

République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieure et de la Recherche Scientifique

Université Abderrahmane Mira

Faculté de Technologie



Département d'Automatique, Télécommunication et d'Electronique

Projet de Fin d'Etudes

Pour l'obtention du diplôme de Master en Électronique

Option : Instrumentation

Thème

Étude de régulation en automatique de la boucle de chauffage d'un bac de stockage d'huile brute raffinerie d'huile. Cevital-Beiaia

Préparé par :

Melle : Bordjah Céline

Melle : Abdiche Yasmina

Encadreur:

Mr Berboucha.Ali

Co- promoteur :

Mr Charfaoui. Faouzi

Examiné par :

Mr Hanfoug. Saleh

Mme Ouali. Kahina

Année universitaire : 2021/2022

Remerciements

Avant tout, il ne saurait question d'entreprendre les remerciements sans avoir remercié le BONDIEU de nous avoir permis de réaliser ce travail.

Nous tenons à remercier :

Tout d'abord à nos chers parents, d'avoir été là toutes ces années de nous avoir aidé à faire ce très long parcours, et de nous avoir guidé vers le droit chemin.

Nous remercions infiniment notre encadreur Mr BERBOUCHA pour sa confiance, pour son aide si précieuse et surtout infini, pour son côté humain, son indulgence, tout sa compétence, sa compréhension et surtout son immense patience qui nous permis d'achever ce modeste travail. Nous lui témoignons ainsi toute notre gratitude pour son aide.

Nous remercions vont particulier à Mr Charfaoui, Faouzi notre prometteur de l'entreprise CEVITAL, qui nous a diriger et nous aider beaucoup afin de réaliser ce travail.

Nous remercions également les membres du jury qui se sont donné la peine de lire et de le réviser notre mémoire.

Nous désirons aussi à remercier les enseignants de l'université d'Abderrahmane Mira, qui nous ont fourni les outils nécessaires à réussite de nos études universitaires.

Nous voudrions, enfin, dit un grand merci pour nos amis, sœurs, frères qui nous ont encourages pour faire notre fameux mémoire.



Dédicace



J'ai le grand honneur de dédier ce travail :

- *À mon cher père*

Tu as toujours à mes côtés pour me soutenir et m'encourager. Que ce travail traduit ma gratitude et mon affection.

- *À ma chère mère*

Quoi que je fasse ou que je dise .je ne serai point te remercier comme il se doit. Ton affection me couvre, ta bienveillance me guide et ta présence à mes cotés a toujours été ma source de force pour affronter les différents obstacles

- *À mon cher frère Habib*

- *À mes chères Sœurs ; Faiza, Yasmine, Meriem.*

- *À mes ancêtres que Dieu leur fasse miséricorde.*

- *À mes grands-parents vivants, que Dieu les préserve.*

• *A toutes la famille Bordjah et Izem surtout mes tantes Zakia, Rabia, Razika, Farida, Kahina, et mes oncles Nassim, Atman et Cherif, Halim.*

• *À mes cousins et cousines, Yasma, Lilia, Ania, Chahinaz, Mahmoud, Yacine, Zaid, Saleh.*

- *À ma chère amie intime Sara Kherbache et sa famille.*

- *À ma cher binôme Abdiche Yasmina.*


- *À toute la promotion Instrumentation 2022.*

- *Et à tous ceux que j'aime.*

Melle. Céline



Dédicace



C'est avec une profonde gratitude et sincères mots que je tiens à dédicace ce travail à:

Les êtres les plus chères de ma vie ; mes parents. Aucune dédicace ne saurait exprimer mon respect et mon amour éternel que je vous porte, ma considération pour toutes ces années de sacrifices que vous avez consentie pour moi, depuis ma naissance, durant mon enfance et même à l'âge adulte. Vous qui m'ont guidé à travers tout le long du chemin. Je tiens à vous remercier pour votre amour, patience, soutien, encouragement et surtout les valeurs Nobles et éducation.

A la mémoire de mon père HANAFI, Ce travail est dédié à mon père, décédé trop tôt, qui m'a toujours poussée et motivé dans mes études j'espère que du monde qui sien maintenant, il est apprécié et humble geste comme preuve de reconnaissance de part d'une fille qui a toujours prié pour le salut de son âme. Puisse dieu, le tout puissant, l'avoir en sainte miséricorde.

À mon frère Amine; mon exemple, mon ange gardien, source d'amour ; de Courage et de persévérance. Je tiens à te remercier d'avoir été toujours là pour moi, ton soutien dans les bons et mauvais moments dans cette vie Que le bonheur soit ton compagnon et la réussite ton logis.

À ma chère sœur Nedjma; Aucune dédicace ne peut exprimer mon amour et ma gratitude de t'avoir comme sœur.

*Je ne pourrais jamais imaginer la vie sans toi, tu comptes énormément pour moi, tu es la sœur qui assure sone rôle comme il faut, je n'oublierais jamais ton encouragement et ton soutien le long de mes études, je t'estime beaucoup et je t'aime beaucoup.
Je te souhaite beaucoup de succès, de prospérité et une vie pleine de joie et de bonheur.*

Melle. YASMINA



À ma belle-sœur Célia, Cela fait maintenant l'ans que tu partages la vie de mon frère, celle de notre famille, et la mienne par la même occasion.

Ma belle-sœur, mais aussi ma grande Sœur. Je sais enfin ce que sais que le bonheur d'avoir une grande sœur sur laquelle on peut compter, moi qui n'en ai jamais eu. Je te dis merci et je te souhaite bonheur, réussite et prospérité.

À mon neveu Hanafi Abdiche

Avoir un neveu est le plus beau cadeau qu'un frère puisse vous faire. Tes petites mains, ton envie de parcourir le monde, ton enthousiasme, tes sourires, tes yeux brillants sont incomparables. Tu as apporté beaucoup de bonheur à notre famille. Je t'aime

À mes grands-parents paternels et maternels, Que ce modeste travail, soit l'expression des vœux que vous n'avez cessé de formuler dans vos prières. Que Dieu vous préserve santé et longue vie.

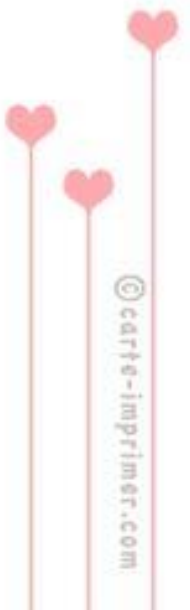
À ma cher binôme céline bordjah

A la plus chère personne de ma vie qui a pu être, un père, un frère, une mère, une sœur, un tout pour moi je le souhaite tout le bonheur du monde.

À notre encadreur, ainsi tous les membres de la faculté de Technologie et de département Génie Électrique. Enfin je dédie

Ce travail à moi, à mon travail, mes efforts, mes ambitions, ma patience, courage et volonté. Et ici se trouve le résultat. À toutes les personnes, qui par un geste, une phrase où même un mot m'a donné la force et le courage de continuer à me battre.

Melle. YASMINA



Abréviations

API: Automate Programmable Industriel

CP: Processeur de communication

CPU: Unité Centrale de l'automate programmable

CONT_C : Contrôleur Continu

DB: Bloc de donnée

EPROM: Memoire Mort Programmable effaceable

EEPROM: Memoire Mort Programmable Effaceable Électriquement

FB: Bloc des fonctions

FBD: schéma fonctionnel

FC: Fonction

FM: Module de fonction

HMI : Interface Homme-Machine

IL: Liste d'instructions

LD: Diagrammes échelle

MPI: interface multipoint

OB: Bloc d'Organisation

Op: Partie Opérative

PC: Partie Commande

P: Proportionnel

PI: Proportionnel Intégral

PID : Proportionnel Intégral Dérivé

RAM: Memoire vive

ROM: Memoire mort.

SPA: Système Programmable autorisé

SM: Modules de simulation

SFC: Diagramme fonctionnel sequentiel (Grafcet)

ST: Text structuré

TOR: Tout Ou Rien

USA: États Unis

Liste des figures

Figure I.1 : Différentes structures de réchauffement d'huile.	7
Figure I.2: Bac d'huile.....	9
Figure I.3: Agitateur dynamique	10
Figure I.4: Capteur de niveau	10
Figure I.5: Pompe centrifuge	11
Figure I.6: Principe de fonctionnement d'une pompe centrifuge	12
Figure I.7:Echangeur a plaque.....	13
Figure I.8: Vanne de régulation.....	14
Figure I.9: Vanne tout ou rien (TOR).....	15
Figure I.10: Purgeur de vapeur	15
Figure I.11: Capteur de pression	16
Figure I.12: Capteur de température.....	17
Figure II.1: Automates Mitsubishi	19
Figure II.2: Automate programmable Allen-Bradley.....	19
Figure II.3 : Automate Omron.....	19
Figure II.4: Automate Siemens.....	20
Figure II.5 : Structure de système automatisé	21
Figure II.6 : Automate compact ABB	23
Figure II.7 : Automate modulaire Mitsubishi.....	23
Figure II.8 : Aspect extérieur.....	24
Figure II.9: Structure interne de l'API	25
Figure II.10: langage à contacte (Ladder)	26
Figure II.11: Cycle d'exécution d'un programme	27
Figure II.12 : Temps de réponse totale (TRT).....	28
Figure II.13 : Automate CROUZET Millénium II	29
Figure II.14 : Logiciel de programmation SIMATIC S7-300	30
Figure II.15 : Architecture réelle Automate SIMATIC S7-300	30
Figure II.16 : Disposition des modules de l'API S7-300	31
Figure II.17 : Vue de CPU S7 -300(312).	32
Figure III.1: Interface du logiciel STEP7.	36
Figure III.2: Fenêtre de Création d'un bloc d'Organisation.....	37
Figure III.3: Fenêtre de Création d'une fonction.....	39
Figure III.4: Fenêtre de Création d'un bloc de donnée	39
Figure III.5: Structure d'un programme	40
Figure III.6: Création d'un projet	41
Figure III.7 : sélection du Rack	42
Figure III.8 : Choix de la CPU	42
Figure III.9 : Choix des modules d'entrées et sorties	43
Figure III.10: Choix d'alimentation	43
Figure III.11 : Table mnémonique.....	44

Figure III.12: Chargement de programme dans la CPU	45
Figure III.13: Simulation de programme	46
Figure III.14: Visualisation cas actif	47
Figure III.15 : Visualisation cas inactif	47
Figure III.16 : Logiciel WinCC Flexible	48
Figure III.17 : Choix type de pupitre	49
Figure III.18 : Aperçu de la fenêtre WinCC flexible	49
Figure III.19 : liaison entre la station S7-300 et la station HMI.....	51
Figure III.20 : Simulateur S7-PLC SIM	54
Figure III.22 : Graphe de la valeur manuelle sur WinCC flexible	55
Figure III.23: Graphe de la regulation PID sur WinCC flexible	56

Liste des tableaux

Tableau I.1 : Étude des actions PID	5
Tableau II.1 : Position du commutateur du mode de fonctionnement	33

Sommaire

Remerciements

Dédicace

Abréviations

Introduction Générale 1

Chapitre I : Description de système de régulation avec la boucle de chauffage

I. 1 Introduction 2

I.2 Définition de la régulation automatique 2

I.3 objectifs de la régulation Automatique 2

I.4 Types de la régulation..... 3

 I.4.1 Régulation tout ou rien (TOR) 3

 I.4.2 Régulation numérique 3

 I.4.3 Régulation analogique 3

I.5 Critère de performance d'une régulation 3

I.6 Régulation PID 4

 I.6.1 Définition..... 4

 I.6.2 Étude des actions PID 4

 I.6. 3 Rôle des paramètres PID 5

 I.6.4 FB 41 CONT-C 5

 I.6.4.1 Description **6**

 I.6.4.2 États de fonctionnement **6**

 I.6.4.3 Information d'erreur **6**

I.7 Description du système actuel..... 7

I.8 Cahier de charges..... 7

I.9 Description du problème 8

I.10 solution proposé au problème 8

 I.10.1 Bac d'huile 9

 I.10.1.1 Agitateur Dynamique **9**

I.10.1.2 Capteur de niveau.....	10
I.10.2 pompe centrifuge	10
I.10.2.1 principe de fonctionnement	11
I.10.3 Échangeur á plaque	12
I .10.3.1 principe de fonctionnement	13
I.10.4 Vanne de régulation	14
I.10.5 Vanne Tout ou Rien (Tor)	14
I.10.6 Purgeur de vapeur.....	15
I.10.6.1 Principe de fonctionnement	16
I.11 Capteurs	16
I.11.1 Définition.....	16
I.11.2 Types de capteurs utilisés dans l'installation	16
I .11.2.1Capteur de pression.....	16
I .11.2.2 Capteur de température	17
I.12 Conclusion.....	17

Chapitre II : Automates programmables industriels

II.1 Introduction.....	18
II.2 Systèmes automatisés	20
II.2.1Définition	20
II.2.2 But de l'automatisation	20
II.2.3 Structure d'un système automatisé	21
II.3 Automate programmables industriel	21
II.3.1 Définition	21
II.3.2 Caractéristique de l'automate programmable	22
II.3.3 Domaine d'utilisation de l'automate programmable	22
II.3.4 Nature des informations traitées par l'automate.....	22
II.3.5 Description de l'automate programmable :.....	22
II.3.5.1 Aspect extérieur :	22
II.3.5.2 structure interne	24
II.3.6 Langage de programmation d'un API.....	25
II.3.7 Traitement du programme automate.....	27
II.3.7 Critère de choix d'un automate programmable.....	28
II.4 Présentation de l'automate S7-300	29

II.4.1 Description de l'automate S7-300	30
II.4.2 Structure de l'automate S7-300.....	31
II.4.2.1 Modules d'alimentation PS :	32
II.4.2.2Présentation de la CPU (S7-300)	32
II.4.2.3 Module de couplage (IM):.....	34
II.4.2.4 Module de signaux d'entrées/ sortie SM	34
II.4.2.5 Module de simulation (SM) :	34
II.4.2.6Module de fonction (FM).....	35
II.4.2.7: Modules de communication CP	35
II.5 Conclusion	35

Chapitre III: Programmation et supervision

III.1 Introduction.....	36
III.2 Description du logiciel STEP7.....	36
III.3 Structure du programme en STEP7	37
III.3.1 Bloc d'organisation (OB):.....	37
III.3.2 Blocs d'organisation de traitement cyclique (OB1).....	38
III.3.3 Bloc fonctionnel (FB).....	38
III.3.4 Fonction (FC) système.....	38
III.3.5 Blocs de données (DB).....	39
III.3.5.1 Blocs de données d'instance (DB d'instance).....	40
III.3.5 .2 Blocs de données globaux (DB)	40
III.4 Réalisation du programme	40
III.4.1 Création de projet sous STEP7.....	40
III.4.2 Configuration matérielle	41
III.5.2.1 Étape 1: sélection du Rack.....	41
III.5.2.2 Étape 2: Sélection de la CPU	42
III.5.2.3 Étape 3: sélectionné les entée /sortie.....	42
III.5.2.4 Étape 4: Sélection de l'alimentation	43
III.5 Création de la table des mnémoniques	44
III.6 Simulation.....	44
III.6.1 Chargement du programme	44
III.6.2 Présentation de PLCSIM.....	45

III.7 Description de logiciel Win CC flexible	48
III.7.1 Utilisation de SIMATIC Win CC flexible	48
III.7.2 Présentation du système Win CC flexible	48
III.7.3 Intégration dans Simatic STEP7	50
III.8 Partie programmation.....	51
III.8.1 Programmation du mode manuel et du mode automatique	51
III.9 Simulation.....	54
III.9.1 Simulation du programme sous STEP7 AVEC PLCSIM.....	54
III.9.2 Simulation du programme sous WinCC	55
III.9.3 Vue de régulateur PID	55
III.10 Conclusion	56
Conclusion générale	57



Introduction générale

Introduction Générale

L'automatisation est la priorité absolue dans les industries modernes, le débat sur ce phénomène est toujours sujet à controverse vu que la machine a tendance à remplacer les opérations manuelles (homme) dans plusieurs tâches, il ne peut hélas assurer les tâches que fait la machine dans plusieurs domaines.

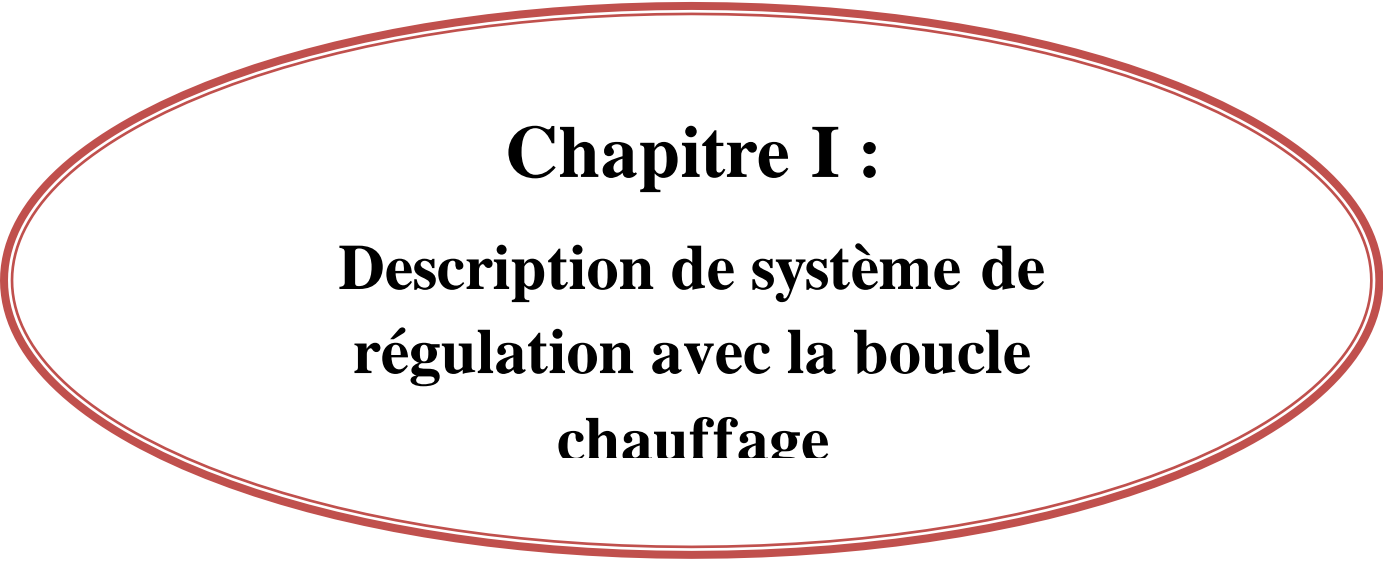
Les automates programmables industriels sont opérationnels dans nombre de domaines, de l'aéronautique jusqu'à l'électroménager. de par la simplicité de leur mise en œuvre et de leur implantation ils occupent une place importante dans les technologies utilisées en automatisation.

Notre stage effectué au sein du complexe CEVITAL et plus précisément à l'unité raffinerie d'huile, nous a donné une idée générale sur l'importance de l'automatisation des machines qui suit leurs bons fonctionnements, et l'intervention rapide et efficace en cas de défaut. L'étude et l'amélioration du programme d'automatisation de boucle chauffage bac stockage au sein de la raffinerie d'huile, est le cœur de cette étude.

Notre objectif consiste à réaliser un programme sous STEP7, avec un automate Programmable SIEMENS S7-300, afin de superviser la variation de température de la boucle de chauffage du bac stockage d'huile brute établie au sein de l'entreprise CEVITAL Bejaia, suivi d'une supervision sous Win CC flexible.

Suite à cela, le présent mémoire décrit l'essentiel du travail réalisé lors de ce stage, il comporte quatre chapitres :

- ✓ Le premier chapitre présentera une généralité sur la régulation, objectif de la régulation et son rôle dans l'industrie puis on donnera une description fonctionnelle du système étudié.
- ✓ Le deuxième chapitre portera sur une présentation complète d'un système automatisé en suite on passera vers la présentation générale de l'automate programmable.
- ✓ Le troisième chapitre présentera les étapes qu'on a suivies pour la programmation sous STEP7, et une supervision sous Win CC flexible.
- ✓ Enfin on terminera par une conclusion générale.



Chapitre I :
Description de système de
régulation avec la boucle
chauffage

Chapitre I : Description de système de régulation avec la boucle de chauffage

I. 1 Introduction

Le terme de régulation renvoie dans son sens concret à une discipline technique qui se rattache au plan scientifique à l'automatique.

La Régulation est une partie de la science technique appelée Automatique. On considère généralement que l'automatique (et donc la régulation) a débuté dans les années 1930, avec les premiers asservissements de position et surtout une première définition de la stabilité, naturellement, des systèmes à fonctionnement « autonome » existaient auparavant (les automates), mais ils n'étaient pas théorisés. Après ces premiers pas, tout s'accéléra, avec le développement des premières méthodes de synthèse de correcteurs au cours de la décennie 1940-1950, puis dès 1960, avec l'explosion de l'informatique.

I.2 Définition de la régulation automatique

La régulation automatique fait partie des sciences de L'ingénieur. Cette discipline traite la modélisation, l'analyse, la commande et la régulation des systèmes dynamiques. Elle a pour fondements théoriques les mathématiques, la théorie du signal et l'informatique théorique. L'automatique permet l'automatisation des tâches par des machines fonctionnant sans intervention humaine. On parle alors de système asservi ou régulé. L'état désiré du système est nommé la consigne [1].

I.3 objectifs de la régulation Automatique

Réguler une grandeur, c'est obtenir d'elle un comportement donné, dans un environnement susceptible de présenter des variations. On ne peut pas parler de principe de régulation sans parler des lois de commandes. En fait, les grandeurs physiques commandées varient de façon continue dans le temps. Pour celles qui ne présentent que 2 états (système binaire ou « tout ou rien », tel les feux de signalisation, les commandes d'ascenseurs, de transfert de pièces par convoyeurs, etc. ...) on utilise une autre approche différente a la structure de boucle utilisée dans la plupart des systèmes.

