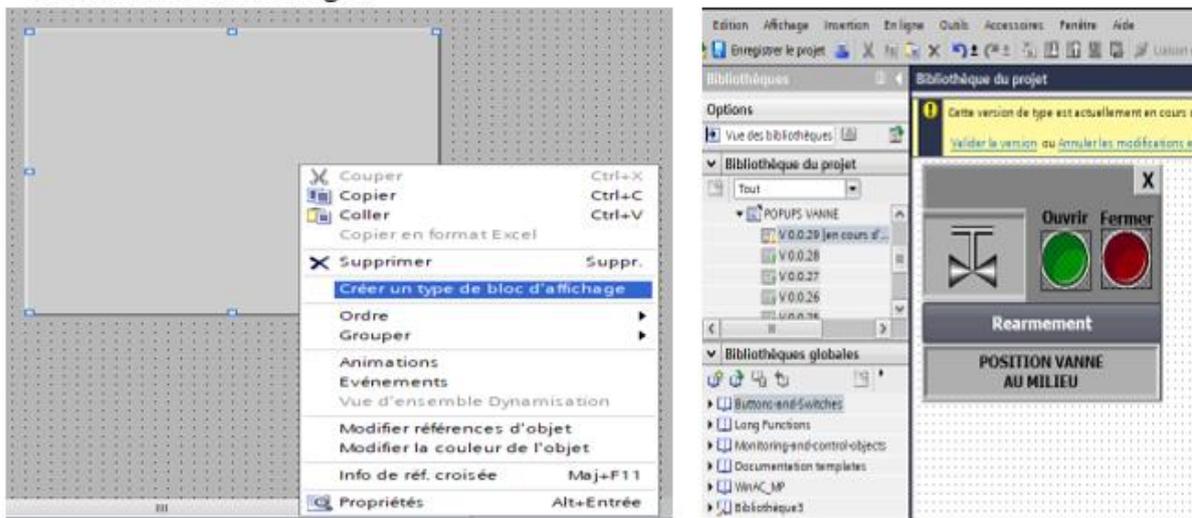


Figure 91: Création d'une Faceplate pour la vanne

VII.3 Animation de bouton d'état de vanne

La figure ci-dessous représente l'animation du bouton de l'état actuel de vanne de régulation (ouvert /fermé...)

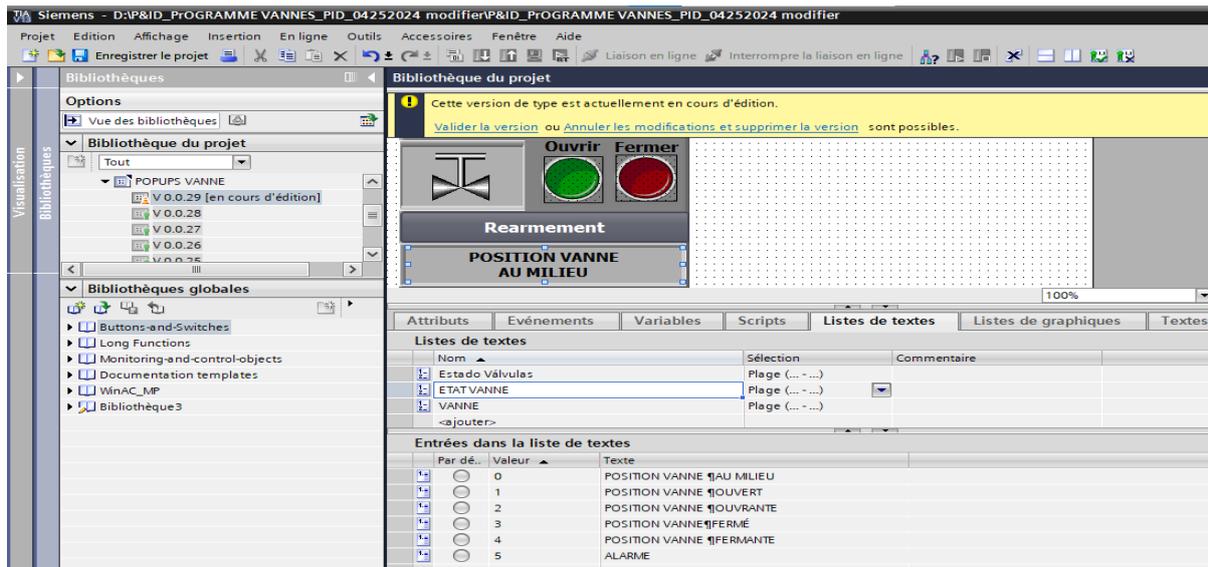


Figure 92: Animation de bouton d'état de vanne

VII.4 Animation du bouton arrêt

La figure ci-dessous représente l'animation du bouton d'ordre de fermeture de la vanne régulatrice

Afin d'animer le bouton il faut la relier à une variable de l'API

- Ouvrir la vue d'ensemble des variables
- Sélectionner sous "HMI " > "Variables IHM" la "Table de variables standard"
- Puis "Ajouter objet"

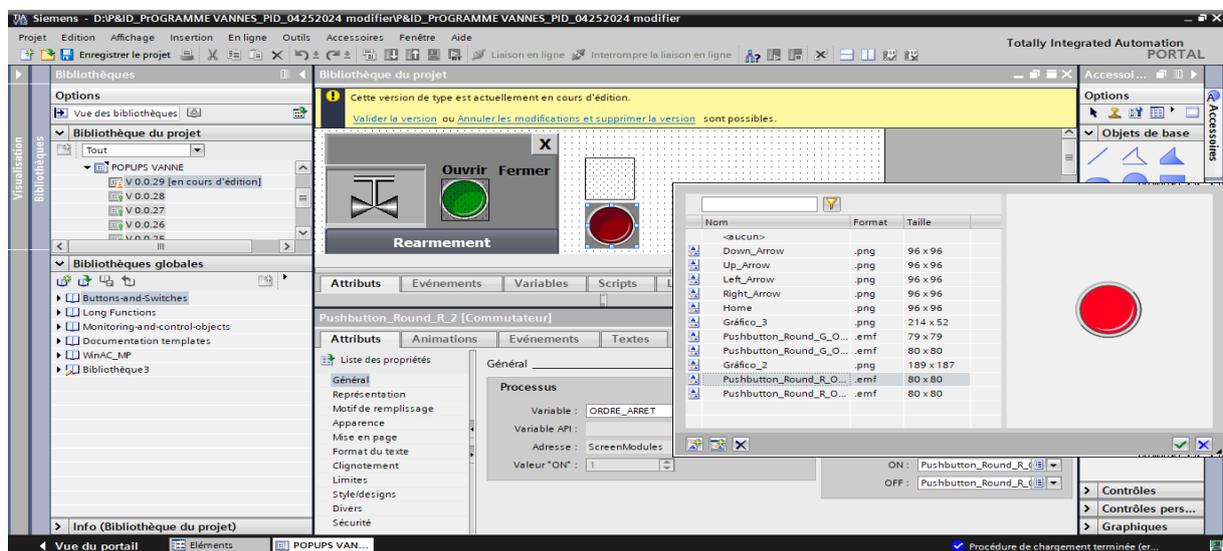


Figure 93: Animation de bouton arrêt

VII.5 Exemple sur le fonctionnement de PID Compact

VII.5.1 La visualisation des courbes

Après avoir activé la visualisation, Nous pouvons observer l'évolution des paramètres contrôlés par PID en cliquant sur « Ouvrir la fenêtre pour la mise en service » , ce qui nous amène à la page ci-dessous, où le graphique sera affiché :