Les systèmes automatiques assurent en fait deux types de fonctions :

1. Maintenir la grandeur commandée, ou grandeur réglée, à une valeur de référence malgré les variations des conditions extérieures; on parle de la régulation en sens strict.
2. Répondre à des changements en temps réel d'objectif, ou à un objectif variable tel que la poursuite de cible, on parle d'asservissement. [2]

I.4 Types de la régulation

I.4.1 Régulation tout ou rien (TOR)

Un régulateur «tout ou rien» est un régulateur qui élabore une action de commande discontinue qui prend deux positions ou deux états 0 et 1 (ou 0 et 100%). On les appelle 'on-off control' ou 'two Steps Controller'. [3]

I.4.2 Régulation numérique

C'est le type de régulation où le signal du régulateur et la mesure varient d'une manière continue dans le temps. Le mode d'action analogique le plus simple est l'action proportionnelle est réalisée par un régulateur (P), il convient en générale aux installations ayant une grande inertie.

I.4.3 Régulation analogique

C'est un processus qui fonctionne à partir d'un signal continu spécifique sous forme de tension nécessaire à la réalisation des régulateurs analogiques. Les valeurs des grandeurs physiques constituant les signaux analogiques doivent être représentés par des nombres. [4]

I.5 Critère de performance d'une régulation

Les performances d'une régulation peuvent se définir à partir de l'allure du signal de mesure suite à un échelon de consigne.

Notons toutefois que les critères de performance classique peuvent se résumer comme suit :

- **Stabilité** : Cette condition est impérative mais avec un certain degré de stabilité (marge de sécurité). En général on impose une marge de gain de 2 à 2.5. L'utilisateur parle en termes de «pompage».

- **Précision** : L'exploitant demande à ce que le système possède une bonne précision en régime permanent d'où une nécessité de mettre un régulateur PI ou d'afficher un gain important dans le cas d'un régulateur P.
- **Rapidité** : On demande en pratique que le système soit capable rapidement de compenser les perturbations et de bien suivre la consigne. [5]

I.6 Régulation PID

I.6.1 Définition

C'est un organe de contrôle permettant d'effectuer une régulation en boucle fermée d'un système industriel. C'est le régulateur le plus utilisé dans l'industrie, et il permet de contrôler un grand nombre de procédés.

L'erreur observée est la différence entre la consigne et la mesure. Le PID permet 3 actions en fonction de cette erreur :

- **Une action Proportionnelle** : l'erreur est multipliée par un gain G .
- **Une action Intégrale** : l'erreur est intégrée sur un intervalle de temps S , puis divisée par un gain T_i .
- **Une action Dérivée** : l'erreur est dérivée suivant un temps S , puis multipliée par un gain T_d . [6]

I.6.2 Étude des actions PID

On résume dans ce tableau les avantages et les limitations des actions de base des régulations PID :

Tableau I.1: Étude des actions PID [7]

Action	Points forts	Points faible
P	Action instantanée	Ne permet pas d'annuler une erreur statique mais permet de la réduire
I	Annule l'erreur statique	Action lente, ralentit le système (effet déstabilisant)
D	Action très dynamique Améliore la rapidité Apporte un effet stabilisant	Sensibilité aux bruits, forte sollicitation de l'organe de commande

I.6.3 Rôle des paramètres PID

- **L'action proportionnelle:** l'action proportionnelle applique une correction instantanée pour tout écart entre la mesure et la consigne, plus que la perturbation est grande, plus la correction apportée est grande. Cette composante seule ne permet pas une grande précision surtout dans les systèmes à faible inertie, comme dans le traitement de l'aire, cette rapidité d'action engendre un phénomène appelé le pompage.
- **L'action intégrale:** cette action apporte une notion de temps d'intégration à la correction, cette notion de temps s'exprime généralement en seconde.

Cette action est complémentaire à l'action proportionnelle, elle permet de stabiliser dans le temps l'action proportionnelle, plus l'erreur mesurée est constante plus la correction est constante.

- **L'action dérivée:** cette action permet d'anticiper la réponse de la régulation en cas de la perturbation rapide ou de modification de consigne ce qui améliore la stabilité du système. On peut donc dire que cette composante permet de compenser tout dépassement excessif de la consigne. [8]

I.6.4 FB 41 CONT-C

Le bloc FB 41 « CONT_C » (continuous Controller) sert à réguler des processus industriels à grandeurs d'entrée et de sortie continues sur les automates programmables

SIMATIC S7. Le paramétrage nous permet d'activer ou de désactiver des fonctions partielles du régulateur PID et donc d'adapter ce dernier au système régulé. [9]

I.6.4.1 Description

En plus des fonctions traitant la consigne et la mesure, le FB réalise un régulateur PID prêt à l'emploi avec sortie continue de la grandeur de réglage et possibilité d'influencer à la main la valeur de réglage. [9]

Selon le type de CPU, il sera mis en œuvre grâce au SF FB41 (pour les CPU) ou au FB (pour les CPU sans interface profibus). Il propose les fonctions partielles suivantes :

- Branche de consigne.
- Branche de mesure.
- Formation du signal d'erreur.
- Algorithme PID.
- Traitement de la valeur de réglage manuelle.
- Traitement de la valeur de réglage.
- Action anticipatrice.

I.6.4.2 États de fonctionnement

Démarrage et redémarrage : le bloc FB 41 « CONT-C » dispose d'un sous-programme de démarrage qui s'exécute quand le paramètre d'entrée COM-RST=TRUE. A la mise en route, l'intégrateur est positionné de façon interne sur la valeur d'initialisation I-ITVAL. En cas d'appel dans un niveau d'alarme d'horloge, il continue à travailler à partir de cette valeur. Toutes les autres sorties sont positionnées sur leur valeur par défaut. [9]

I.6.4.3 Information d'erreur

Ce bloc ne procède pas à un contrôle interne d'erreur. Le mot indicateur RET-VAL n'est pas employé. [9]. Pour plus d'information sur FB 41 CONT-C consulter l'annexe 1

I.7 Description du système actuel

Le bac d'huile brute sera rempli via la vanne de remplissage par l'huile brute. Cette dernière, sera réchauffée par la vapeur à 9bar et à la température de 80°C, la pompe va aspirer, par la suite, l'huile brute qui sera dirigée vers l'échangeur.

L'huile brute revient vers le bac après un échange thermique avec un liquide à une température inférieure à la consigne désirée, la vanne régulatrice s'ouvre pour atteindre la température de la consigne.

Le schéma ci-dessous explique le réchauffement de l'huile brute dans le bac via la vanne régulatrice :

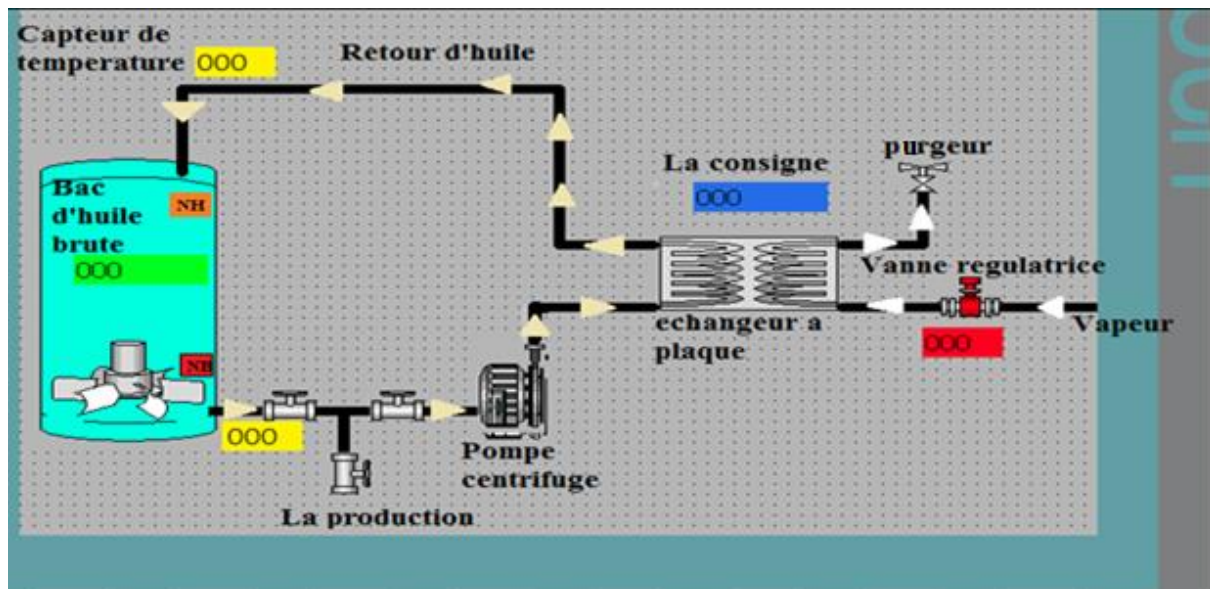


Figure I.1: Différentes structures de réchauffement d'huile.

I.8 Cahier de charges

Nous avons élaboré ce cahier de charge afin de pouvoir réchauffer l'huile brute dans le bac avec une bonne régulation de température en automatisant la vanne de régulation.

À titre d'information, le bac est équipé par

- Des agitateurs frontaux et Dynamique.
- Un détecteur de niveau.
- Vanne régulatrice.

- Vanne Tout ou Rien (TOR).
- Pompe centrifuge.
- Échangeur à plaque.
- Purgeur à flotteur.
- Indicateur de pression.
- Transmetteur de température.

I.9 Description du problème

Au départ notre système est composé de :

- Un bac d'huile, une pompe qui démarre à l'indication d'absence de niveau bas.
- Une vanne thermostatique.
- Agitateur dynamique.
- Transmetteur de température.

Pour réchauffer l'huile dans un bac on a utilisé un serpentant en cuivre, mais avec le temps, on a remarqué que l'huile se gâtait, et après enquête, on a découvert qu'il y a des trous dans le serpentant et des fuites de vapeur qui se transforme en eau, l'eau se mélange avec l'huile ce qui a entraîné l'endommagement d'huile et baissé la température désirée. Dans ce cas la pompe est impossible d'aspirer l'huile mais capable uniquement d'aspirer de l'eau, tel est le problème posé.

I.10 solution proposé au problème

Pour remédier à ce problème, c'est-à-dire trouver une solution qui permet le bon fonctionnement de réchauffage, on a installé un nouveau système on dehors du bac, on a ajouté un échangeur à plaque, cette dernière permet de réchauffer l'huile à une température demandée afin de l'utiliser. Des options ont été ajoutées pour l'activation ou la désactivation des actions de régulateur PID, en cas de perte des paramètres de cette régulation.

I.10.1 Bac d'huile

Un bac c'est un capacitif fixe destiné au stockage d'huile, contient des agitateurs dynamiques, capteur de niveau (Haute, bas). Ce dernier peut stoker jusqu'un 7000 tonnes d'huile brute.



Figure I.2: Bac d'huile

I.10.1.1 Agitateur Dynamique

L'agitateur dynamique est un outil très efficace pour la circulation ou le mélange de tous types de fluides, de viscosité faible ou élevée, avec ou sans particules solides, pour petits ou grands réservoirs. [10]

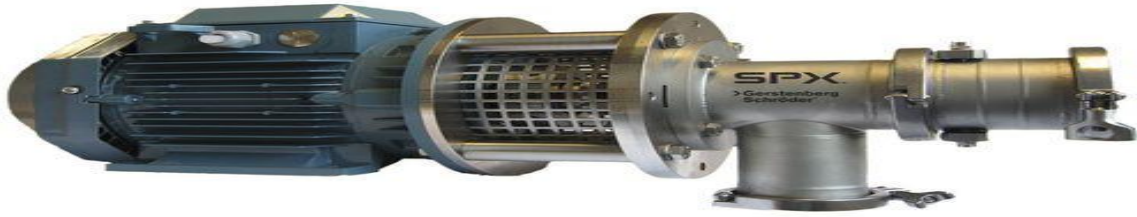


Figure I.3: Agitateur dynamique

I.10.1.2 Capteur de niveau

Un capteur de niveau est un dispositif électronique qui permet de mesurer la hauteur du matériau, en général du liquide, dans un réservoir ou un autre récipient. [11]



Figure I.4: Capteur de niveau

I.10.2 pompe centrifuge

Les pompes centrifuges sont des appareils qui sont utilisés pour transporter des fluides par la conversion de l'énergie cinétique de rotation en énergie hydrodynamique de l'écoulement de fluide. [12]

Une pompe centrifuge est une machine rotative qui pompe un liquide en le forçant au travers d'une roue à aube ou d'une hélice appelée impulser (souvent nommée improprement

turbine). C'est le type de pompe industrielle le plus commun. Par l'effet de la rotation de l'impulser, le fluide pompé est aspiré axialement dans la pompe, puis accéléré radialement, et enfin refoulé tangentielllement. la plaque signalétique de ce genre de pompe est illustrée dans l'annexe [13].(voir l'annexe 3).



Figure I.5: Pompe centrifuge

I.10.2.1 principe de fonctionnement

Dans la volute de la pompe, le fluide pénètre axialement dans la pompe par l'œil de la roue (zone à basse pression) qui tourne à grande vitesse. Lorsque la roue et les pales tournent, elles transmettent la quantité de mouvement au fluide entrant. Le fluide accélère radialement vers l'extérieur à partir de la pompe et un vide est créé au niveau de l'œil de la roue qui attire en permanence plus de fluide dans la pompe. À mesure que la vitesse du fluide augmente, son énergie cinétique augmente. Un fluide à haute énergie cinétique est forcé hors de la zone de la roue et pénètre dans la volute. Dans la volute, le fluide s'écoule dans une section de plus en plus importante, où l'énergie cinétique est convertie en pression de fluide (selon le principe de Bernoulli).

Les pales de la turbine sont généralement incurvées vers l'arrière, mais il existe également des conceptions de pales radiales et incurvées vers l'avant. La pression de sortie

varie légèrement en fonction de la conception utilisée. Les pales peuvent être ouvertes ou fermées. De plus, le diffuseur peut être équipé d'ailettes fixes pour aider à guider le flux vers la sortie. L'énergie transférée au liquide correspond à la vitesse au bord de la roue. Plus la roue tourne vite ou plus la roue est grosse, plus la vitesse sera élevée. [14]

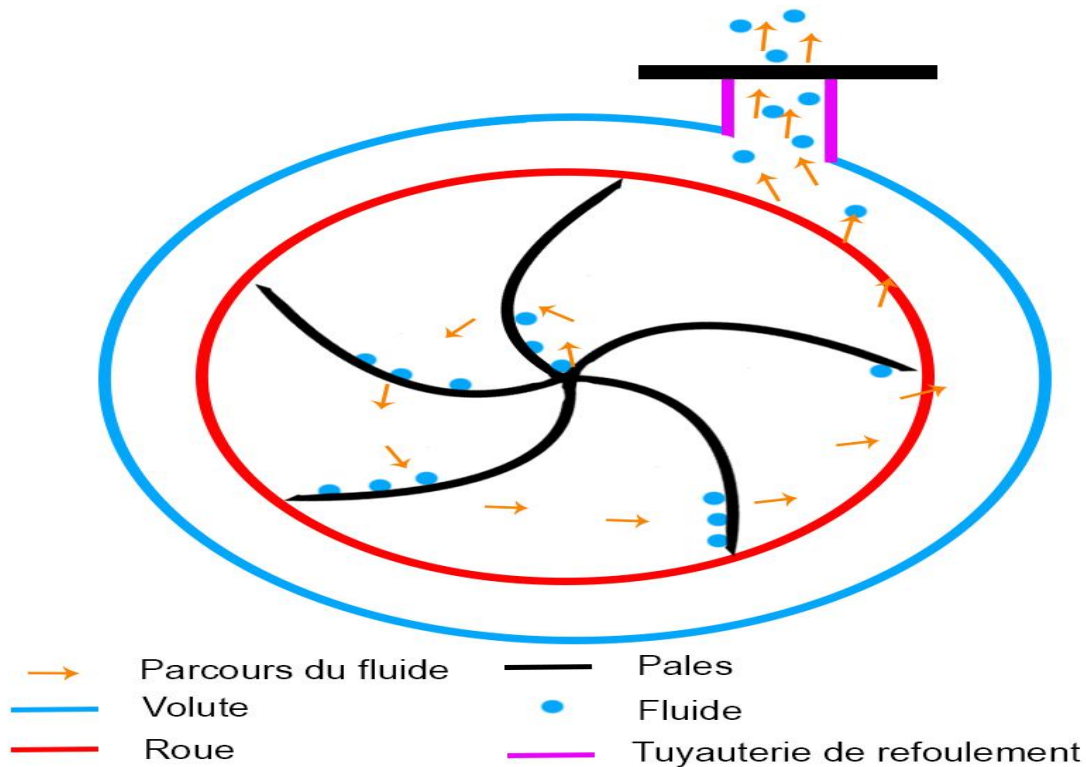


Figure I.6: Principe de fonctionnement d'une pompe centrifuge

I.10.3 Échangeur à plaque

Un échangeur à plaque est constitué d'un ensemble de plaques métalliques embouties à travers lesquelles s'effectue le transfert de chaleur entre deux fluides, il est constitué de :

- **Les plaques** : sont serrées entre un bâti fixe et un bâti mobile. Elles sont positionnées et guidées par deux barres support inférieure et supérieure.
- **Les bâtis** : maintiennent les plaques serrées au moyen de tirants. Les deux barres guides sont-elles mêmes supportées par un pied support.
- **Un joint** : par plaque assure l'étanchéité de l'échangeur ainsi que la répartition des fluides dans les canaux formés par les deux plaques.

. Le raccordement des fluides s'effectue généralement sur le bâti fixe de l'appareil. L'emboutissage des plaques en chevrons ou en cannelures (industries alimentaires) favorise la turbulence des fluides et assure une bonne tenue à la pression. [17].(voir l'annexe 3).



Figure I.7:Echangeur a plaque

I.10.3.1 principe de fonctionnement

Dans un échangeur, les courants entre les fluides primaires et secondaires peuvent être parallèles, opposés ou croisés. Les deux fluides convergent dans des canaux séparés l'un pair et le second impair, à proximité afin que l'un réchauffe ou refroidisse l'autre.

Il existe également trois modes d'écoulement différents suivant les applications:

1. Une circulation à Co-courants ou anti-méthodique : l'écoulement est parallèle, les fluides convergent vers la même direction. L'entrée des deux fluides se situe du même côté de l'échangeur.
2. Une circulation à contre-courants ou méthodique : l'écoulement est parallèle, mais les fluides traversent les canaux dans des directions inverses. Il s'agit de la meilleure configuration pour optimiser les performances d'un échangeur à plaques.

3. Une circulation à courants croisés : l'écoulement est alors perpendiculaire entre les deux fluides. [16]

I.10.4 Vanne de régulation

Une vanne de régulation est un actionneur qui associe un corps de vanne avec une motorisation électrique, voire pneumatique dans des applications industrielles. La vanne de régulation est souvent modulante et plus rarement 'TOR' Tout ou Rien. Son ouverture et le débit dans ses voies varient en fonction d'une loi de régulation. [17]

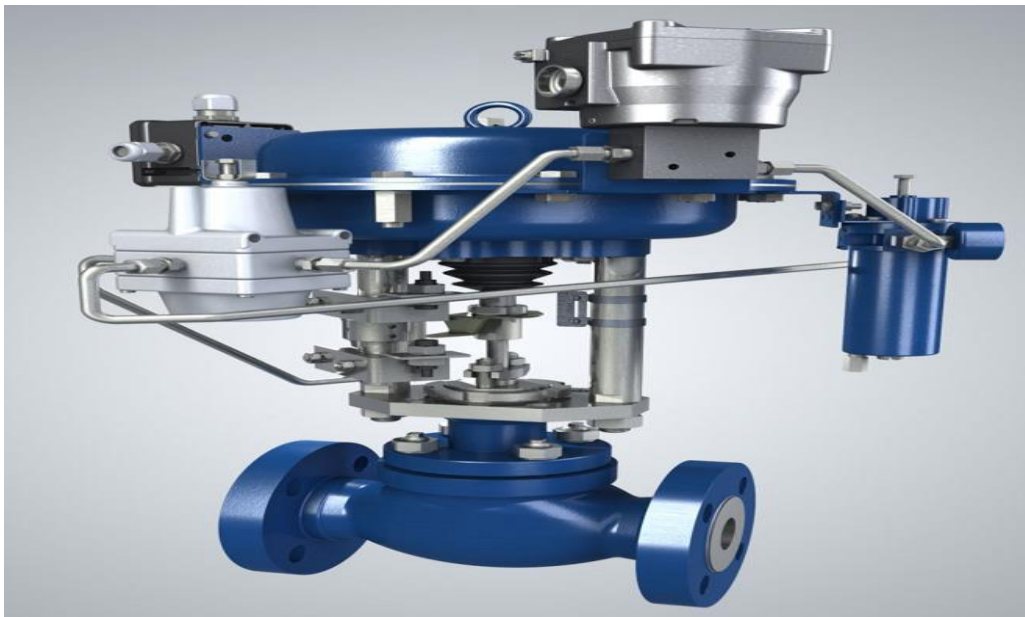


Figure I.8: Vanne de régulation [18]

I.10.5 Vanne Tout ou Rien (Tor)

Les vannes automatiques tout ou rien est des équipements automatisés dont le rôle est d'interrompre ou de permettre le passage d'un fluide (gaz ou liquide). Elle exécute une action discontinue qui prend deux position ou deux état 0 et 1 (ou 0 et 100%), c'est-à-dire ouvert ou fermée.

Les vannes tout ou rien sont utilisées pour commander les systèmes ayant une grande Inertie ou la précision de la régulation n'est pas importante. [19]



Figure I.9: Vanne tout ou rien (TOR). [19]

I.10.6 Purgeur de vapeur

Un purgeur de vapeur est une sorte de vanne autonome qui évacue le condensât (c.-à-d. la vapeur d'eau condensée) et autres gaz non condensables sans laisser échapper de vapeur. Dans le secteur industriel, la vapeur d'eau est souvent utilisée pour chauffer ou comme force motrice. Le purgeur de vapeur est un appareil utilisé pour minimiser les pertes inutiles de vapeur dans ces procédés. [20]

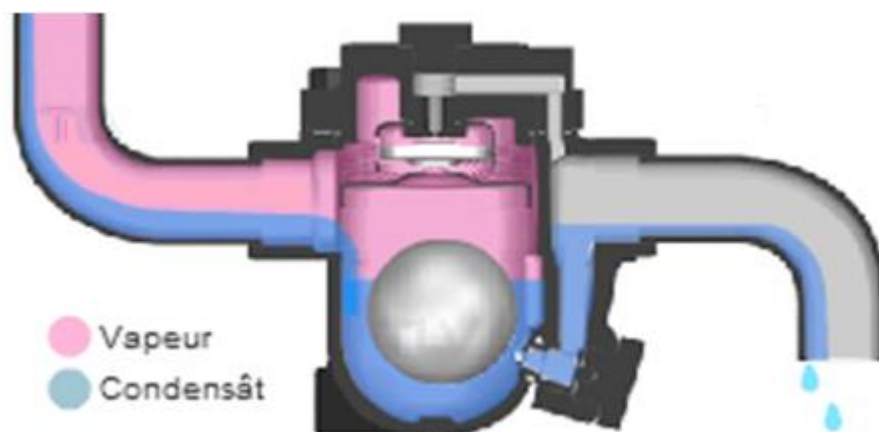


Figure I.10: Purgeur de vapeur. [20]

I.10.6.1 Principe de fonctionnement

Le purgeur a pour fonction de permettre l'évacuation des condensats sans laisser passer la vapeur. Il doit également permettre l'évacuation de l'air et des incondensables présents, au démarrage comme en fonctionnement stabilisé. [21]

I.11 Capteurs**I.11.1 Définition**

Un capteur est un dispositif permettant de détecter, en vue de le quantifier et de le représenter, un phénomène physique sous la forme d'un signal, généralement électrique. Le capteur se différencie du détecteur et du senseur par sa possibilité de délivrer une grandeur physique directement utilisable pour une mesure ou une commande.

I.11.2 Types de capteurs utilisés dans l'installation**I.11.2.1 Capteur de pression**

Un capteur de pression est un instrument composé à la fois d'un élément sensible à la pression pour déterminer la pression réelle appliquée au capteur et de certains composants pour convertir cette information en un signal de sortie. [22]



Figure I.11: Capteur de pression

I.11.2.2 Capteur de température

Un capteur de température est un appareil qui convertit généralement une valeur de température en un signal électrique lisible. Elle est aussi souvent appelée sonde de température ou thermo-capteur. [23]

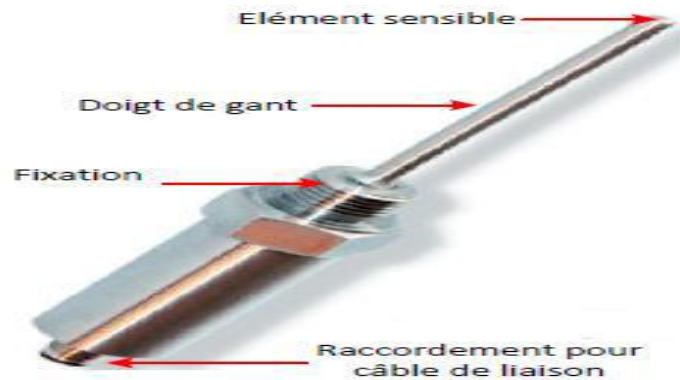


Figure I.12 : Capteur de température

I.12 Conclusion

Dans ce chapitre, Nous avons présenté l'objectif de la régulation, une description générale des systèmes industriels ainsi que quelques méthodes de régulation, ce qui résume dans le rôle des paramètres des régulateurs PID, l'ensemble de ces informations va nous aider à aborder le prochain chapitre qui est l'automate programmable industriel(API).

Dans ce chapitre, on a exposé le principe des chaînes de régulation et on a décrit les principaux régulateurs industriels. On a surtout insisté sur le régulateur PID, qui reste, sous différentes versions, à la base de la majorité des équipements de contrôle-commande de l'industrie.



Chapitre II :
Automate programmable
industriel

Chapitre II : Automate programmable industriel

II.1 Introduction

Les Automates Programmables Industriels (API) sont apparus aux États-Unis vers 1969 où ils répondaient aux désirs des industries de l'automobile américaines (General Motor) de développer des chaînes de fabrication automatisées qui pourraient suivre l'évolution des techniques et des modèles fabriqués. Ils sont apparus en France en 1971, ils sont de plus en plus employés dans l'industrie. Avant ils utilisaient des relais électromagnétiques et de systèmes pneumatiques pour la réalisation des parties commandes (logique câblé). [24]

De 1970 à 1974, la technologie des microprocesseurs (du moins les premiers) ajouta une plus grande flexibilité et une « intelligence » à l'automate programmable. Les capacités d'interface avec l'utilisateur s'améliorent. L'automate programmable peut maintenant exécuter les opérations arithmétiques en plus des opérations logiques; il manipule les données et les adresses ; il effectue la communication avec d'autres automates ou ordinateurs, donnant ainsi une nouvelle dimension aux applications de l'automate programmable.

Les automates programmables sont désormais utilisés dans de nombreux secteurs d'activité, pour la commande des machines et des chaînes de production, la régulation de processus ou encore dans le secteur du bâtiment, pour le contrôle de l'éclairage, du chauffage, de la sécurité....

Parmi les marques d'automates les plus utilisées, nous avons : Siemens, Allen Bradley, Schneider, Omron, Mitsubishi, Proface, Unitronics, Yokogawa, Honeywell, General Electric, etc....quelque exemple montrer au-dessus :

La programmation et l'exploitation sont simples par la structure de langage, souvent de type symbolique et de moyens de programmation accessible à l'électricien de maintenance. [25]



Figure II.1: Automates Mitsubishi



Figure II.2: Automate programmable Allen-Bradley



Figure II.3 : Automate Omron

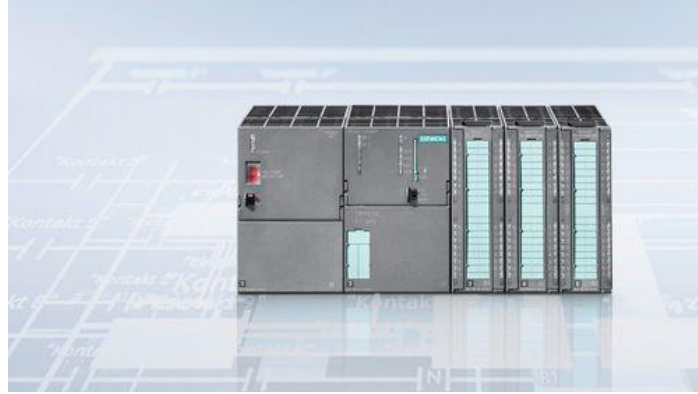


Figure II.4: Automate Siemens

La société SIEMENS produit l'automate SIMATIC S7-300 qui est un automate à extensibilité modulaire, utilisé particulièrement dans l'industrie manufacturière, Possédant une gamme régulière de CPU, qui offre un niveau très élevé de performances, avec une grande vitesse de traitement. [25]

II.2 Systèmes automatisés

II.2.1 Définition

Un système automatisé est un ensemble d'éléments qui effectue des actions sans intervention de l'utilisateur. Celui-ci se contente de donner des ordres de départ et si besoin d'arrêt.

II.2.2 But de l'automatisation

L'automatisation permet d'apporter les éléments supplémentaires à la valeur ajoutée par le système. Les objectifs de l'automatisation sont comme suite : [26]

- Accroître la productivité.
- Augmenter la compétitivité.
- Augmenter la sécurité.
- Économiser les matières premières et l'énergie.
- Superviser les installations et les machines.
- Éliminer les tâches répétitives ou sans intérêt.
- Simplifier le travail des agents.

II.2.3 Structure d'un système automatisé

Un système automatisé est composé de plusieurs éléments qui exécutent un ensemble de tâches programmées sans l'intervention de l'homme : [27]

- **Partie commande** : C'est l'ordinateur, microprocesseur, ou un boîtier de commande, elle reçoit les informations et elle donne des ordres à la partie opérative.
- **Partie opérative** : C'est la machine qui réalise les actions physiques, elle regroupe les pré-actionneurs (distributeurs, contacteurs), les capteurs, et les actionneurs (vérins-moteurs).
- **Partie relation** : C'est la partie qui permet le dialogue entre l'homme et la machine, elle permet de commander les systèmes et visualiser les différents états du système. [28]

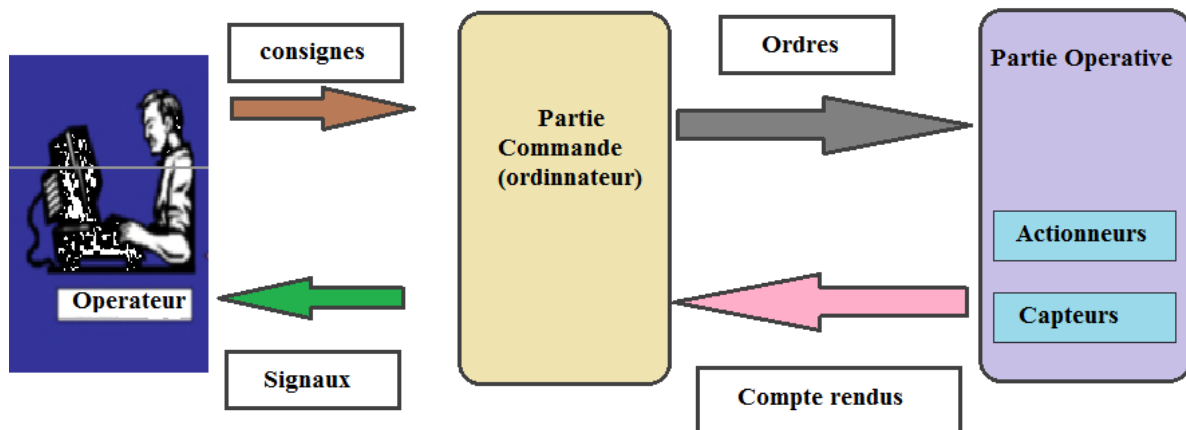


Figure II.5 : Structure de système automatisé

II.3 Automate programmables industriel

II.3.1 Définition

Un automate programmable industriel ou en anglais PLC (Programmable Logic Controller) est un dispositif électronique numérique programmable destiné pour gérer de manière automatique les systèmes de commande des installations électriques industrielles. Il contrôle les actionneurs grâce à un programme informatique qui traite les données d'entrée recueillies par des capteurs. Il existe deux types d'automates qui sont Modulaire et Monobloc (Compact). [25]

II.3.2 Caractéristique de l'automate programmable

Les caractéristiques fondamentales qui distinguent l'Automate Programmable Industriel (API) sont : [24]

- Nombre d'entrée et de sortie (E /S).
- Tensions d'alimentation
- Taille de mémoire
- Langage de programmation.
- Compact ou modulaire.

II.3.3 Domaine d'utilisation de l'automate programmable

On utilise les API dans tous les secteurs industriels pour la commande des machines (convoyage, emballage ...) ou des chaînes de production (automobile, agroalimentaire ...) ou il peut également assurer des fonctions de régulation de processus (métallurgie, chimie ...).

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. [29]

II.3.4 Nature des informations traitées par l'automate

Les informations traitées par un API peuvent être de type :

- **Tout ou Rien** : L'information peut prendre que vrai ou faux (1 ou 0), ce type délivré par un détecteur, bouton poussoir ... [30]
- **Analogique** : l'information peut prendre toutes les valeurs possibles dans un intervalle donné. [31]
- **Numérique** : L'information traitée dans ce cas donne des mots chiffrés en binaire ou bien hexadécimale. [31]

II.3.5 Description de l'automate programmable :

II.3.5.1 Aspect extérieur :

Les automates peuvent être soit de famille compact ou modulaire :

- **Types compact** : c'est les micro-automates, il intègre l'alimentation, le processeur, les entrées /sorties selon les modèles et les fabricants, il pourra réaliser certaines fonctions supplémentaires (comptage rapide, E/S analogique ...etc.). Ces automates sont généralement destinés à la commande de petits automatismes. [25]

- Type modulaire** : Automate modulaire est constitué de modules séparés pour : l'alimentation, le processeur, les entrées /sorties, les interfaces de communication. Les modules sont branchés sur un ou plusieurs racks, ces automates sont utilisés dans les automatismes complexes. [25]



Figure II.6 : Automate compact ABB



Figure II.7 : Automate modulaire Mitsubishi

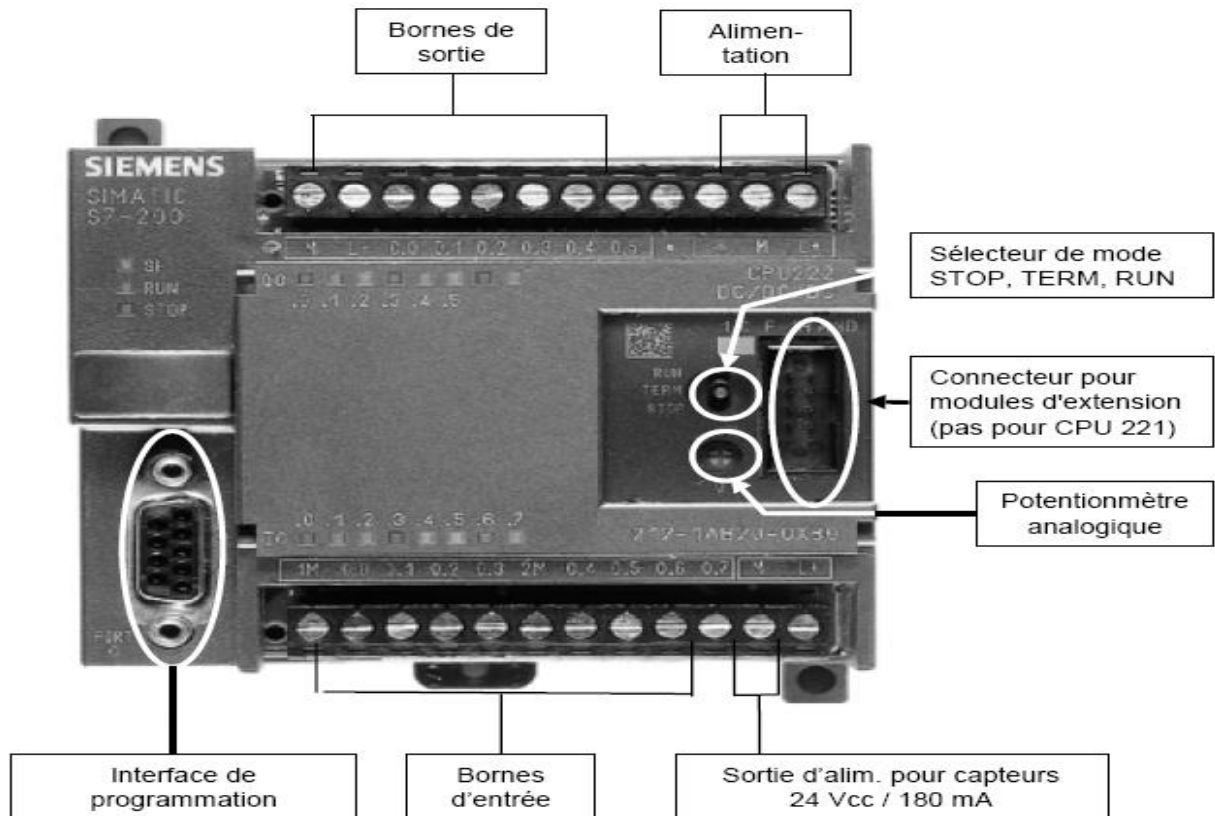


Figure II.8 : Aspect extérieur

II.3.5.2 structure interne

- **Module d'alimentation**: Il assure la distribution d'énergie aux différents modules. [32]
 - **Unité centrale CPU (Centrale Processing Unit)**: Elle représente le regroupement du processeur et de la mémoire centrale (dit le cerveau de l'automate).
 - **Processeur** : réalise toutes les fonctions logiques, arithmétiques et de traitement numérique (transfert, comptage, temporisation). [32]
 - **Mémoire** : gère et stocke les informations issues des différents secteurs du système qui sont le terminal de programmation (PC ou console) et le processeur, qui lui gère et exécute le programme. Elle reçoit également des informations en provenance des capteurs et différents éléments du système. Elle est composée de : [33]
 - ✓ La mémoire RAM (Random Access Memory)
 - ✓ La mémoire ROM (Random Only Memory)
 - ✓ La mémoire EPROM (Erasable Programmable Read Only Memory)
 - ✓ La mémoire EEPROM (Electrically-Erasable Programmable Read Only Memory).
- [28]

- **Le bus interne:** Il permet la communication de l'ensemble des blocs de l'automate et des éventuelles extensions.
- **Modules d'entrées:** Permettent à l'automate d'acquérir les informations provenant soit du pupitre de commande, ou bien des capteurs et détecteurs. Elles reçoivent les informations
- **Modules de sorties:** Elles commandent les divers pré-actionneurs et éléments de signalisation.

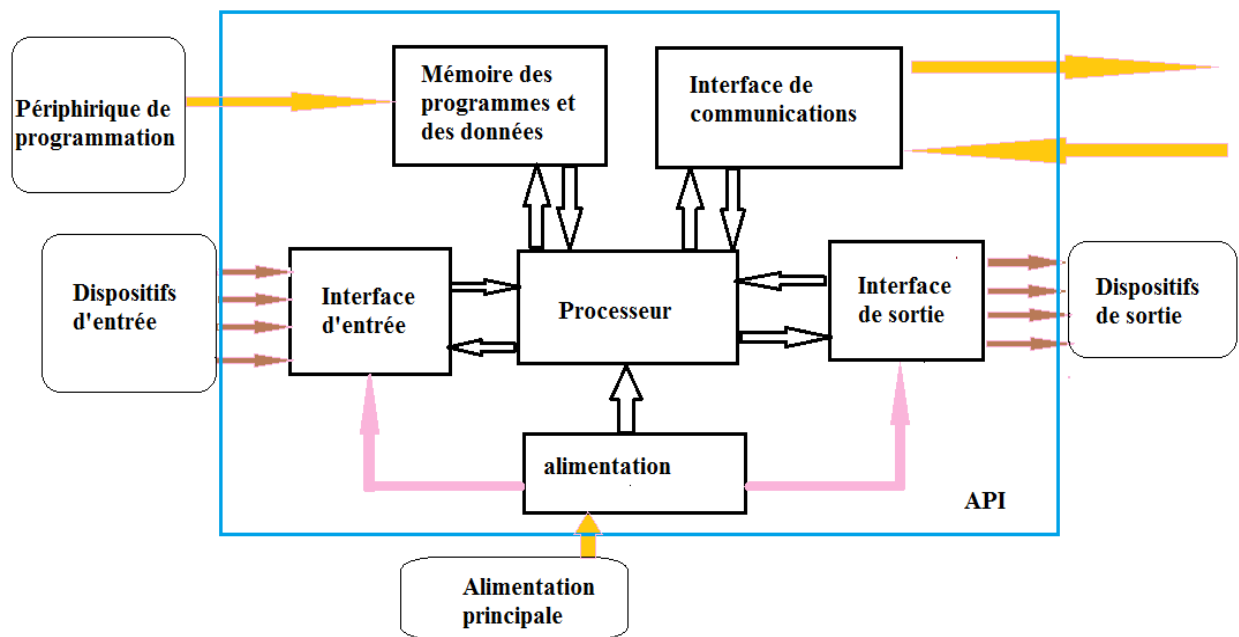


Figure II.9: Structure interne de l'API

II.3.6 Langage de programmation d'un API

Il existe 5 langages de programmation des automates qui sont normalisés au plan mondial qui sont :

- **Langage à contacts (LD : Ladder diagram):** Langage graphique développé pour les électriciens. Il utilise les symboles USA tels que : contacts, relais et blocs fonctionnels et s'organise en réseaux (labels). C'est le plus utilisé. Mais il n'est pas adaptable pour les réseaux complexes, ni aux calculs complexes. [34] C'est le langage qu'on va utiliser dans notre travail.

Nom	Élément graphique	Fonction
Contact à ouverture		Contact passant lorsque l'objet bit de contrôle se trouve à l'état 1.
Contact à fermeture		Contact passant lorsque l'objet bit de contrôle se trouve à l'état 0.
Contact de détection d'un front montant		Front montant : détecte le passage de 0 à 1 de l'objet bit de contrôle.
Contact de détection d'un front descendant		Front descendant : détecte le passage de 1 à 0 de l'objet bit de contrôle.

Nom	Élément graphique	Fonction
Bobine directe		L'objet bit associé prend la valeur du résultat de la zone de test.
Bobine inverse		L'objet bit associé prend la valeur du résultat inverse de la zone de test.
Bobine d'enclenchement		L'objet bit associé est réglé sur 1 lorsque le résultat de la zone de test est 1.
Bobine de déclenchement		L'objet bit associé est réglé sur 0 lorsque le résultat de la zone de test est 1.

Nom	Élément graphique	Fonction
Connexion horizontale		Relie en série les éléments graphiques de test et d'action entre les deux barres verticales.
Connexion verticale		Relie les éléments graphiques de test et d'action en parallèle.