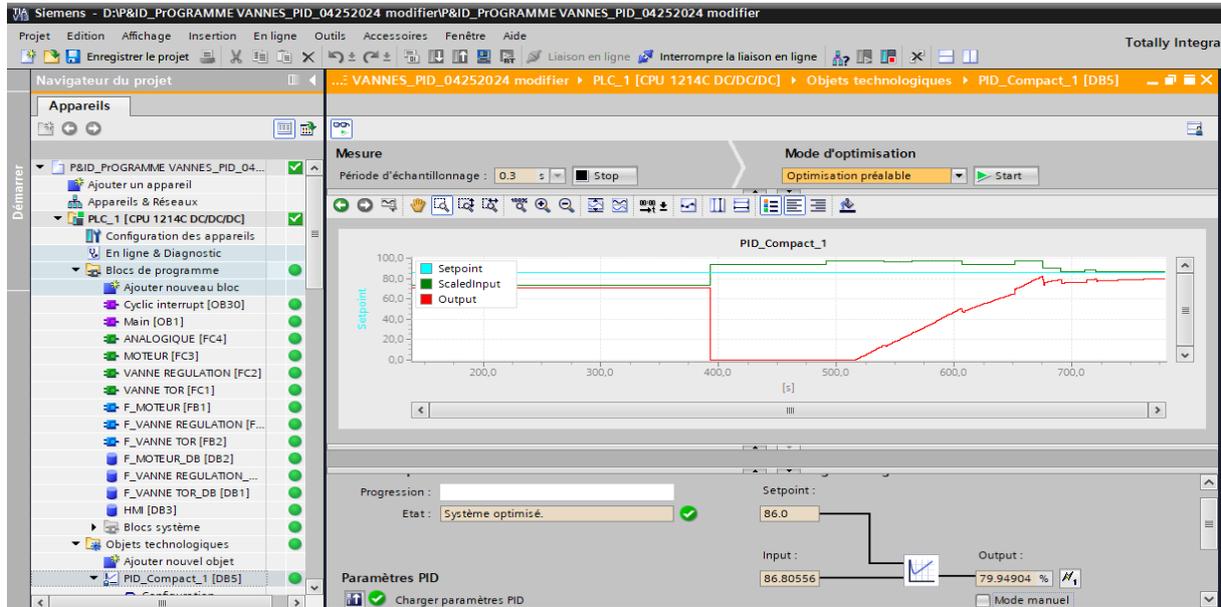


Figure 94: La visualisation des courbes

- La sortie « output » est en rouge
- L'entrée « Scaledinput » est en vert
- La consigne « set point » est en bleu

Lorsque le bouton "Start" est appuyé, notre système semble optimisé. Sur cette page, différentes cases affichent des informations telles que la valeur de consigne, la mesure, et la valeur de réglage représentant la sortie.

La figure ci-dessus montre un exemple de graphique avec une consigne réglée à 86°, une mesure réglée à 86,60° (l'entrée étant ajustée à partir du PLCSIM), et une sortie réglée automatiquement à 80%.

VII.6 La vue du process

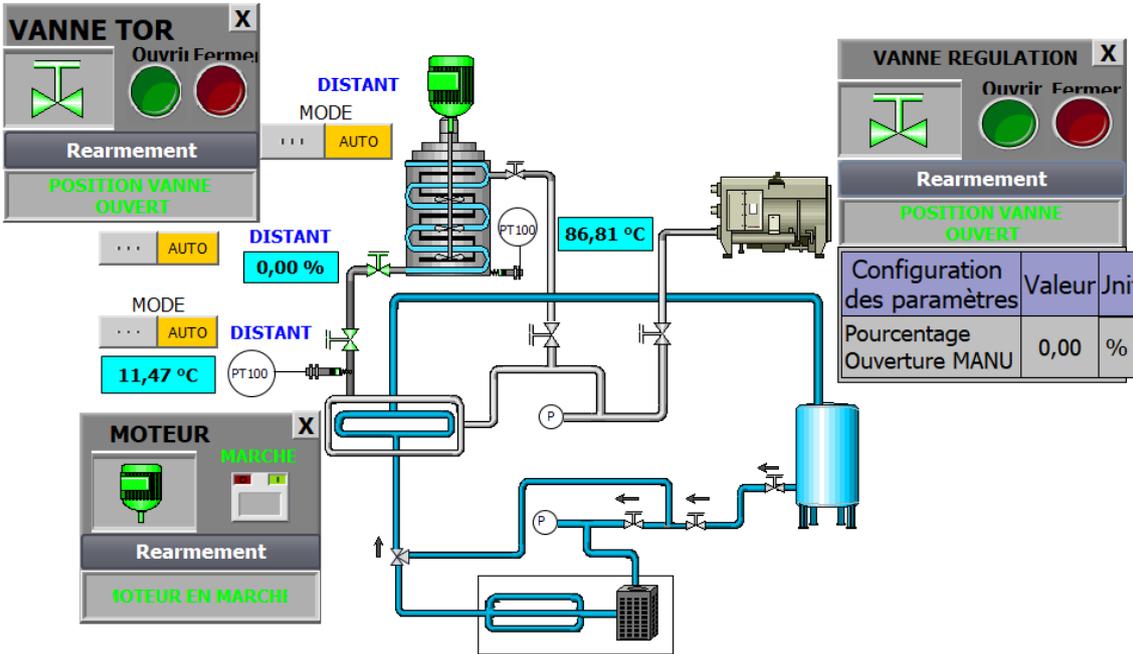


Figure 95: Simulation d'HMI

VII.6.1 Exemple vue alarme

ALARMES

| No. | Heure | Date | Texte |
|-----|----------|------------|------------------------------------|
| 1 | 22:42:22 | 09/06/2024 | ALARME TEMPERATURE HAUTE EAU CHAUD |

The table shows a single alarm entry. At the bottom of the window, there are icons for help, alarm, and refresh.

VUE PROCESS

Figure 96: Exemple d'alarme

VII.7 Sécurité

Pour sécuriser notre système, un mot de passe a été mis en place pour empêcher toute modification non autorisée. Les étapes pour mettre en œuvre cette sécurité sont les suivantes : cliquer sur "Gestion des utilisateurs", puis définir un mot de passe et choisir le nombre de groupes.

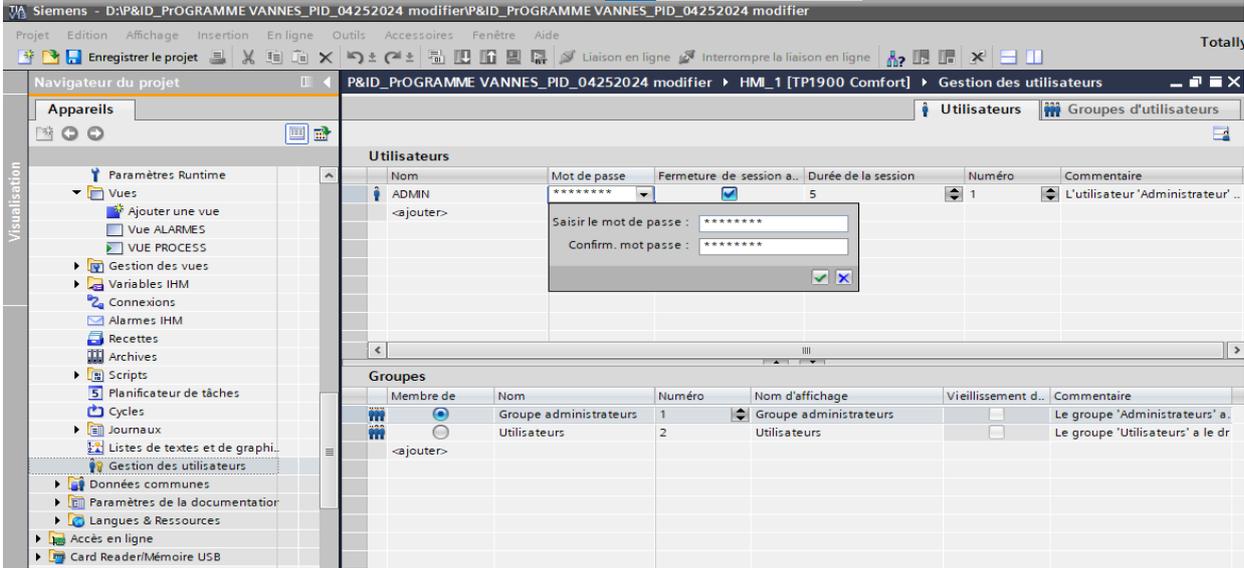


Figure 97:Création de mot de passe

Après clique sur groupes d'utilisateurs on peut donner les autorisations

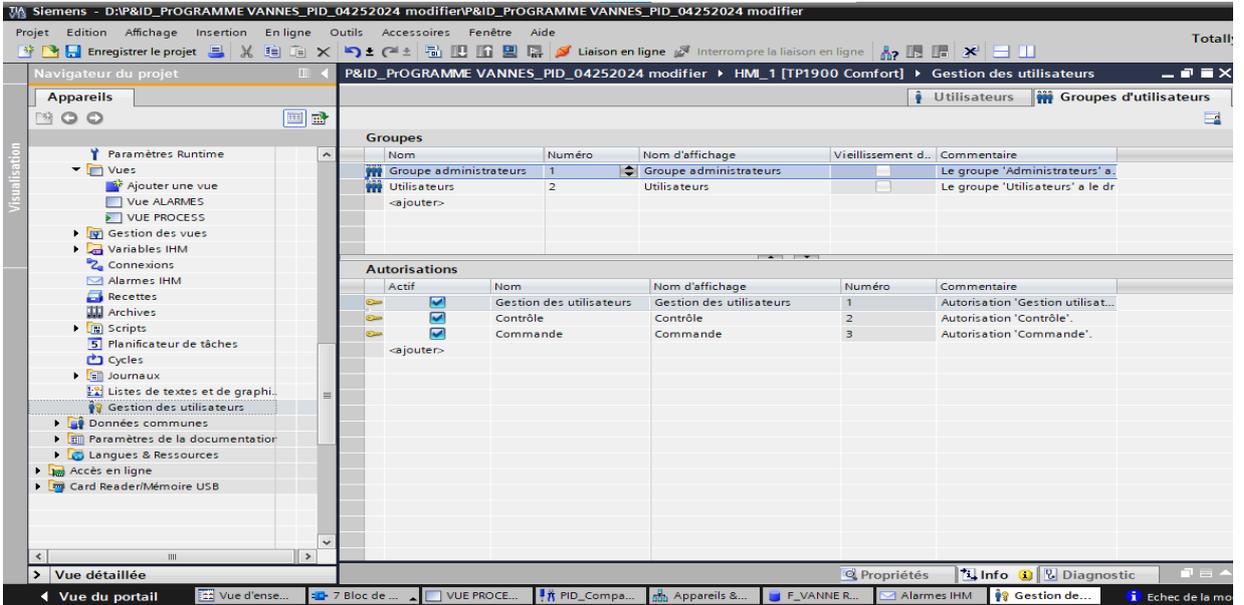


Figure 98: création de mot de passe

Enfin, dans la vue racine, l'instrument pour lequel la sécurité doit être appliquée est cliqué. Ensuite, dans les attributs, les options "sécurité" et "autorisation" sont sélectionnées.