Figure II.10: langage à contacte (Ladder)

- **Liste d'instructions ou mnémonique (IL : Instruction List):** Langage textuel de même nature que l'assembleur (programmation des microcontrôleurs). Très peu utilisé par les automaticiens.
- **Langage littéral structuré (ST : Structured Text) :** Langage informatique de même nature que le Pascal, il utilise les fonctions comme if ... then ... else ... (si ... alors ... sinon ...) Peu utilisé par les automaticiens.
- **Blocs Fonctionnels (FBD : Function Bloc Diagram) :** Langage graphique où des fonctions sont représentées par des rectangles avec les entrées à gauche et les sorties à droites. Les blocs sont programmés (bibliothèque) ou programmables. Utilisé par les automaticiens.
- **Le langage GRAFCET (SFC : Sequential Function Chart) :** C'est un langage graphique qui permet de tracer directement le schéma Grafcet de l'automatisme considéré. Dans ce cas, pour plus de facilité, on construit un Grafcet niveau 3 qui est le même que le niveau 2, mais les variables d'E/S du niveau 2 sont remplacés par les adresses de l'automate (appelés références). [35]

II.3.7 Traitement du programme automate

Généralement les automates programmables industriels ont un fonctionnement cyclique : [36]

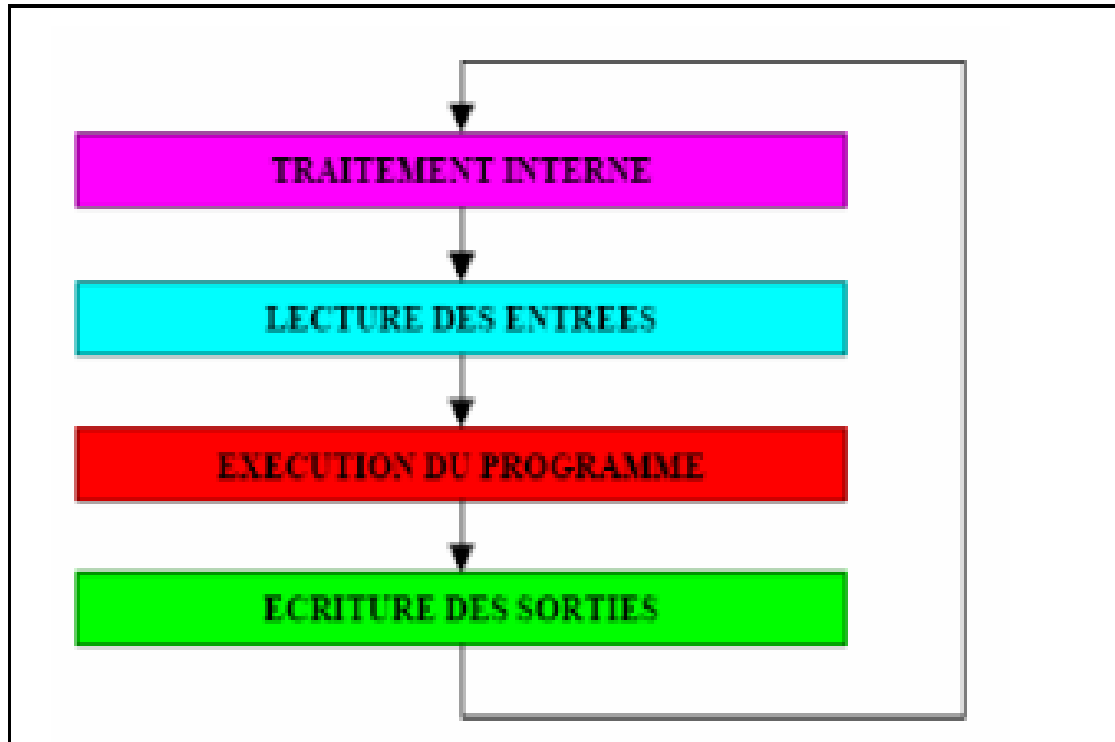


Figure II.11: Cycle d'exécution d'un programme

- **Traitement interne:** L'automate effectue des opérations de contrôle 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 recopie 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.
- **Écriture des sorties :** L'automate bascule les différentes sorties (de façon synchrone) aux positions définies dans la mémoire image des sorties.
- **Temps de réponse :** C'est le temps qui s'écoule entre le changement d'état d'une entrée et le changement d'état de la sortie correspondante : [28]

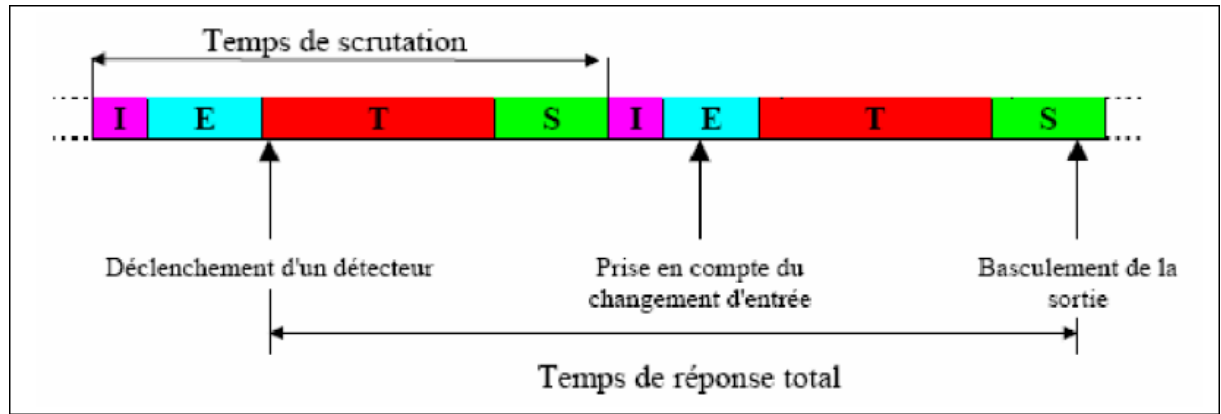


Figure II.12 : Temps de réponse totale (TRT)

Le temps de réponse total est au plus égal à deux fois le temps de scrutation (sans traitement particulier). Le temps de scrutation est directement lié au programme implanté.

Ce temps peut être fixé à une valeur précise (fonctionnement périodique), le système indiquera alors tout dépassement de période. [28]

II.3.7 Critère de choix d'un automate programmable

Pour bien choisir un automate programmable, On doit tenir compte de plusieurs critères : [30]

En première lieu le choix d'une société ou d'un groupe, Le matériel, le logiciel et les services annexes ainsi que

- **Nombres d'entrées/sorties intégrés** : Il faut toujours évaluer combien d'entrées et de sorties (E/S) numériques et analogiques nous aurons besoin. Ci-dessous un A.P.I. le CROUZET (8e/4s) qui est compact. [37]
- **Type de processeur**: La taille mémoire, la vitesse de traitement et les fonctions spéciales offertes par le processeur permettront le choix dans la gamme souvent très étendue.
- **La marque** : Tenir compte de la marque est un critère à prendre en compte lorsqu'on est déjà familiarisé avec un API d'une certaine marque. En effet, un automate d'une autre marque sera peut-être plus long à maîtriser car quelque peu différent. [39]



Figure II.13 : Automate CROUZET Millénium II [27]

- **Le prix :** Même si l'on s'accorde à dire qu'un automate industriel constitue un bon investissement à long terme, il convient de garder en tête un prix juste et raisonnable.
- **Fonctions de communication:** L'automate doit pouvoir communiquer avec les autres systèmes de commande (API, supervision ...) et offrir des possibilités de communication avec des standards normalisés (Profibus ...). [25]
- **Fonctions ou modules spéciaux :** Certaines cartes (commande d'axe, pesage...) permettront de soulager le processeur de devront offrir les caractéristiques souhaitées.
- **Simplicité et facilite de l'utilisation :** Il faut choisir un automate qui est simple et facile à l'utilisation, exemple l'automate Rockwell qui utilise pour les systèmes de commande simple.

Pour notre cas l'entreprise CEVITAL a choisie d'utilisée l'automate siemens S7-300.

II.4 Présentation de l'automate S7-300

Le SIMATIC S7-300 est un automate à extensibilité modulaire, utilisé particulièrement dans l'industrie manufacturière, possédant une gamme régulière de CPU, qui offre un niveau très élevé de performances, avec une grande vitesse de traitement, et les Modules entrée /sorties (TOR, Analogique), L'automate est programmable à l'aide d'un PC avec le logiciel STEP 7 ou TIA Portal.

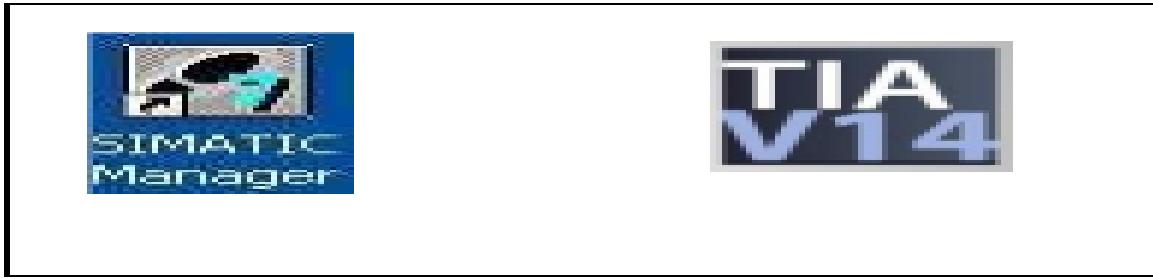


Figure II.14 : Logiciel de programmation SIMATIC S7-300

Le S7-300 est l'automate conçu pour des solutions dédiées au système manufacturier et constitue à ce titre une plate-forme d'automatisation universelle pour les applications avec des architectures centralisées et décentralisées. (Figure II-14) [40]



Figure II.15 : Architecture réelle Automate SIMATIC S7-300

II.4.1 Description de l'automate S7-300

Le SIMATIC S7-300 est utilisé dans de nombreuses applications dans le monde entier et a fait ses preuves des millions de fois. Les contrôleurs universels SIMATIC S7-300 permettent d'économiser de l'espace d'installation et se caractérisent par une conception modulaire. Une large gamme de modules permet d'étendre le système de manière centralisée ou de créer des structures décentralisées en fonction de la tâche à accomplir, et facilite un

stockage économique des pièces de rechange. SIMATIC est connu pour sa continuité et sa qualité. [41]

Les SIMATIC S7-300 et S7-400 sont et restent des automates entièrement développés et approuvés. De nombreuses nouvelles exigences en matière de manipulation et de performance ne peuvent plus être entièrement satisfaites avec l'architecture traditionnelle.

C'est pourquoi la nouvelle génération de contrôleurs S7-1500 a été développée avec des propriétés système améliorées. Les propriétés de base de SIMATIC ont été conservées pour des raisons de compatibilité. [42]

Dans notre système on a utilisé les modules suivants :

- + Alimentation : PS 307 10A
- + CPU (Unité central) : CPU 312
- + Modules TOR : Entrées : AO4×0/4...20mA. Ex
- + Sortie : DO16×AC120V/0.5A
- + Module Analogique : AI2×12bit

II.4.2 Structure de l'automate S7-300

Parmi les caractéristiques essentielles du S7-300, le fait qu'il dispose d'une vaste gamme de Modules suivant: [43]

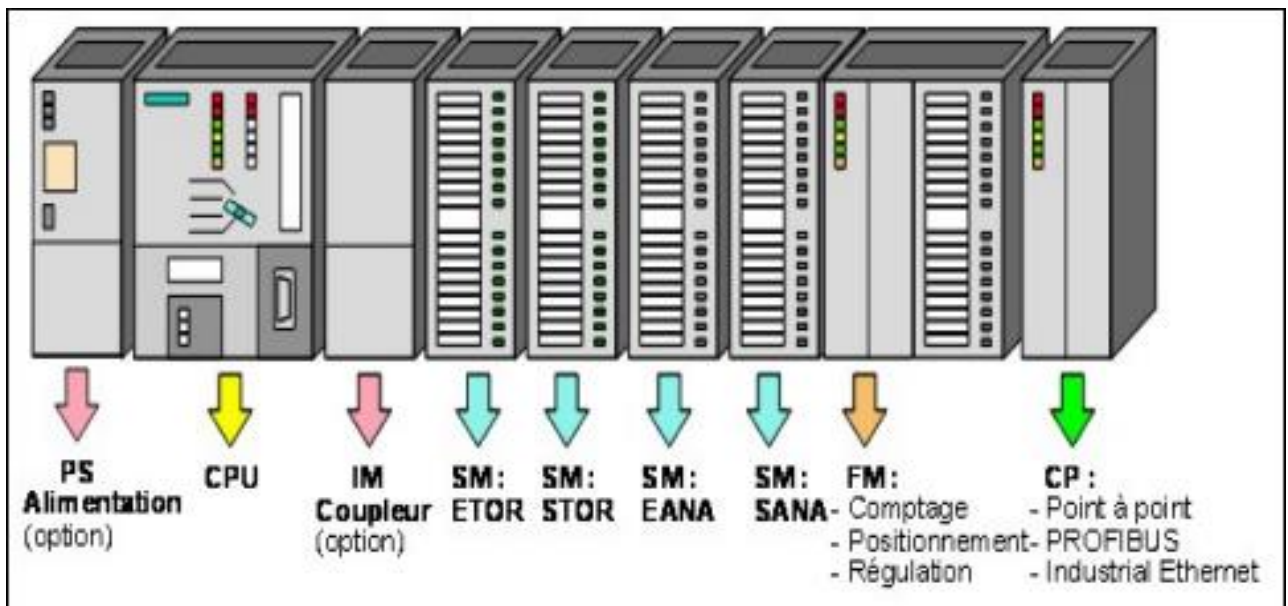


Figure II.16 : Disposition des modules de l'API S7-300

L'automate est équipé des modules ci-dessus : [43]

- ✓ Emplacement 1 : alimentation.
- ✓ Emplacement 2 : CPU.
- ✓ Emplacement 3 : module de complémentaire.
- ✓ Emplacement 4 : entrée TOR (digital input).
- ✓ Emplacement 5 : sorties TOR (digital output).
- ✓ Emplacement 6 : entrée analogique (analog input).
- ✓ Emplacement 7 : sorties analogique (analog output).
- ✓ Emplacement 8 : module de comptage.
- ✓ Emplacement 9 : module de communication.

II.4.2.1 Modules d'alimentation PS :

Le module d'alimentation utilisé dans ce projet est : PS 307, 10A, assigné se 10A, sa tension nominale de sortie 24 Vcc, stabilisée, tenue aux courts-circuits et à la marche à vide

II.4.2.2 Présentation de la CPU (S7-300)

On a le choix entre plusieurs CPU, avec différentes performances, la CPU 312 représente des grandes performances [33]

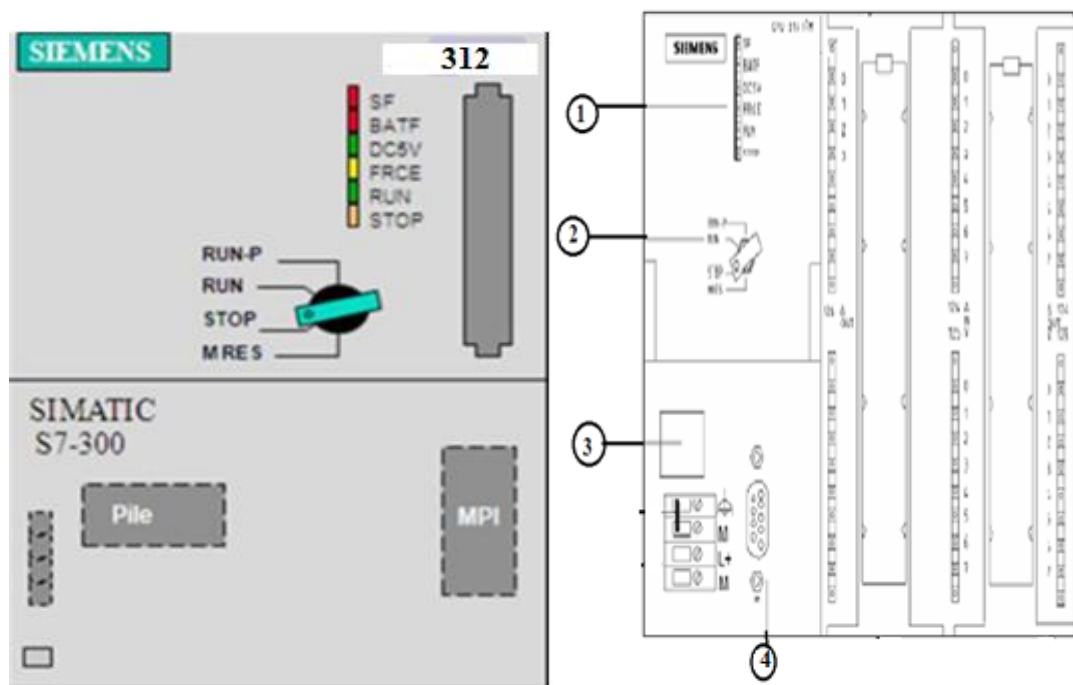


Figure II.17 : Vue de CPU S7 -300(312). [43]

1) LED de visualisation d'état et de défaut : Allumage des LED signifier ce qui suit :

(Rouge) SF : indique le défaut matériel ou logiciel

(Rouge) BATF : indique la défaillance de la pile.

(Verte) 5V DC : indique l'alimentation 5V CD est correcte

(Jaune) FRCE : indique le forçage permanent est actif.

(Verte) RUN : indique la CPU en RUN.

(Jaune) STOP : indique la CPU en STOP ou en ATTENTE ou en démarrage. [44]

2) Commutateur de mode de fonctionnement

Tableau II-1 : Position du commutateur du mode de fonctionnement. [44]

Position	Signalisation	Explication
RUN-P	Mode de fonctionnement RUN-PROGRAMME	La CPU traite le programme utilisateur le programme peut être modifié. Dans cette position la clef ne peut être retirée.
RUN	Mode de fonctionnement RUN	La CPU traite le programme utilisateur. Le programme ne peut être modifié qu'avec légitimation par mot de passe. La clef peut être retirée
STOP	Mode de fonctionnement STOP	La CPU ne traite aucun programme utilisateur. La clef peut être retirée
MRES	Effacement général	Position instable du commutateur, pour effectuer l'effacement général il faut respecter un ordre particulier de commutation.

3) Logement pour pile de sauvegarde ou accumulateur

L'utilisation de l'accumulateur ou de la pile de sauvegarde est nécessaire pour l'horloge temps réelle. La pile de sauvegarde est aussi utilisée pour :

- ✚ La sauvegarde du programme utilisateur s'il n'est pas enregistré dans la mémoire morte.
- ✚ Pour étendre la zone rémanente de données ; L'accumulateur est rechargé à chaque mise sous tension de la CPU. Son autonomie est de quelques jours voire quelques semaines

au maximum. La pile de sauvegarde n'est pas rechargeable mais son autonomie peut aller jusqu'à une année. [44]

6) Interface multiple MPI des CPU

L'interface MPI (Multi Point Interface ou interface multipoint) est l'interface nécessaire de la CPU utilisée pour la console de programmation (PG ou PC), le pupitre opérateur (OP) ou par la communication au sein d'un réseau MPI. La vitesse de transmission typique est de 187.5K Bauds. Le contrôle-commande avec HMI et l'échange de données entre des CPU SIMATIC S7.

II.4.2.3 Module de couplage (IM):

Ce sont des cartes électroniques utilisées pour assurer la communication entre l'unité centrale et les périphériques de l'automate (entrées/sorties), console de programmation, etc...). Les coupleurs (IM360, IM361, ou IM365) permettent de réaliser la configuration à plusieurs châssis. Ils occupent module complémentaire dans l'API et ce dernier reste vide au cas on n'utilise pas les coupleurs (IM). [45]

II.4.2.4 Module de signaux d'entrées/ sortie SM

Un ou plusieurs modules utilise les E/S TOR ou analogiques et qui est divisé :

- **Module d'entrée**

Permettent à l'automate de recevoir des informations prévenantes soit de la part des capteurs (entrée logique, analogique ou numérique) ou du pupitre de commande.

- **Modules de sortie**

Permettent de raccorder l'automate avec les différents pré-actionneurs (contacteur, relais...etc.) ainsi qu'avec les actionneurs (vanne régulatrice, pompes). [44]

II.4.2.5 Module de simulation (SM) :

C'est un module spécial qui offre à l'utilisateur la possibilité de tester son programme lors de la mise en service et en cours de fonctionnement. Dans le S7-300, ce module fonctionne telles que :

- ✚ La simulation des signaux de capteurs aux moyens d'interrupteurs.
- ✚ La signalisation d'état des signaux de sortie par des LEDs. [44]

II.4.2.6 Module de fonction (FM)

Ces modules réduisent la charge de traitement de la CPU et offrent les fonctions suivantes :

- Comptage.
- Régulation.
- Positionnement [46].

II.4.2.7: Modules de communication CP

Ils permettent d'établir des liaisons Homme-Machine que sont effectuées par les interfaces de communication suivantes :

- Point à point ;
- Profibus.
- Industriel Ethernet.

II.5 Conclusion

L'automatisation joue un rôle très important dans le domaine industriel, elle peut améliorer la productivité et l'efficacité.

Dans ce chapitre nous avons étudié le système automatisé et des généralités sur les automates programmables ainsi que son architecture interne. Par suite on a focalisé notre étude sur l'automate S7-300 en mettant en avant ses caractéristiques techniques pour une meilleure exploitation pendant sa programmation qui sera l'objet du chapitre III.



Chapitre III :
Programmation et supervision

Chapitre III : Programmation et supervision

III.1 Introduction

Afin de piloter la boucle de chauffage, nous avons réalisé un programme et l'implanter dans l'automate S7-300 grâce au logiciel de conception des programmes pour les systèmes d'automatisation SIMATIC STEP7.

Ce chapitre est consacré à présenter le logiciel Step7 et a expliqué les différentes étapes qu'on a suivi pour réaliser le programme de notre projet sur le STEP 7 et la supervision sur le Win CC flexible.