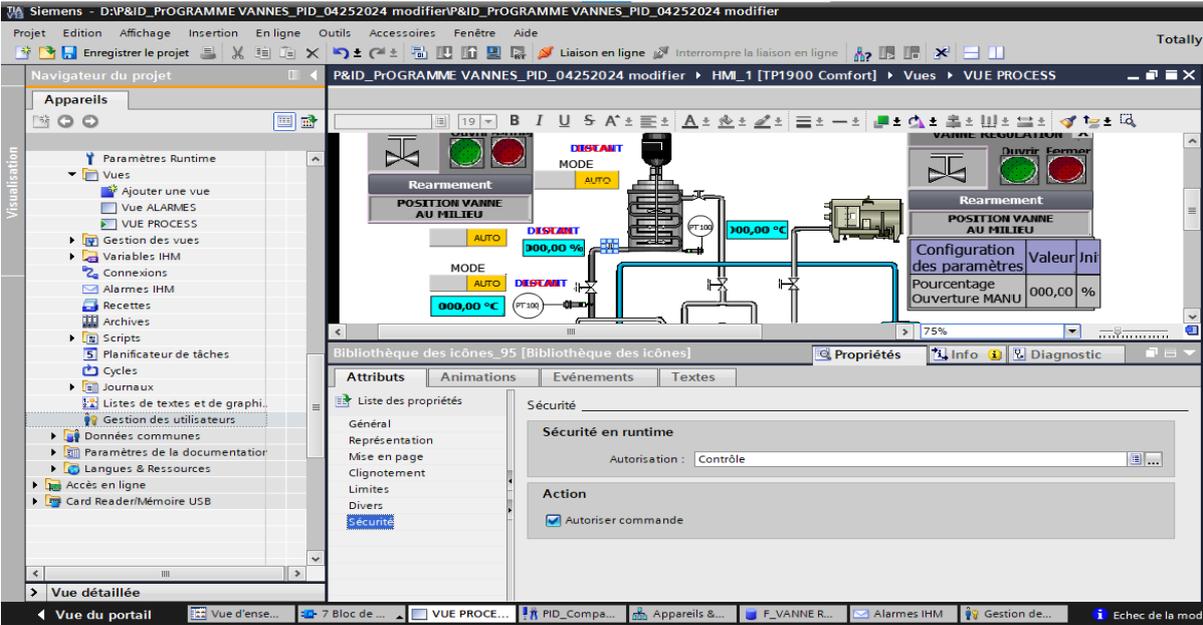


Figure 99: Création de mot de passe

Exemple

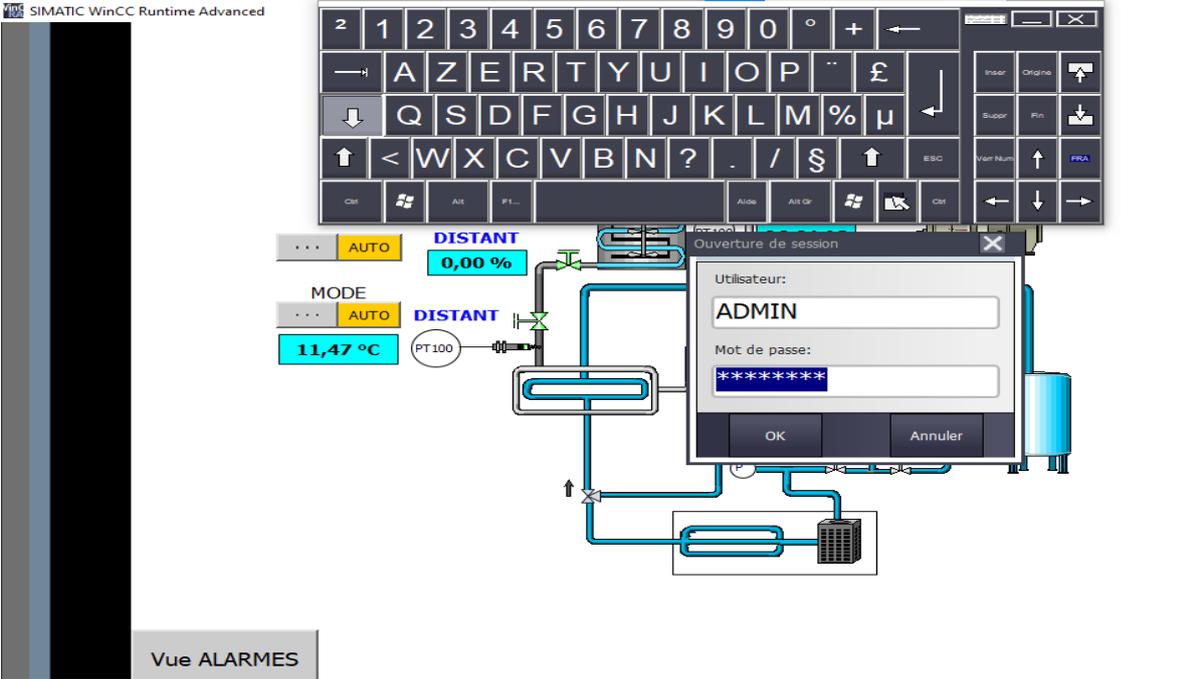


Figure 100: Exemple de création de mot de passe

VI. conclusion

Dans ce chapitre, nous avons utilisé les concepts de programmation et d' IHM pour développer le projet et à l'aide de tests bien choisis qui donnent les résultats attendus nous avons pu le valider. Ces tests concernent : la régulations en simulant le PID qui donnent des résultats satisfaisants, la validité des affichages sur l'écran utilisateur, l'affichage des alarmes et enfin la sécurisation du process.

Conclusion générale

Conclusion générale

Le système de supervision permet à l'opérateur de connaître l'état d'avancement du processus en temps réel et d'intervenir directement sur le pupitre de commande à partir de la salle de contrôle. Le logiciel de supervision Win cc flexible permet de mettre en œuvre le système de supervision de l'installation étudiée d'une manière simple, efficace et facile à utiliser.

Afin de remédier aux différents problèmes existants, dans la méthode actuelle (manuelle) de préparation polyvinyle acrylique, on a élaboré une méthode automatisée qui nécessite un ensemble d'instruments (exemple : API)

En dernier lieu, Nous avons effectué une simulation avec le logiciel PLCSIM et WinCC flexible qui nous a permis de valider le modèle et évaluer nos résultats obtenus.

Nous souhaitons que ce travail puisse être réalisé sur le procédé réel, et qu'il apportera un plus à l'entreprise, et une aide efficace aux prochaines promotions.