III.2 Description du logiciel STEP7

STEP7 est le logiciel de base pour la configuration et la programmation des systèmes d'automatisation SIMATIC S300 et S400. Il fait partie du logiciel industriel SIMATIC. Le logiciel de base assiste dans les phases de processus de création de la solution d'automatisation : la conception de l'interface utilisateur du logiciel STEP7 répond aux connaissances ergonomiques modernes. [47]



Figure III.1: Interface du logiciel STEP7.

III.3 Structure du programme en STEP7

La programmation structurée permet la rédaction claire et transparente de programmes. Elle permet la construction d'un programme complet à l'aide de modules qui peuvent être échangés et/ou modifiés à volonté. Pour permettre une programmation structurée confortable, il faut prévoir plusieurs types de modules : les modules d'organisation (OB), de programmes (FB), fonctionnels (FC), de pas de séquences (SB), de données (DB). [48]

III.3.1 Bloc d'organisation (OB):

Ils sont utilisés pour l'organisation interne du programme et forment ainsi un moyen puissant et essentiel pour la programmation structurée. Ils servent par exemple au déroulement cyclique du programme principal, à l'exécution de programmes d'interruption par des fonctions d'alarmes ou de temps, ou par des fonctions diagnostics internes autant du point de vue hardware que software du système complet.

On clique sur le répertoire « blocs », puis avec un clic droit sur cette fenêtre « Insérer un nouvel objet, Bloc d'organisation » qui est un bloc d'organisation pour le programme, dans lequel on fait appel aux différentes fonctions utilisées dans notre projet.

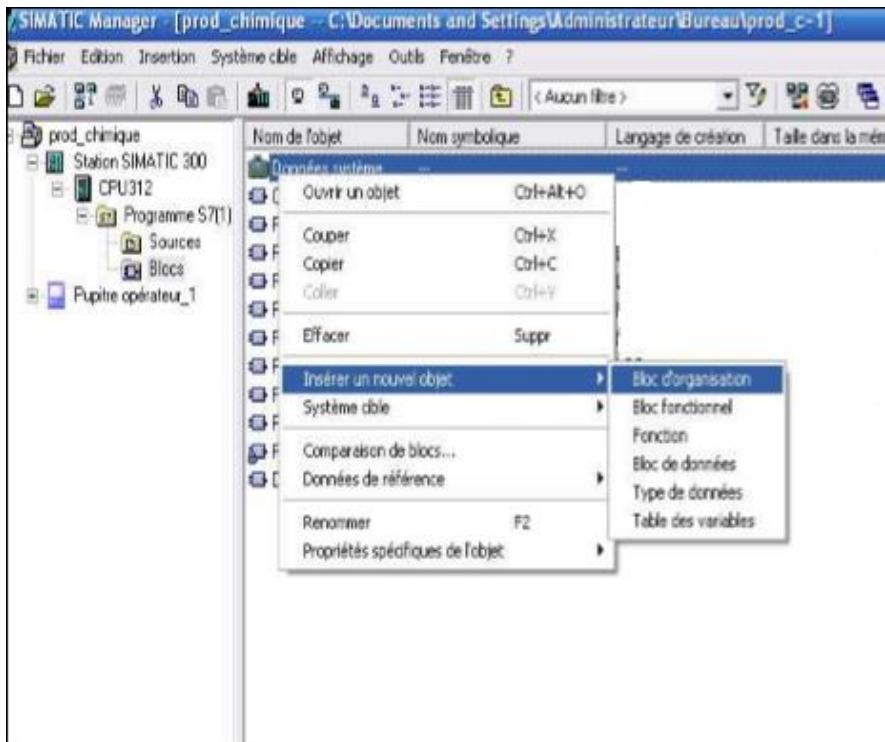


Figure III.2: Fenêtre de Création d'un bloc d'Organisation

Les blocs d'organisation définissent l'ordre (événement de déclenchement) dans lequel les différentes parties du programme sont traitées. L'exécution d'un OB peut être interrompue par l'appel d'un autre OB. Cette interruption se fait selon la priorité : les OB de priorité plus élevée interrompent les OB de priorité plus faible.

III.3.2 Blocs d'organisation de traitement cyclique (OB1)

Le bloc d'organisation OB1 sert à l'exécution du programme utilisateur. On programme dans l'OB1, des appels correspondant aux blocs fonctionnement FB, aux fonctions FC ou d'autres types de structure. Ces blocs s'appellent entre eux avec l'OB1 (FB peut appeler FC et FC ne peut pas appeler FB, par contre OB1 peut les appeler tous les deux). L'OB1 ne peut être appelé que par le programme système dès que l'exécution du programme de mise en route est achevée.

III.3.3 Bloc fonctionnel (FB)

Les blocs fonctionnels sont subordonnés aux blocs d'organisation. Ils renferment une partie du programme qui peut être appelée dans l'OB1 ou dans un autre bloc fonctionnel FB. Avant de commencer la programmation du bloc fonctionnel, il est indispensable de remplir la table de déclaration des variables d'entrées/sorties dans chaque bloc fonctionnel, en utilisant des noms qui ne figurent pas dans la table des mnémoniques, ainsi que les paramètres formels et les données statiques.

III.3.4 Fonction (FC) système

Fonction FC est une fonction sans mémoire contient un programme qui est exécuté quand cette fonction est appelée par un autre bloc, on fait appel à la fonction pour :

-Envoyer une valeur de la fonction au bloc appelant.

-Exécuter une fonction.

On clique sur le répertoire « blocs », puis avec un clic à droite sur cette fenêtre « Insérer un nouvel objet, Fonction ».

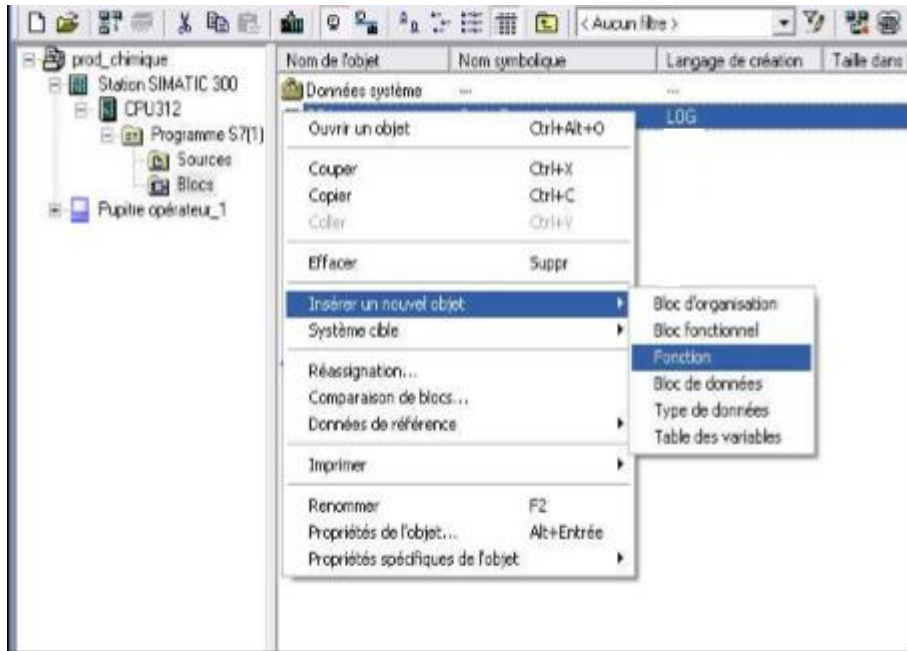


Figure III.3: Fenêtre de Création d'une fonction

III.3.5 Blocs de données (DB)

Dans les blocs sont mémorisées les données nécessaires au traitement du programme et les données affectées à chaque bloc fonctionnel. On clique sur le répertoire « bloc », puis avec un clic droit sur cette fenêtre, on choisit « Insérer un nouvel objet, Type de données » comme illustré dans la Figure III.4 ci-dessous

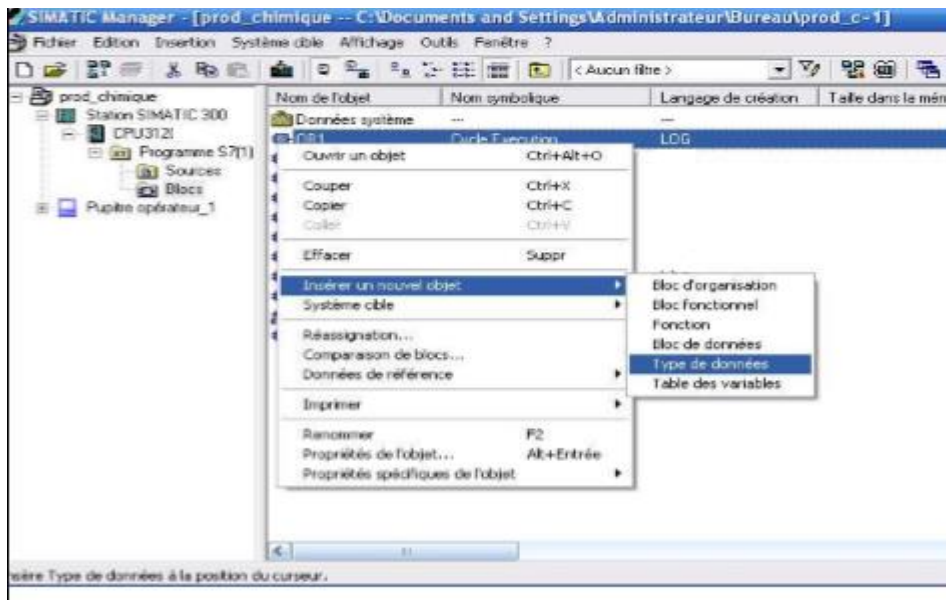


Figure III.4: Fenêtre de Création d'un bloc de donnée

On distingue deux types de blocs de données :

III.3.5.1 Blocs de données d'instance (DB d'instance)

Associés aux FB et aux SFB. Les variables déclarées dans FB déterminent la structure du bloc de donnée d'instance.

III.3.5.2 Blocs de données globaux (DB)

Contenant les données utilisateur communes à tous les blocs.

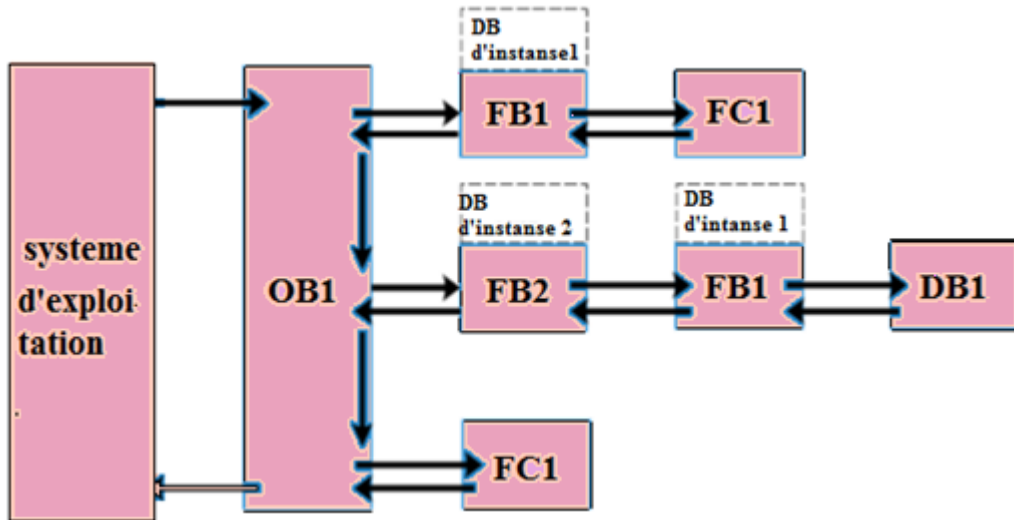


Figure III.5: Structure d'un programme

III.4 Réalisation du programme

III.4.1 Création de projet sous STEP7

Pour commence á créer le projet sur STEP7, On lance le logiciel STEP7 avec l'icône SIMATIC Manager sur l'écran de l'ordinateur.



Une fois le logiciel SIMATIC Manager ouvert, on clique sur l'item fichier puis assistant nouveau projet. Après la sélection du type de la CPU (pour notre projet, on a choisi une CPU312) et l'insertion du bloc d'organisation et le langage à Contacte, une fenêtre s'ouvre pour donner un nom au projet. Pour notre cas (cevitalll) et on clique sur créer.

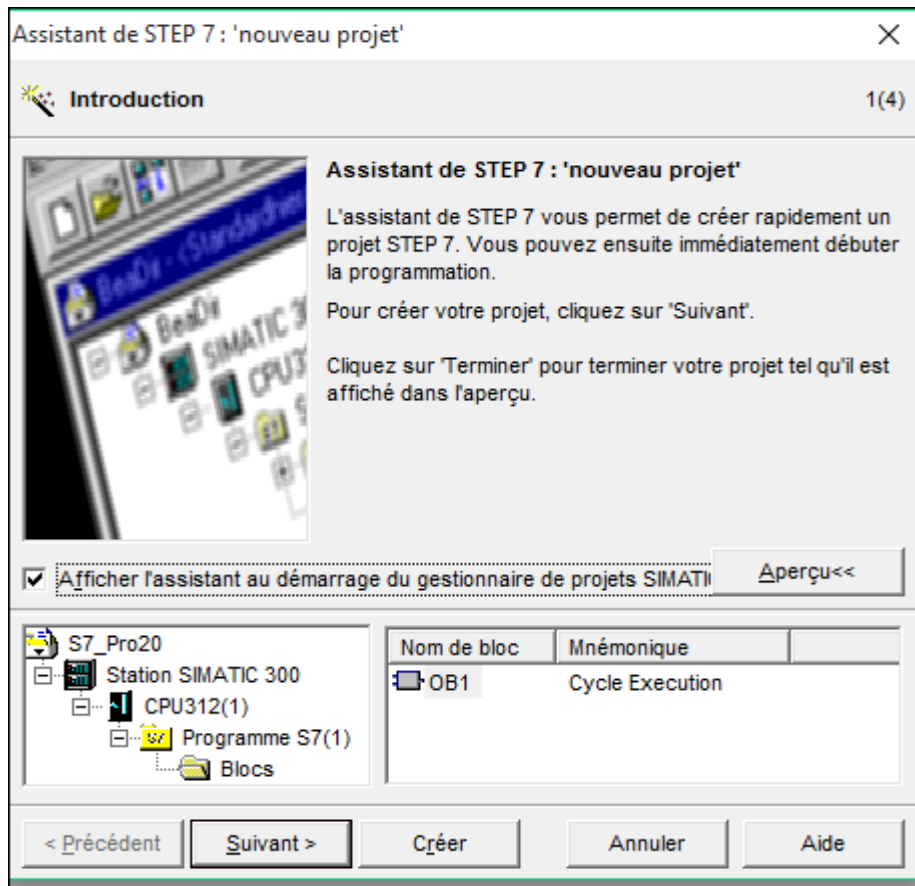


Figure III.6: Création d'un projet

III.4.2 Configuration matérielle

Dans une table de configuration, vous définissez les modules que vous allez mettre en œuvre dans votre solution d'automatisation ainsi que les adresses permettant d'y accéder depuis le programme utilisateur. Vous pouvez en outre y paramétrer les caractéristiques des modules (voir aussi Manipulations de base pour la configuration matérielle).

III.5.2.1 Étape 1: sélection du Rack

Pour choisir le Rack en tenant compte des dimensions de l'armoire électronique de nombre des modules entrée / sortie utilisés dans le projet d'automatique.

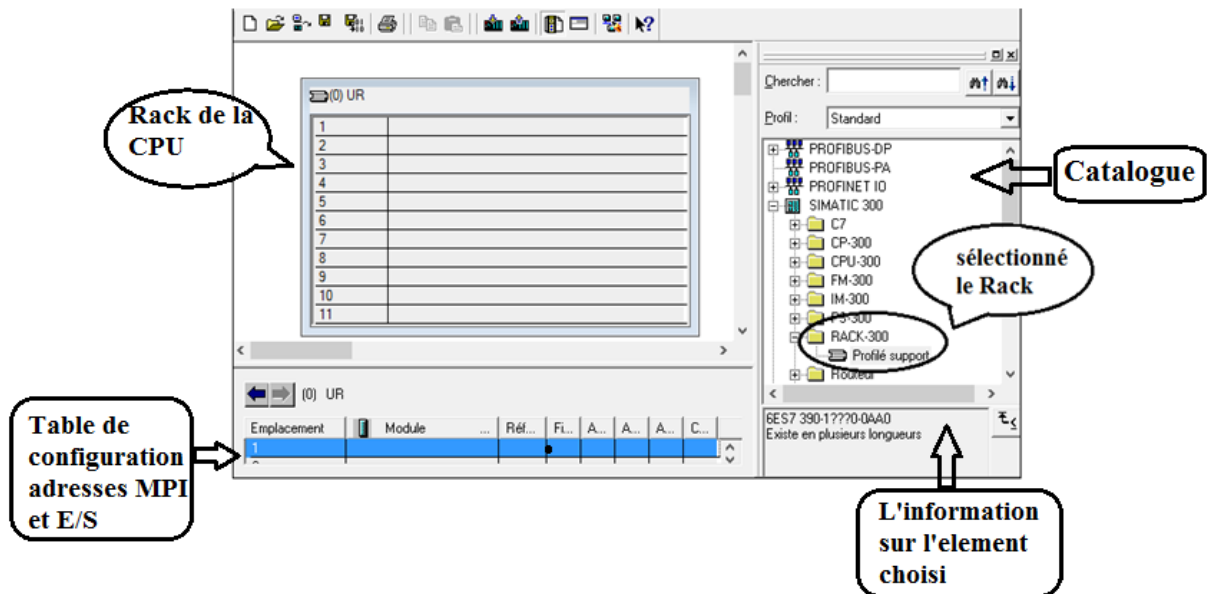


Figure III.7 : sélection du Rack

III.5.2.2 Étape 2: Sélection de la CPU

Après on va choisir la CPU (312 dans notre cas), et choisir l'interface MPI (CPU avec réseau PROFIBUS_DP).

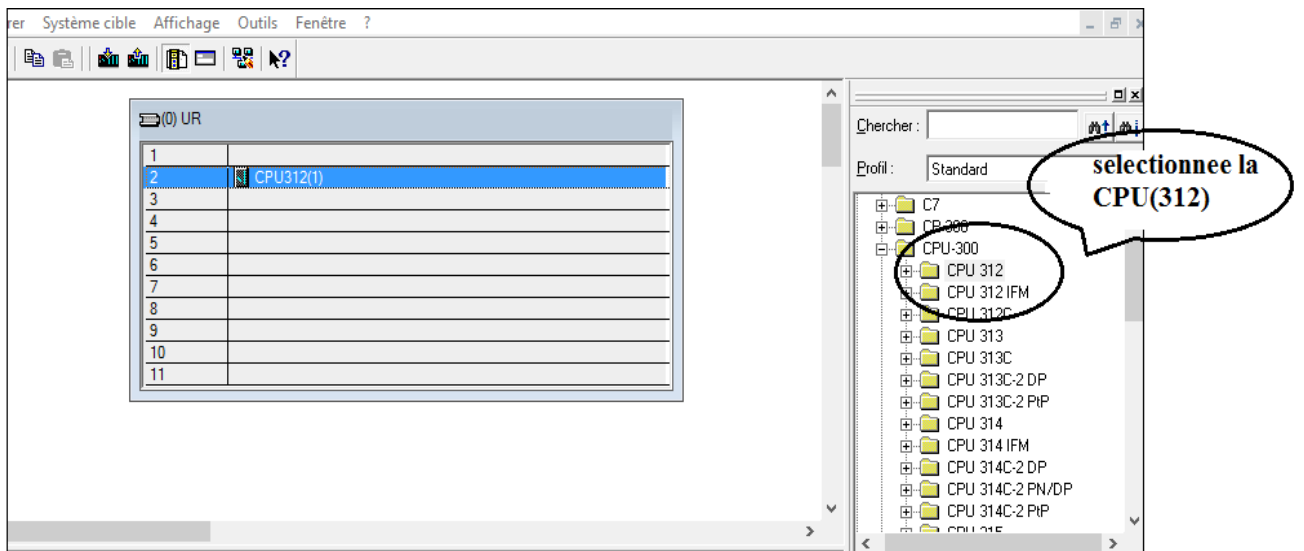


Figure III.8 : Choix de la CPU

III.5.2.3 Étape 3: sélectionné les entrée /sortie

On va choisir les modules d'entrées et sorties analogiques selon le nombre qu'on possède

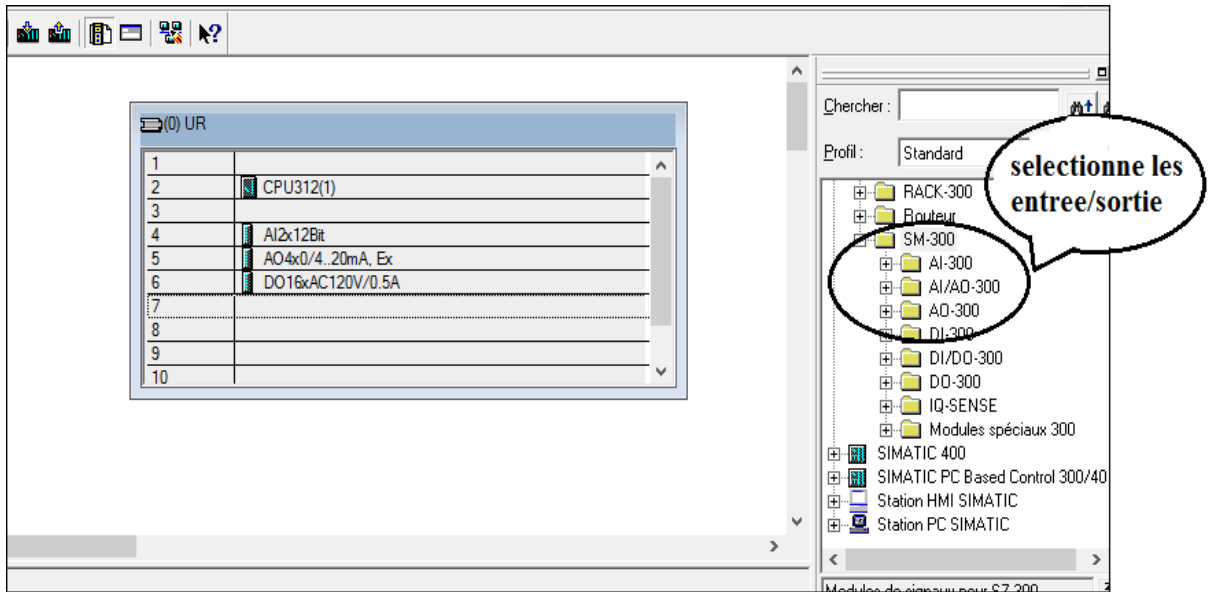


Figure III.9 : Choix des modules d’entrées et sorties

III.5.2.4 Étape 4: Sélection de l’alimentation

Dans nos cas on va choisir l’alimentation (PS307 10A)