Références

- ([1]: Bernard PIGERON, HENRI MULLOT, Aain CHAIX, Louis FELIX, Yves Aubert « BOUCLES DE RÉGULATION », Livre(2005))
- ([2]: A.HAMIDA, cours de régulation industrielle ,(cours en PDF), 2020/2021.)
- ([3]: Daniel dindeleux , Bernard Poussery « PRATIQUE DE LA RÉGULATION INDUSTRIELLE », Livre)
- ([4]: Sakli MOUADH, Mémoire de fin d'études : «Système de régulation de niveau d'eau», Université de TUNISIE, promotion 2006/2007.)
- ([5]: www.novusautomation.com, consulté le: 09/06/2024)
- ([6]: M.MOURAD BOUDAH, Mémoire de fin d'études : «automatisation d'une ligne de préparation de jus à COJEK CEVITAL», Université de BEJAIA , promotion 2017.)
- ([7]: BENHAMZA MOHAMED FOUAD, Mémoire de fin d'études : «Etude et programmation du brûleur d'un four commandé par S7-300», Université de Biskra, promotion 2018/2019.)
- ([8] : https://cira-descartes.etab.ac-lyon.fr/spip/IMG/pdf/Affiche_Pt100v3.pdf, consulté le 15/04/2024)
- ([9]: Documentation technique du fournisseur , documentation interne de ENAP.)
- ([10] : <https://www.fujielectric.fr/blog/9-questions-a-se-poser-pour-choisir-un-capteur-de-pression-industriel/> consulté le :15/04/2024)
- ([11] : <https://dokumen.tips/documents/yaes-sb-qpak-sb-instcommopermain.html>, consulté le :12/02/2024)
- ([12]: Benfekir, cours d'automatisme industriel, (cours en PDF) 2019/2020
- ([13]: ZERROUK IMAD EDDINE et YEDDOU YOUCEF, Mémoire de fin d'études : «Supervision d'un niveau d'eau via un automate programmable :Etude et Conception », Université de TIARET , promotion 2021/2022.)
- ([14] : <https://www.enap.dz/> consulté le 25/02/2024)
- ([15]: M. Bertrand, Automates Programmable industriel , (document PDF).)
- ([16]: MICHEL. G, Les API, Architecture Et Application Des Automates Programmables Industriels. Dunod, Paris, [1987].)
- ([17]: <https://www.usinenouvelle.com/expo/simatic-s7-1200-controleur-modulaire-p196958.html>, consulté le : 15/04/2024.)
- ([18]: <https://www.google.com/search?q=definition+d%27un+automate+s71500&aq=chrome.2.69i57j35i39l2j0i512l7.13644j0j15&sourceid=chrome&ie=UTF-8>, consulté le : 15/04/2024.)
- ([19]: 01 1200 Initiation au programme du SIMATIC S7-1200.pdf.)

[[20]: Automatisation et supervision de l'unité de traitement des eaux par osmose inverse du complexe CEVITAL via le logiciel TIA portal v13 de SIEMENS.pdf)

[[21]: Automatisation et supervision d'une station de Thermolaquage par un automate S7-1200 complet.pdf; [12]:Benfekir, cours d'automatisme industriel,(cours en PDF))

[[22]: William Bolton, « LES AUTOMATES PROGRAMMABLES INDUSTRIELS », Livre, Paris :Dunod, 2010)

[[23] : Dr.Mohsen Ben Ammar : « Automates programmables industriels (Théorie et application), Document PDF publié le 09 mars 2013.)

[[24]: IZRAN EN Massinisa et AIT KHELIFA Hafid, Mémoire de fin d'études : «Automatisation d'une poseuse de poignée », Université de Béjaïa, promotion 2015.)

[[25]: ABDERRAZAK TAHMI et ZAKARIA TAHMI, Mémoire de fin d'études : «Automatisation d'une poseuse de poignée », Université de ALGER , promotion 2017.)

[[26]: Philippe GRARE et Imed KACEM, « AUTOMATISME, CE QU'IL FAUT SAVOIR »édition ellipses,250p, 2008.)

[[27] : Ammar Mzoughi, Ali Zitouni, Hatem Labidi, Mounir Ben Henda et Fethi Ayari « GENIE ELECTRIQUE » Manuel de cours 4ème année de l'enseignement secondaire tunisien, OMEGA EDITIONS, 2013.)