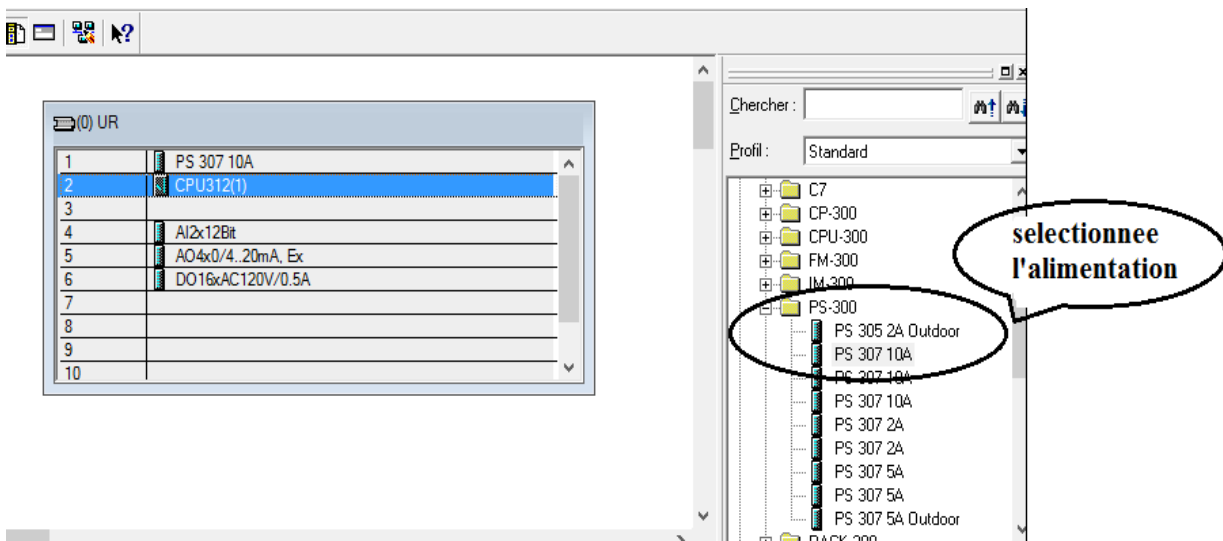


Figure III.10: Choix d’alimentation

Après avoir configurée le matériel, l’enregistré et le compilé, on passera à la programmation de processus de marche de notre système.

Cliquer sur programme Simatic 300 qui nous donne {source, bloc}.

III.5 Création de la table des mnémoniques

Pour améliorer la lisibilité et la clarté de notre programme, nous avons utilisé des mnémoniques à la place des adresses absolues. Pour cela nous avons créé une table de mnémoniques dans laquelle nous avons défini pour chaque opérande utilisée un nom d'adresse absolue, le type de données ainsi qu'un commentaire. Les mnémoniques ainsi définies pourront être utilisées dans l'ensemble du programme.

	Etat	Mnémonique	Opérande	Type de données	Commentaire
1		affichage tt	MD 24	REAL	
2		consigne	MD 4	REAL	
3		CONT_C	FB 41	FB 41	Continuous Control
4		CYC_INT5	OB 35	OB 35	Cyclic Interrupt 5
5		DB temperature	DB 2	FB 41	
6		dssel	M 0.5	BOOL	
7		Entree transmetteur	PEW 256	INT	
8		gain	MD 16	REAL	
9		isel	M 0.4	BOOL	
10		manuel	MD 12	REAL	
11		manuel quit	M 0.1	BOOL	
12		pompe de regulation	A 8.0	BOOL	
13		psel	M 0.3	BOOL	
14		pv in	MD 8	REAL	
15		SCALE	FC 105	FC 105	Scaling Values
16		sortie	MD 20	REAL	
17		sortie vanne	PAW 272	INT	
18		td	MD 40	TIME	
19		tddd	MD 44	TIME	
20		ti	MD 32	TIME	
21		tims	MD 36	TIME	
22					

Figure III.11 : Table mnémonique

III.6 Simulation

Le fonctionnement de du programme STEP7 visualisé via PLCSIM :

III.6.1 Chargement du programme

On procède de deux manières suivantes pour charger le programme dans la CPU éditée:

- 1) Pour charger le classeur des blocs dans la CPU de simulation, en appuyant sur « système cible » puis cliquer sur bouton « charger ».
- 2) Ou directement cliquer sur l'item charger comme le montre la figure ci dessus :

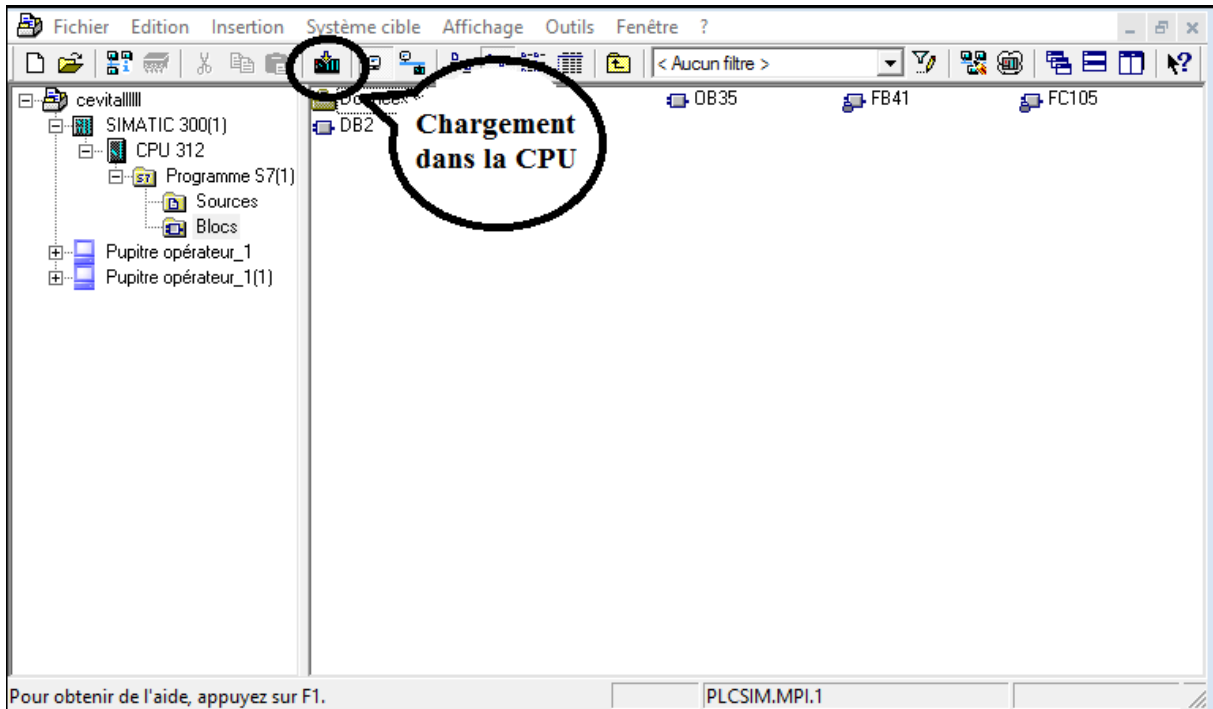


Figure III.12: Chargement de programme dans la CPU

III.6.2 Présentation de PLCSIM

L'application de simulation de modules S7-PLCSIM nous permet d'exécuter et de tester notre programme dans l'automate programmable (AP) que l'on simule dans l'ordinateur ou dans la console de programmation. La simulation étant complètement réalisée au sein du logiciel STEP 7, il n'est pas nécessaire qu'une liaison soit établie avec un matériel S7 quelconque (CPU ou module de signaux). S7-PLCSIM dispose d'une interface simple qui nous permet de visualiser et de forcer les différents paramètres utilisés par le programme (comme, par exemple, d'activer ou de désactiver des entrées). Tout en exécutant notre programme dans l'AP de simulation, nous avons également la possibilité de mettre en œuvre les diverses applications du logiciel STEP7 comme, par exemple, la table des variables (VAT) afin d'y visualiser et d'y forcer des variables

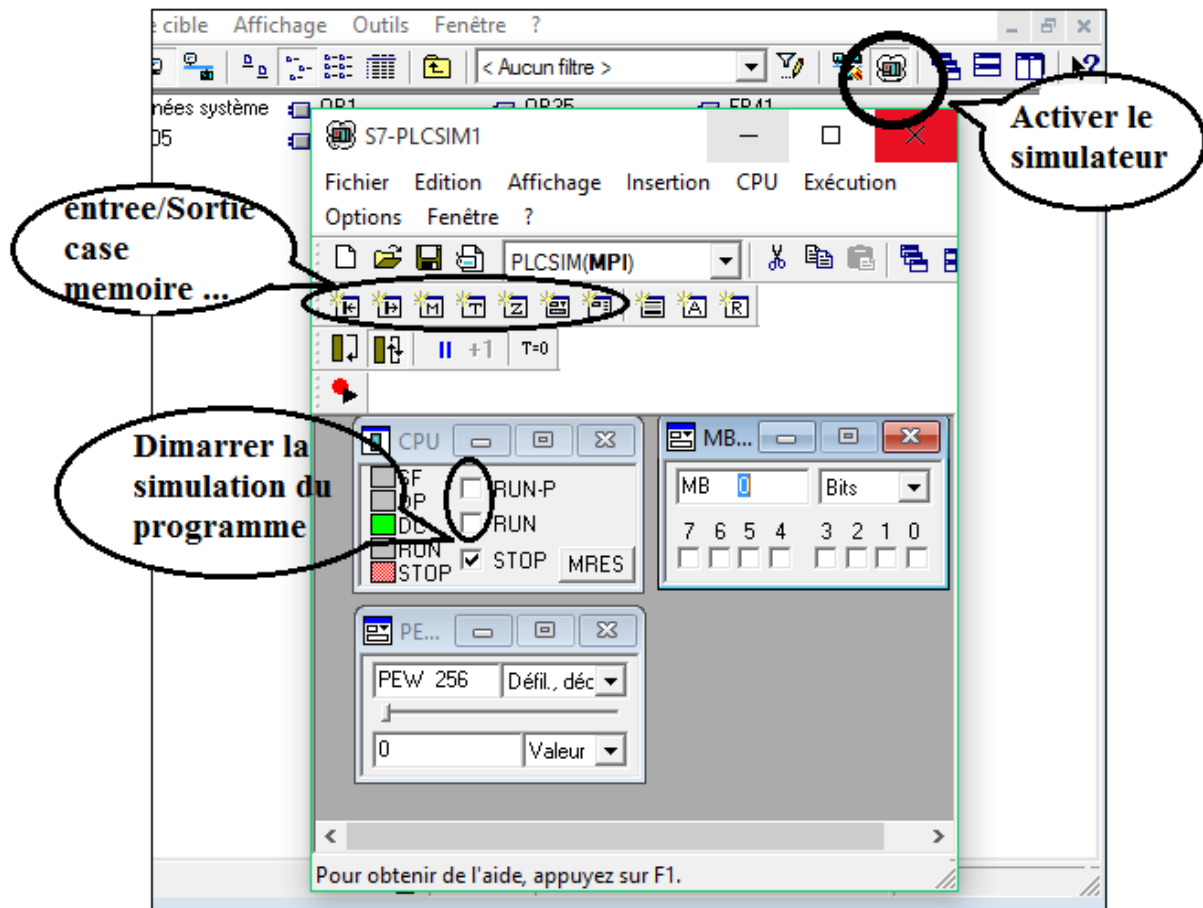


Figure III.13: Simulation de programme

Après avoir ouvrir le simulateur, on va passer aux étapes suivant :

- Faire l'appel aux entrées et sorties selon notre besoin.
- Mettre à (0) ou á (1) l'état des entrées.
- Cocher la case RUN ou RUN-P pour commencer la simulation ou arrêter le programme sur STOP.
- Double clic sur le bloc OB1 l'éditeur sera lancé et le contenu du programme sera affiche.
- Cliquer sur la paire de lunette pour visualiser le changement d'état des variables ainsi les éléments actifs en trait vert gras, et les éléments inactifs s'affiche en trait pointille en bleu. Comme illustre en bas

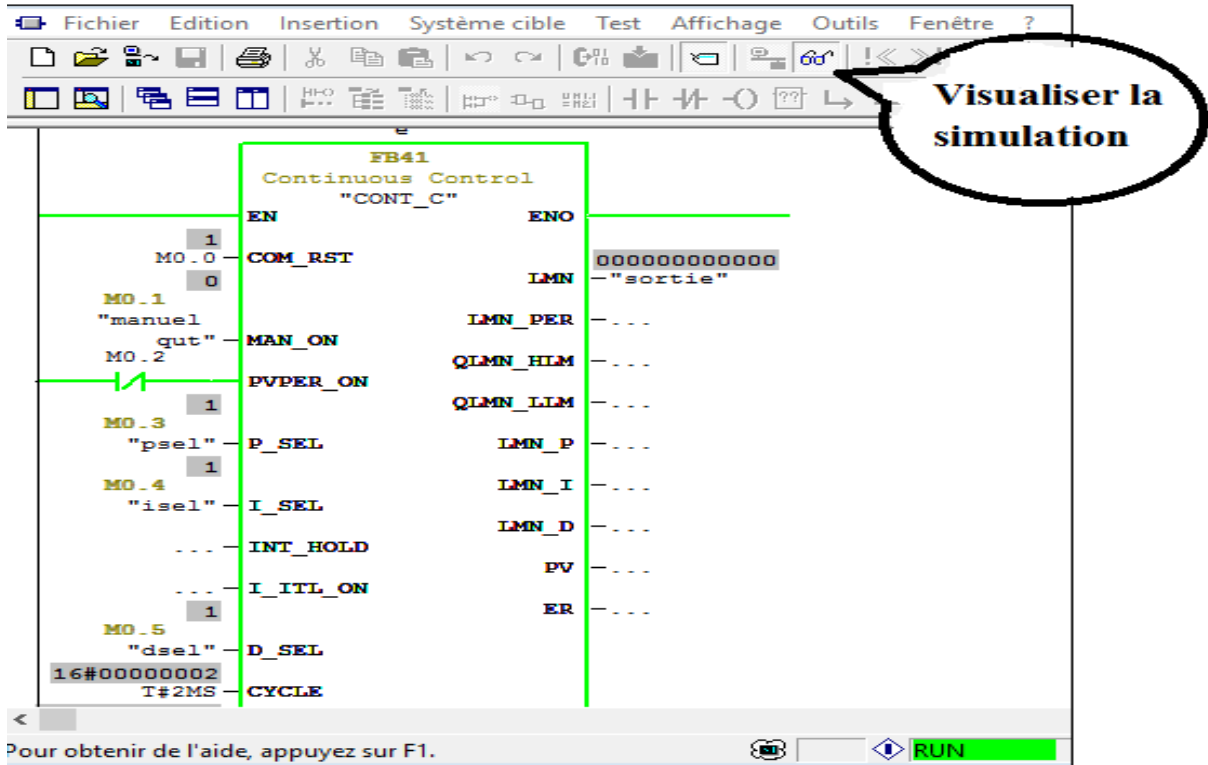


Figure III.14: Visualisation cas actif

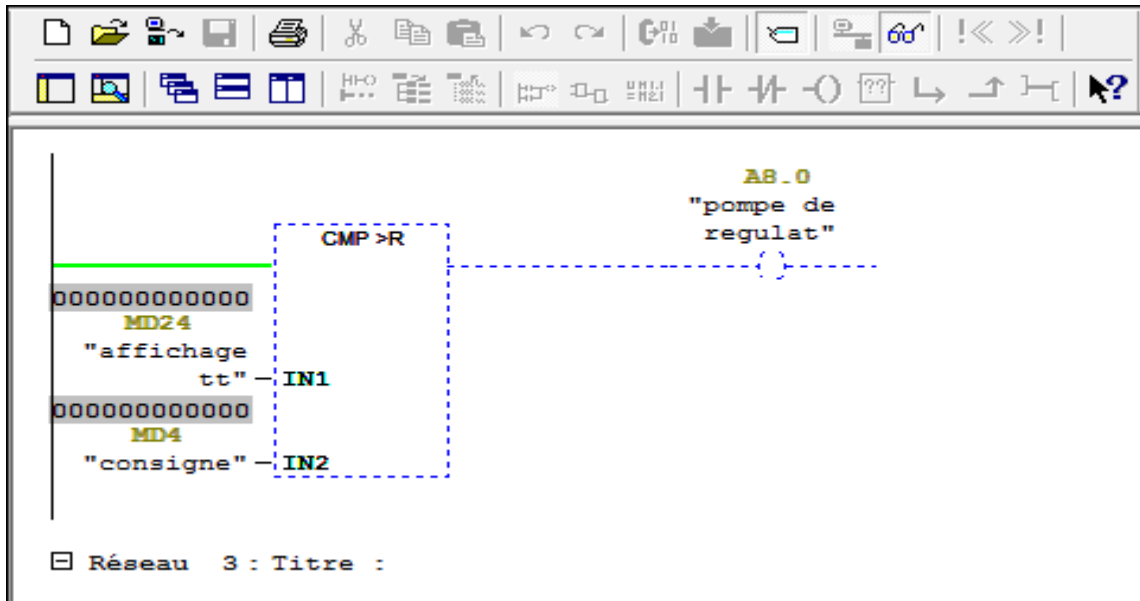


Figure III.15 : Visualisation cas inactif

III.7 Description de logiciel Win CC flexible

Win CC flexible est l'interface homme-machine (IHM) pour les applications de la machine et de processus dans la construction d'installation, il nous permet de visualiser le processus, ce qui facilite la surveillance par graphisme à l'écran, l'utilisateur peut par exemple démarrer ou arrêter un moteur, l'écran affiche un message en cas d'alarmes ou présence de défaut.



Figure III.16 : Logiciel WinCC Flexible

III.7.1 Utilisation de SIMATIC Win CC flexible

Win CC flexible est le logiciel IHM pour la réalisation, par des moyens d'ingénierie simples et efficaces, de concepts d'automatisation évolutifs, au niveau machine. Win CC flexible réunit les avantages suivants :

- Simplicité
- Ouverture
- Flexibilité

III.7.2 Présentation du système Win CC flexible

Pour réaliser la supervision sur WinCC Flexible on va choisir un type de pupitre par exemple dans notre cas (Ktp1000Basic DP) Voir la Figure

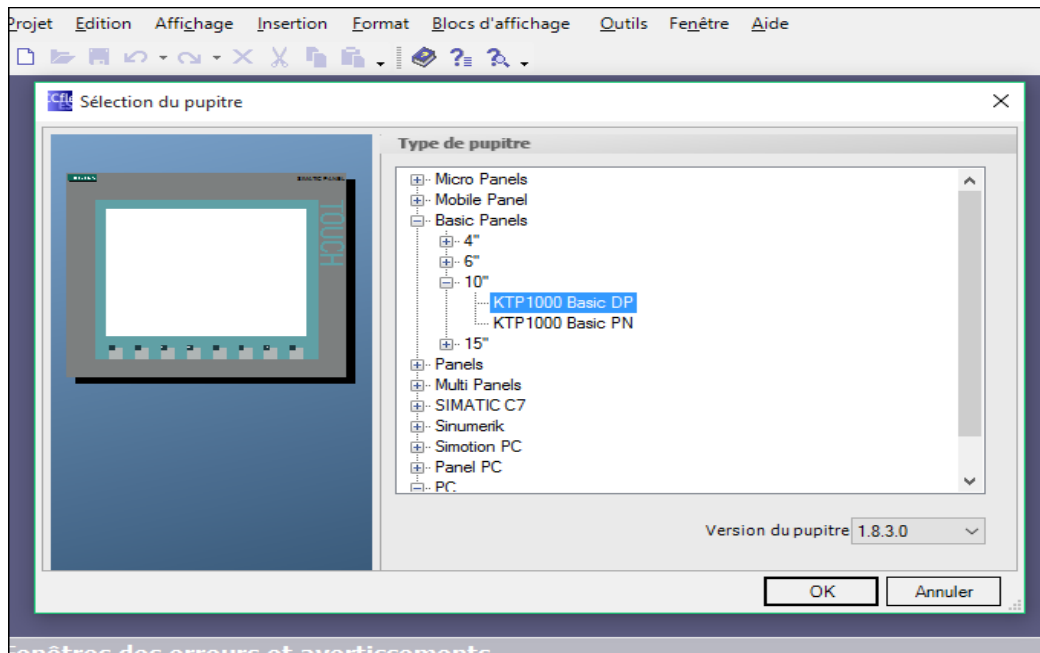


Figure III.17 : Choix type de pupitre

- **Éléments de Win CC flexible**

Après avoir choisir le pupitre pour la supervision en va commencer un projet vide, l'environnement de travail de Win CC Flexible se compose de plusieurs éléments. Certain de ces éléments sont liés à des éditeurs particuliers et uniquement visible lorsque cet éditeur est activé. Il met à disposition un éditeur spécifique pour chaque tâche de configuration.