[[28]: Michel Grout et Patrick Salaün « INSTRUMENTATION INDUSTRIELLE, SPÉCIFICATION ET INSTALLATION DES CAPTEURS ET VANNES DE RÉGULATIONS », Livre, Paris : Dunod , 568 p, 2012)

[[29]: SEBKHI ROSA et RAHMANI SOUHILA, Mémoire de fin d'études : «régulation et supervision d'une station de création de vide à base des automates siemens », Université de Bejaia , promotion 2016.)

[[30]: REZOUG SOFIANE et SAHNOUNE HAFSA, Mémoire de fin d'études : «pilotage d'une cellule robotisée simulée à l'aide d'un automate programmable », Université de Bouira , promotion 2018/2019.)

[[31]:Siemens, SITRAIN Formation Automatisations et entrainements, cours S7-MAI Gamme SIMATIC S7, livre maintenance niveau 1 Siemens.)

[[32]= CHÉRIF CHIRAZ et BENZIANE MATTALAH MOHAMED, Mémoire de fin d'études : «Automatisation, Supervision et régulation de vapeur par Tia Portal dans un processus de raffinage du sucre », Université de Oran 2, promotion 2021/2022.)

[[33]:(<https://www.siemens.com/global/en/products/automation/industry-software/automation-software/tia-portal.html>, consulté le: 09/062024)

[[34]:<https://dspase.univ-ouargla.dz/handle/123456789/11759>,consulté le : le11/06/2024)

[[35]: <https://www.abcclim.net/types-daction.html>,consulé le :14/05/2024)

[[36] : document interne de laboratoire de l'entreprise ENAP -SIG- Mascara)

([37] : <https://www.monsieurpeinture.com/blog-composition-de-la-peinture/>consulté le 13/06/2024)

Résumé.

L'étude d'un système automatisé d'une station de fabrication de PVA fait l'objet de ce travail réalisé au cours de notre projet de fin d'études de Master. L'objectif est de mettre en œuvre l'automatisation et la supervision en utilisant l'API S7-1200 et le programmer en langage Ladder sous l'environnement logiciel TIA Portal. Notre système a été testé par le simulateur PLCSIM. Egalement, l'outil WinCC flexible a été utilisé pour configurer notre interface IHM pour la supervision en temps réel. Ce travail nous a permis de nous familiariser beaucoup plus avec l'API S7-1200, de maîtriser le langage Ladder ainsi que l'explorateur WinCC, ce qui représente pour nous une expérience très importante dans le domaine de l'automatisation/supervision industrielle.

Mots clés: station de fabrication de PVA, Automatisation, Supervision, API, TIA Portal, Ladder

Abstract.

The study of an automated system of a manufacturing station of PVA is the subject of this work carried out during our Master's degree project. The objective is to implement automation and supervision using the S7-1200 API and program it in Ladder language under the TIA Portal software environment. Our system has been tested by PLCSIM simulator. Also, the WinCC flexible tool was used to configure our HMI interface for real-time supervision. This work allowed us to familiarize ourselves with the S7-1200 API, to master the Ladder language as well as the WinCC explorer, which represents for us a very important experience in the field of industrial automation/supervision. Key words: Manufacturing station of PVA, Automation, Supervision, PLC, TIA Portal, Ladder.

تلخيص

إن دراسة النظام الآلي لمحطة تصنيع PVA هي موضوع هذا العمل الذي تم تنفيذه خلال مشروع درجة الماجستير لدينا. الهدف هو تنفيذ الأتمتة والإشراف باستخدام S7-1200 API وبرمجتها بلغة Ladder ضمن بيئة برنامج TIA Portal. تم اختبار نظامنا بواسطة جهاز محاكاة PLCSIM. كما تم استخدام أداة WinCC المرنة لتكوين واجهة HMI الخاصة بنا للإشراف في الوقت الفعلي. لقد أتاح لنا هذا العمل أن نصبح أكثر دراية بـ S7-1200 API، وإتقان لغة Ladder بالإضافة إلى مستكشف WinCC، الذي يمثل بالنسبة لنا تجربة مهمة جدًا في مجال الأتمتة/الإشراف الصناعي.

الكلمات المفتاحية: محطة تصنيع PVA، الأتمتة والإشراف، API، بوابة TIA، سلم