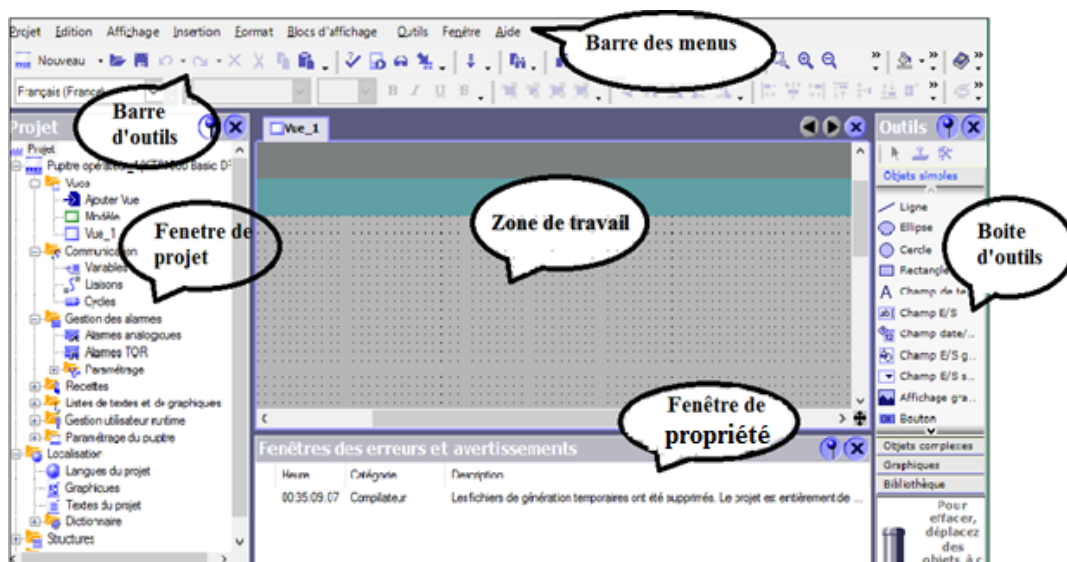


Figure III.18 : Aperçu de la fenêtre WinCC flexible

- **Zone de travail:** Sert à configurer des vues, de façon qu'il soit le plus compréhensible par l'utilisateur, et très facile à manipuler et consulter les résultats.

- **Barre d'outils** : La barre d'outils permet d'afficher tout dont le programmeur a besoin.
- **Barre des menus**: La barre des menus contient toutes les utilisations de WinCC flexible. Les raccourcis disponibles sont indiqués en regard de la commande du menu.
- **Boîte d'outils**: La fenêtre des outils propose un choix d'objets simple ou complexes qu'on insère dans les vues, par exemple commande.
- **Fenêtre des propriétés** : Actuelles dans la zone de travail, lorsqu'un projet est sélectionné, on peut étudier les propriétés de l'objet en question dans la fenêtre des propriétés.

III.7.3 Intégration dans Simatic STEP7

L'intégration dans l'environnement de configuration de Simatic STEP7 autorise la gestion de projet flexible WinCC Flexible à l'intérieur de STEP 7 et utilisation commune des paramètres de communication, variables et messages. Il en résulte une réduction de la fréquence d'erreur et par conséquent du travail de configuration.

Dans le cas d'une application intégrée, la gestion des projets s'effectue par le gestionnaire Simatic manager qui est une composante de STEP7. Le Simatic Manager nous donne accès à tous les objets de WinCC Flexible. On peut créer ou effacer des terminaux IHM, telle que des images ou recettes

Pour intégrer le projet dans STEP. En clique sur « projet » puis « intégrer dans STEP7 »

Pendant la configuration, on accède directement à la table des mnémoniques et aux blocs de donnée STEP7 qu'on a défini au moment d'établissement du programme d'automate. Il n'est pas nécessaire de définir des variables dans WinCC Flexible, la figure en bas illustre la liaison entre la station S7-300 et la station de supervision HMI.

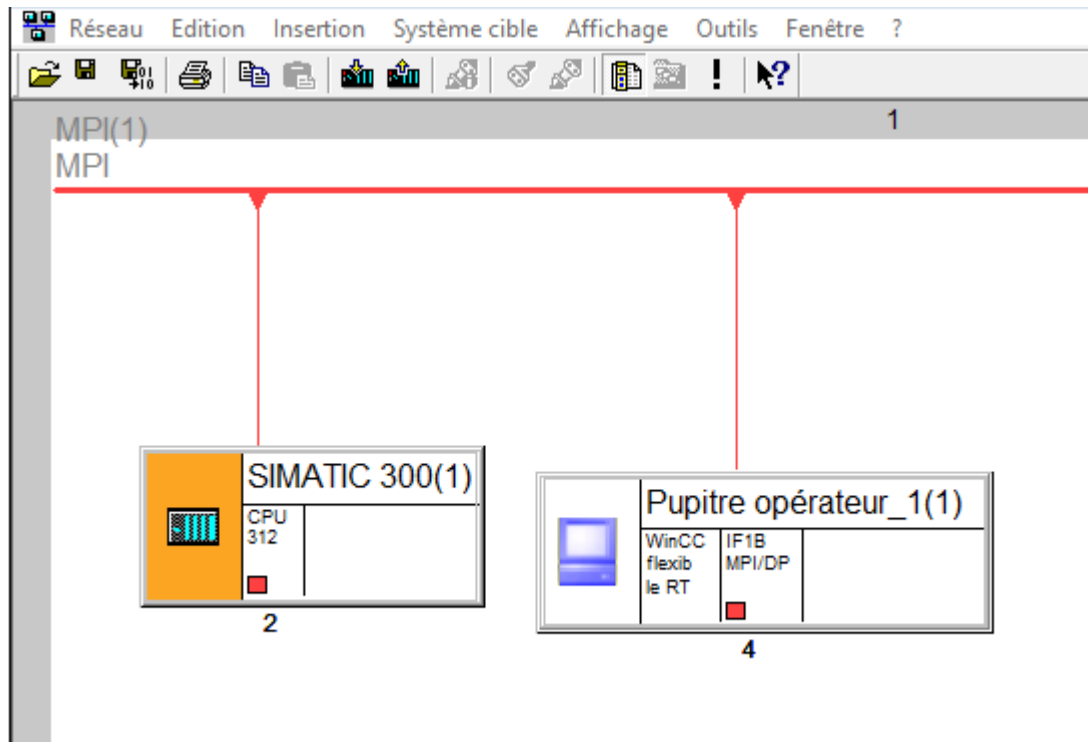


Figure III.19 : liaison entre la station S7-300 et la station HMI

III.8 Partie programmation

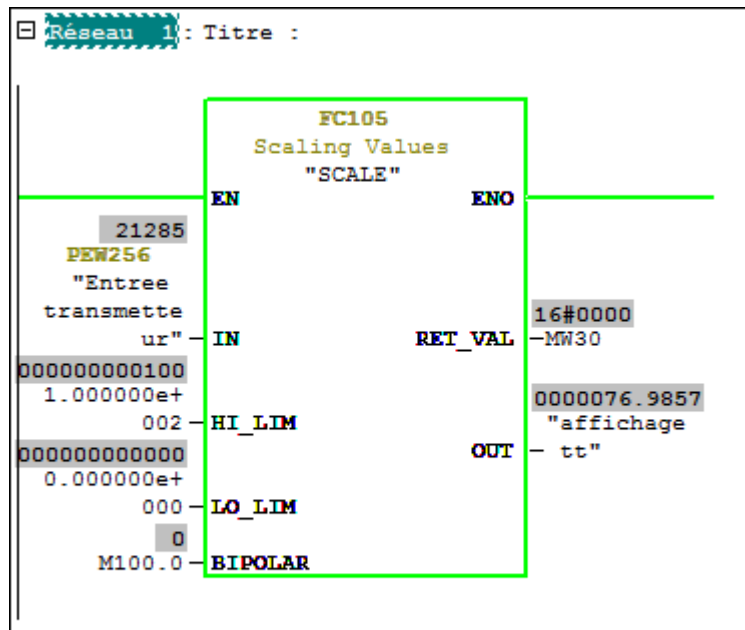
III.8.1 Programmation du mode manuel et du mode automatique

Pour le mode automatique, le fonctionnement se fait automatiquement sans l'intervention humaine, la commande (marche /arrêt) vient de l'automate programmable. Et le mode manuel se fait par l'intervention de l'opérateur, la commande (marche ou arrêt) s'effectuer par clique sur l'équipement du mode manuel à partir de la salle de contrôle.

- **La mise à l'échelle (scale)**

Le transmetteur convertir le signale physique (0C « LO_LIM » - 100C « HI_LIM ») vers un signal électrique courant (4-20mA)

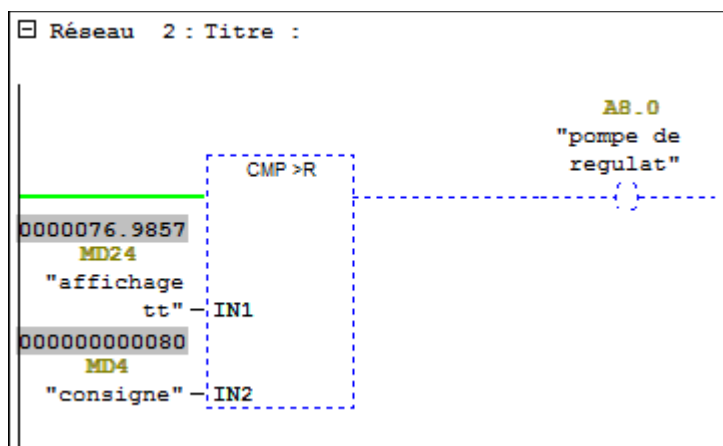
La fonction mise à l'échelle offre la conversion numérique de courant de 4-20 mA a (0« valeur minimal » et 27648« valeur maximal ») pour les envoyer vers l'automate.



• **Compateur**

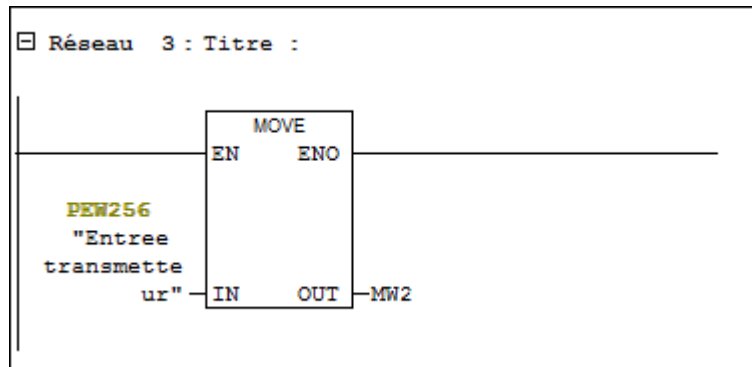
Le compateur va comparer la valeur qui afficher sur le transmetteur et la consigne

- C'est la valeur de transmetteur est supérieur a la valeur de la consigne la pompe va démarrer.
- C'est la valeur de transmetteur est inférieur à la valeur de la consigne la pompe va s'arrêter.



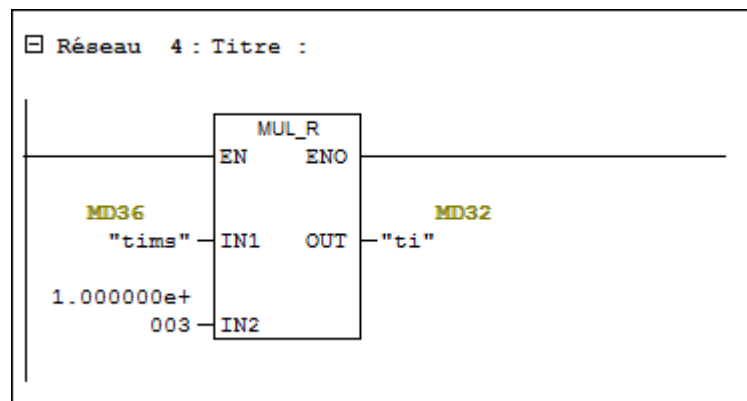
- **Mouvement (Move)**

La fonction move copie l'enterrée PEW 256 entier vers la sortie MV2



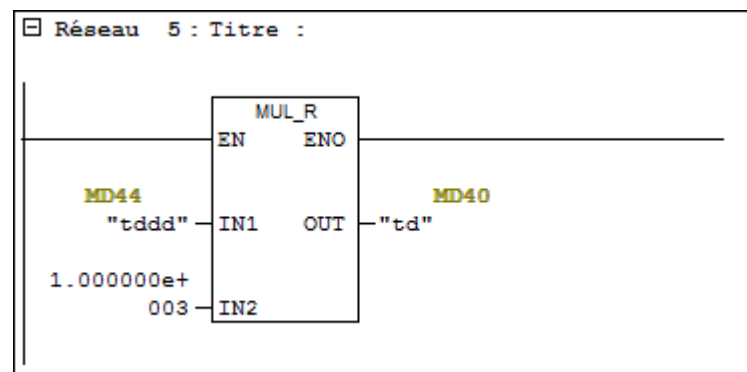
- **Temps de l'intégrales (Multiplication)**

C'est le temps de l'intégrales qui est sur le bloc FB41, il va multiplier le temps de l'intégrale MD36 fois 10^{-3} (de seconde vers milliseconde).



- **Temps de dérivée**

C'est le temps de la dérivée qui est sur le bloc FB41, il va multiplier le temps de dérivée MD44 fois 10^{-3} (de seconde vers milliseconde).



III.9 Simulation

III.9.1 Simulation du programme sous STEP7 AVEC PLCSIM

Après la programmation sous Step7 et on l'a intégré dans le WinCC Flexible pour réaliser le schéma de notre système.

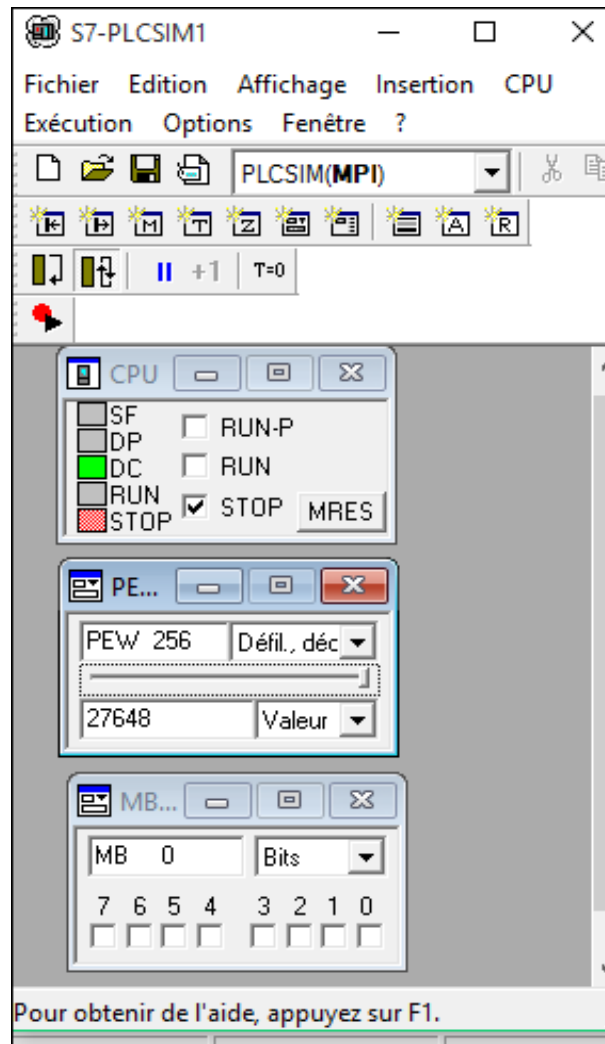


Figure III.20 : Simulateur S7-PLC SIM

PEW256 : C'est la valeur de la température.

MB0 : C'est l'affichage de transmetteur.

III.9.2 Simulation du programme sous WinCC

Système global

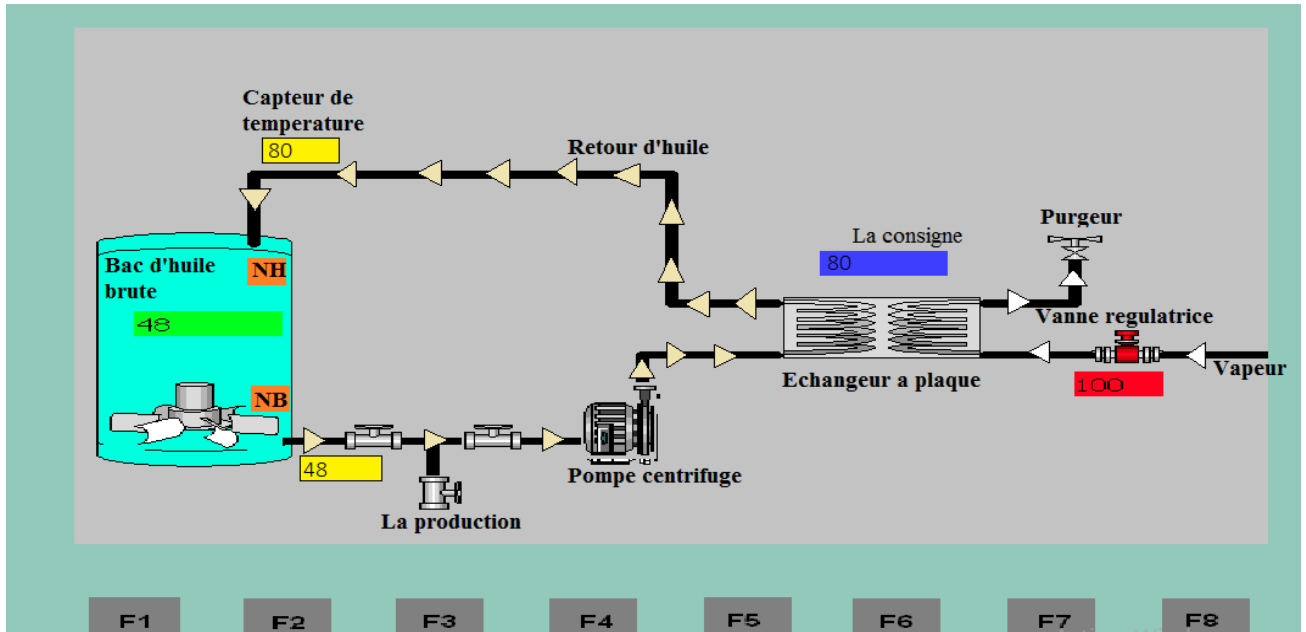


Figure III.21: Vue de système

III.9.3 Vue de régulateur PID

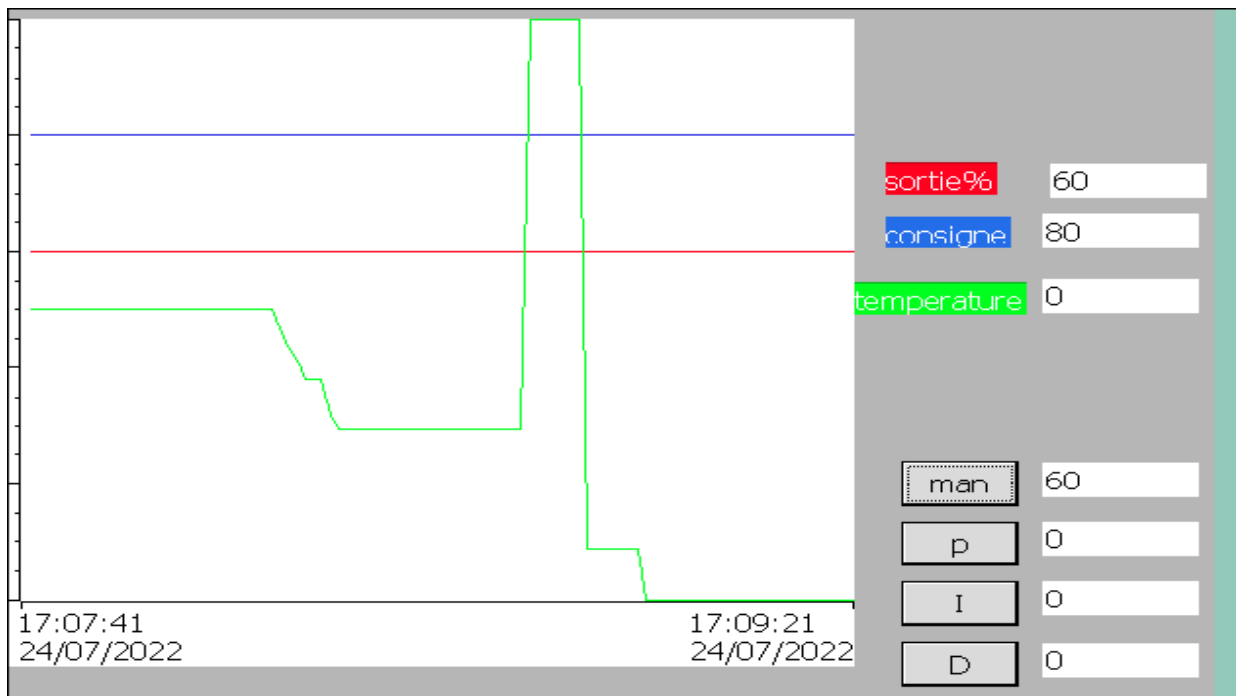


Figure III.22 : Graphe de la valeur manuelle sur WinCC flexible

Pour une valeur manuelle active et égale à 60 : quel que soit la valeur affichée sur le transmetteur (0C-100C), la vanne rester ouvert a (60%)

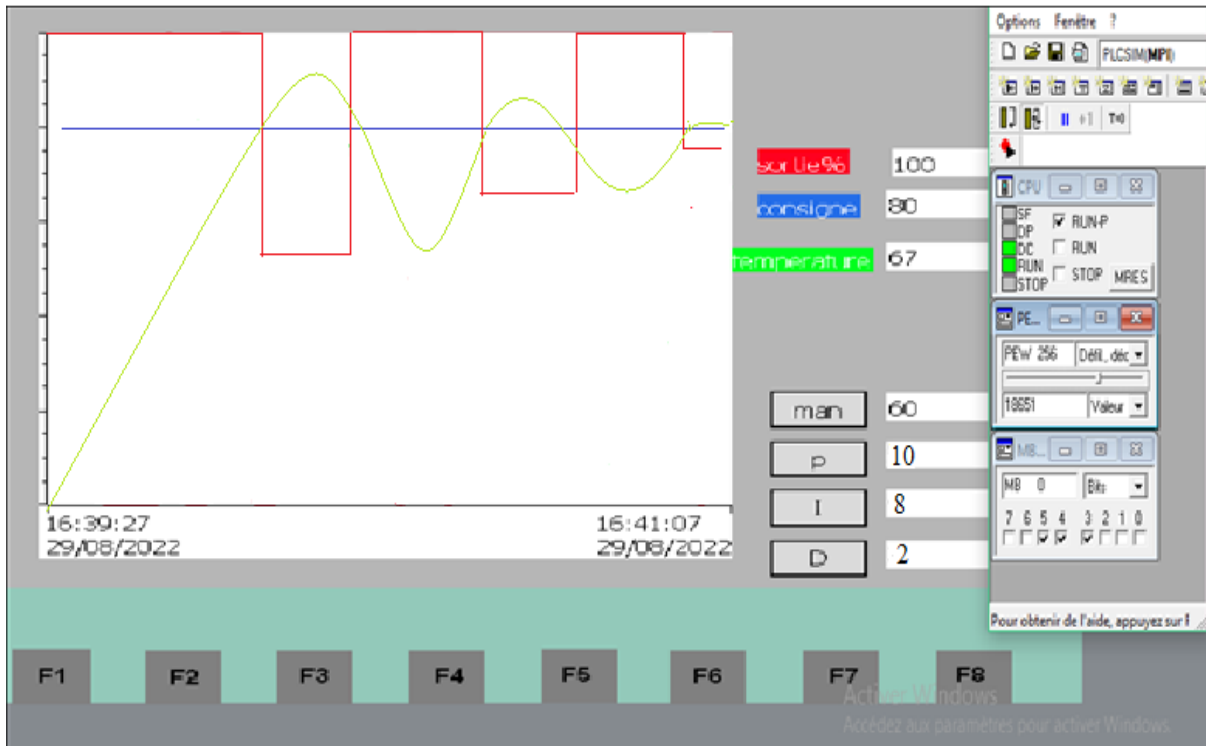


Figure III.23: Graphe de la regulation PID sur WinCC flexible

III.10 Conclusion

Dans ce chapitre, nous avons présenté les différentes étapes de la création de notre programme STEP7 et sa réalisation, après élaboration du programme de la régulation automatique de la boucle de chauffage industriel et on a arrivé à utiliser la simulation avec le S7-PLCSIM qui permettra de charger et de tester le programme à exécuter dans un automate SIMATIC S7-300.

Pour assurer la fiabilité de notre système, nous avons proposé un modèle de supervision qui permet de visualiser en temps réel le fonctionnement de la boucle de chauffage dans une station, et de répondre à la problématique qui a été posé depuis le début de notre stage au sein de raffinerie d'huile du complexe Cevital.



Conclusion général

Conclusion générale

Arrivant à la fin de notre projet de fin d'étude effectué au sein de la société CEVITAL de Bejaia, concernant la supervision d'une régulation automatique de la boucle de chauffage bac stockage huile brute, nous présentons le bilan du travail effectué.

L'étude détaillée de la boucle chauffage nous a permis de toucher à plusieurs disciplines que ça soit de la mécanique, l'instrumentation et la pneumatique, ce travail que nous avons effectué, nous a permis d'acquérir des connaissances techniques et pratiques, et d'améliorer nos connaissances théoriques dans un vaste domaine de l'électronique, et de l'automatique et de nous familiariser avec le milieu industriel.

En effet, nous avons réalisé notre projet de supervision de la station de boucle chauffage comprimé sur l'automate programmable SIEMENS de la gamme SIMATIC de type S7-300, et on a pu le contrôler avec le superviseur simple et puissant Win CC flexible ; afin d'obtenir :

- La supervision des paramètres de système ;
- L'amélioration de la qualité et l'augmentation de la quantité de produit ;
- Pouvoir réchauffer l'huile à la température désirée ;
- La surveillance à distance des différents paramètres de système ;
- Réduire l'effort physique et faciliter la tâche aux techniciens et ingénieurs de complexe CEVITAL.

Enfin, nous espérons que notre travail sera utile à toutes personnes intéressées par ce domaine



Bibliographie

- [1] : Eric KABANSHI, mémoire en ligne « Étude et conception d'un système de régulation automatique de la fréquence et de la tension de sortie d'une hydrolienne en fonction de la vitesse de la roue (cas du projet d'électrification décentralisée de la ferme Benjin Agriculture » Université de Lubumbashi, 2007
- [2] :Régulation industrielle –Notion de Base Article PDF.
- [3] : Abdelhamid Djari, Cour module « élément de régulation numérique » université Tbessa
- [4] : Melle Kerchouche Lamia & Nait Ali Sarah, « Régulation analogique P, PI et PID de la vitesse d'un moteur à courant continu d'un banc didactique 33-033 », mémoire fin d'étude, Département Electronique, Université MOULOU D MAMMERI DETIZI-OUZOU 2019.
- [5] :<https://www.technologuepro.com/Regulationindustrielle/Introduction-a-la-regulation.pdf> (consulté le 04Mai 2022)
- [6] : <http://www1.ucam.ac.ma/cnreee/RegulateurPID.pdf> (consulte 05 mai 2022)
- [7] : Mezaache, « Éléments de regulation numérique » cours, Université de Msila
- [8] : Alaa-eddine Bordjiba, « Conception et réalisation d'un Régulateur PID numérique de température avec ARDUINO », mémoire fin d'étude Département d'Electronique et Télécommunications université 08 Mai 1995 Guelma2019.
- [9] : Document de formation TIA, Module B3 Techniques de régulation avec STEP 7, Edition05/2004
- [10] :<https://www.verderliquids.com/fr/fr/pompes-par-principe/melangeurs-statiques-et-dynamiques> (consulte le 08 Mai 2022)
- [11] : <https://www.omega.fr/prodinfo/mesure-de-niveau.html>(consulte le 12mai 2022)
- [12] :<https://www.thermal-engineering.org/fr/quest-ce-quune-pompe-centrifuge-definition>(consulte le 14 mai 2022)
- [13] :Abbas Celia, «Installation et mise en marche d'un banc d'essai pour la construction des courbes caractéristiques de deux pompes identiques fonctionnant en parallèle », mémoire de master, département hydraulique urbaine, École notionnel 2017.
- [14]: <https://www.thermal-engineering.org/fr/quest-ce-quune-pompe-centrifuge-definition/>(consulte le 16mai 2022)
- [15] : Bouchemal Mohamed Lamine, Louici Houssam Eddine « modélisation des échangeurs de chaleur » mémoire de master, département électromécanique, Université Jijel,2018
- [16] : <https://www.abcclim.net/echangeur-a-plaques.html>(consulte 21 mai 2022)
- [17] :https://www.xpair.com/lexique/definition/vanne_de_regulation.htmc (consulté le 23mai 20220).
- [18] :<http://www.helyon.com/category/protection-contre-la-pression-et-le-vide/vanne-de-regulation> (consulté le 10 juin 2022)
- [19] : Sectoriel S.A, « Robinet 736-746 XS avec actionneur pneumatique AP », Edition 2012.

- [20] : <https://www.tlv.com/global/FR/steam-theory/what-is-a-steam-trap.html>(consulté le 11 juin 2022)
- [21] : <http://processs.free.fr/Pages/VersionWeb.php?page=4417#menuright> (consulté le 13 juin 2022).
- [22] : <https://www.hbm.com/fr/7646/definition-dun-capteur-de-pression-comment-ca-marche/>(consulté le 14 juin 2022)
- [23] : Oriol Planas, Ingénieur Technique Industriel, spécialité mécanique, Date de publication 11 juin 2021
- [24] : Mohamed Ladjal. cours génie électrique « automate-programmable-industriel », Université Med BOUDIAF Msila.
- [25] : William Bolton, « LES AUTOMATES PROGRAMMABLES INDUSTRIELS », Livre, Paris : Dunod, 2010
- [26] : <https://www.cours-et-exercices.com/2014/05/expose-sur-la-automates-programmables.html> (consulté le 17 juin 2022)
- [27] : <http://technologie-sciarretta.ovh/?p=739> (consulté le 22 juin 2022)
- [28] : https://www.geea.org/IMG/pdf/LES_AUTOMATES_PROGRAMMABLES_INDUSTRIELS_pour_GEEA.pdf (consulté le 20 juin 2022).
- [29] : Andre Simon « automate programmable niveau » Eddition L'ELAN-LIEGE 1991.
- [30] : https://fasoeducation.net/espace_eleves/secondaire/eftp/bac_technologique/automates_programmables_industriels/co/grainNatureinformation.html (consulté le 18 juin 2022)
- [31] : <http://1yc58-renardfolleureau.ac-dijon.fr/ljr/cours/supsi/si/commande-systeme/ch1-SystBoucles.pdf> (consulte le 21 juin 2022)
- [32] : <https://slideplayer.fr/slide/13428942/> (consulte le 23 juin 2022)
- [33] : JP VISKOVIc « Support Omron » ref : STA37-V3,25/10/2007.
- [34] : <https://scietech.fr/architectures-des-automates-programmables-industriels/> (consulté le 26 juin 2022)
- [35] : cours génie électrique, module « Automatismes Logiques Industriels », Mme EL Hammoumi, Université Sidi Mohamed Ben Abdellah de Fes
- [36] : <http://niedercorn.free.fr/iris/iris1/sysauto/autom1.pdf> (consulté le 27 juin 2022)
- [37] : <https://www.automation-sense.com/blog/automatisme/choix-d-un-automate-quelques-criteres-a-considerer.html>(consulte le 28 juin 2022)
- [38] : Alain. Canduro, cours « API »
- [39] : www.achoisir.fr/comment-choisir-un-automate-programmable-industriel/30aout 2018(consulte le 28 juin 2022)

[40] : Université de Reims Champagne Ardenne - 2020/descriptif-du-matériel.

[41] : <https://www.directindustry.fr/prod/siemens-automation-and-engineering/product-180278-2335811.html> (consulte le 29 juin 2022)

[42] : Michel G, « Architecture et application des automates programmables industriels », DUNOD, Paris, 1987

[43] : Automates programmables S7-300 caractéristiques électriques techniques des CPUSIMATIC 2001

[44] : AZAKRI Yacine & DRICI Youcef/Automatisation des vibrateurs électriques aux niveaux des trémies de conditionnement de sucre 1 Kg Cevital/2017

[45] : Mr. Amarouche Mohand Larbi/Étude et automatisation d'un filtre à sable au niveau d'une station d'épuration Cevital (Bejaia)/Mémoire fin d'étude, 2018

[46] : ELWE, « système automatisées, bus de terrain, API SIEMENS », système didactiques

[47] : Alain GONZAGA « Automates Programmables Industriels ».

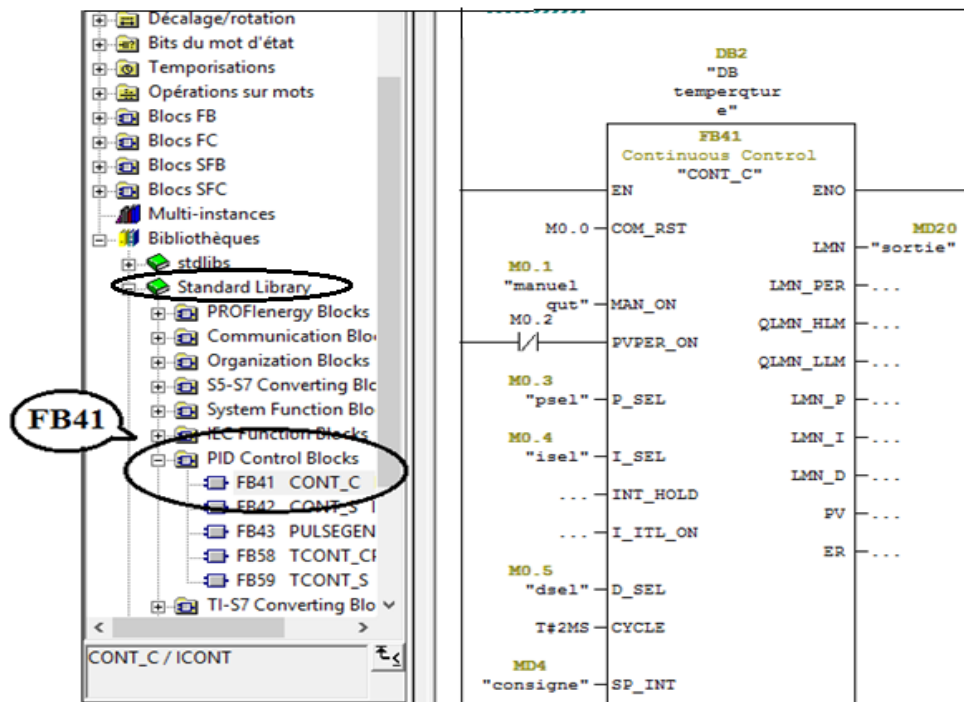
[48] : Siemens logiciel SIMATIC STEP 7 version 5.3.



Annexes

ANNEXE 1

Les étapes de sélectionné de FB41



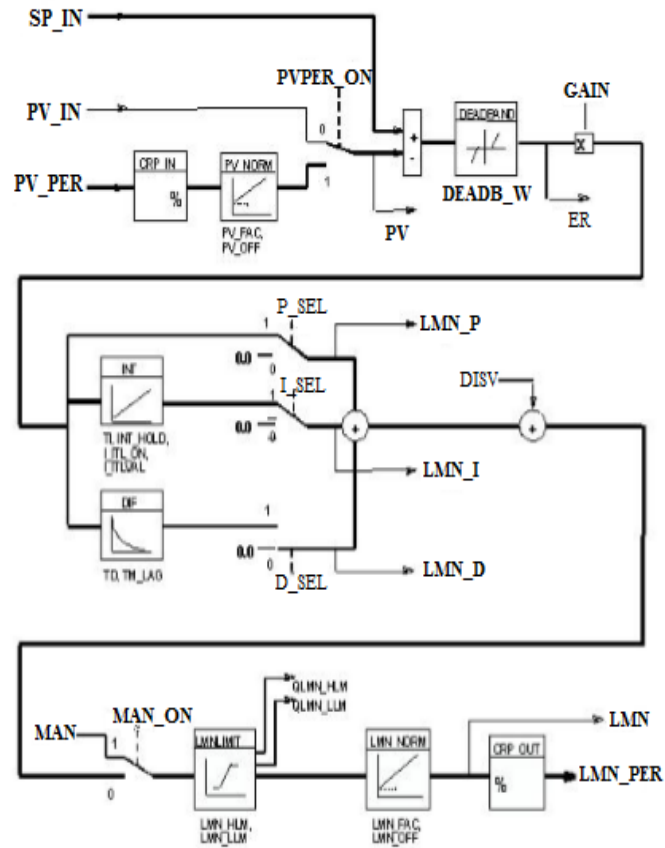
La fonction de l'OB35 (FB41)

DB2 "DB tempertur e"		FB41 Continuous Control "CONT_C"	
	EN	ENO	
MO.0	COM_RST	LMN	MD20 "sortie"
MO.1 "manuel gut"	MAN_ON	LMN_PER	...
MO.2	PVPER_ON	QLMN_HLM	...
MO.3 "psel"	P_SEL	LMN_P	...
MO.4 "isel"	I_SEL	LMN_I	...
...	INT_HOLD	LMN_D	...
...	I_ITL_ON	PV	...
MO.5 "dsel"	D_SEL	ER	...
T#2MS	CYCLE		
MD4 "consigne"	SP_INT		
MD8 "pv in"	PV_IN		
MW2	PV_PER		
MD12 "manuel"	MAN		

MD16 "gain"	GAIN
MD32 "ti"	TI
MD40 "td"	TD
...	TM_LAG
...	DEADB_W
...	LMN_HLM
...	LMN_LLM
...	PV_FAC
...	PV_OFF
...	LMN_FAC
...	LMN_OFF
...	I_ITLVAL
...	DISV

ANNEXE 2

Schémas fonctionnel d'un PID (bloc CONT_C (FB41)) :



ANNEXE 2

Paramètre d'entrée (input) du FB41 : [11]

Paramètre	Type de données	Valeurs admises	Par défaut	Description
COM_RST	BOOL		FALSE	COMPLETE RESTART Le bloc a un sous-programme d'initialisation qui est exécuté quand cette entrée est à 1.
MAN_ON	BOOL		TRUE	MANUAL VALUE ON / Activation du mode manuel Quand cette entrée est à 1, la boucle de régulation est interrompue. C'est une valeur manuelle qui est introduite comme valeur de réglage.
PVPER_ON	BOOL		FALSE	PROCESS VARIABLE PERIPHERY ON / Activation de la mesure de périphérie Pour que la mesure soit lue dans la périphérie, il faut relier l'entrée PV_PER à la périphérie et mettre à 1 l'entrée PVPER_ON.
P_SEL	BOOL		TRUE	PROPORTIONAL ACTION ON / Activation de l'action proportionnelle Dans l'algorithme PID, il est possible d'activer et de désactiver séparément chacune des actions. L'action P est active quand cette entrée est à 1.
I_SEL	BOOL		TRUE	INTEGRAL ACTION ON / Activation de l'action par intégration Dans l'algorithme PID, il est possible d'activer et de désactiver séparément chacune des actions. L'action I est active quand cette entrée est à 1.
INT_HOLD	BOOL		FALSE	INTEGRAL ACTION HOLD / Gel de l'action par intégration La sortie de l'intégrateur peut être gelée. Pour cela, il faut mettre à 1 cette entrée.
I_ITL_ON	BOOL		FALSE	INITIALIZATION OF THE INTEGRAL ACTION / Initialisation de l'action par intégration La sortie de l'intégrateur peut être forcée à la valeur initiale I_ITL_VAL. Pour cela, il faut mettre à 1 cette entrée.
D_SEL	BOOL		FALSE	DERIVATIVE ACTION ON / Activation de l'action par dérivation Dans l'algorithme PID, il est possible d'activer et de désactiver séparément chacune des actions. L'action D

Activer Windows
Accédez aux paramètres pour activer Windows.

CYCLE	TIME	>= 1 ms	T#1s	SAMPLE TIME / Période d'échantillonnage Le temps s'écoulant entre les appels de bloc doit être constant. Il est indiqué par cette entrée.
SP_INT	REAL	-100,0 à 100,0 (%) ou grandeur physique ¹	0,0	INTERNAL SETPOINT / Consigne interne Cette entrée sert à introduire une consigne.
PV_IN	REAL	-100,0 à 100,0 (%) ou grandeur physique ¹	0,0	PROCESS VARIABLE IN / Mesure interne Cette entrée permet de paramétrer une valeur de mise en service ou de relier une mesure externe en virgule flottante.
PV_PER	WORD		W#16#0000	PROCESS VARIABLE PERIPHERIE / Mesure de périphérie La mesure en format de périphérie est reliée au régulateur à cette entrée.
MAN	REAL	-100,0 à 100,0 (%) ou grandeur physique ²	0,0	MANUAL VALUE / Valeur manuelle Cette entrée sert à introduire une valeur manuelle grâce à des fonctions de contrôle-commande.
GAIN	REAL		2,0	PROPORTIONAL GAIN / Coefficient d'action proportionnelle Cette entrée indique le gain du régulateur.
TI	TIME	>= CYCLE	T#20 s	RESET TIME / Temps d'intégration Cette entrée détermine le comportement dans le temps de l'intégrateur.
TD	TIME	>= CYCLE	T#10 s	DERIVATIVE TIME / Temps de dérivation Cette entrée détermine le comportement dans le temps du dérivateur.
TM_LAG	TIME	>= CYCLE/2	T#2 s	TIME LAG OF THE DERIVATE ACTION / Retard de l'action par dérivation L'algorithme de l'action D contient un retard qui peut être paramétré à cette entrée.
DEADB_W	REAL	>= 0,0 (%) ou grandeur physique ¹	0,0	DEAD BAND WIDTH / Largeur de zone morte Le signal d'erreur est conduit par une zone morte. Cette entrée détermine la taille de la zone morte.

Activer Windows
Accédez aux paramètres pour activer Windows.

ANNEXE 2

LMN_HLM	REAL	LMN_LLM à 100,0 (%) ou grandeur physique ²	100,0	MANIPULATED VALUE HIGH LIMIT / Limite supérieure de la valeur de réglage La valeur de réglage est toujours limitée vers le haut et vers le bas. Cette entrée indique sa limite supérieure.
LMN_LLM	REAL	-100,0 à LMN_HLM (%) ou grandeur physique ²	0,0	MANIPULATED VALUE LOW LIMIT / Limite inférieure de la valeur de réglage La valeur de réglage est toujours limitée vers le haut et vers le bas. Cette entrée indique sa limite inférieure.
PV_FAC	REAL		1,0	PROCESS VARIABLE FACTOR / Facteur de normalisation Cette entrée est multipliée par la mesure. Elle sert à adapter la plage de la mesure.
PV_OFF	REAL		0,0	PROCESS VARIABLE OFFSET / Décalage de normalisation Cette entrée est ajoutée à la mesure. Elle sert à adapter la plage de la mesure.
LMN_FAC	REAL		1,0	MANIPULATED VALUE FACTOR / Facteur de valeur de réglage Cette entrée est multipliée par la valeur de réglage. Elle sert à adapter la plage de la valeur de réglage.
LMN_OFF	REAL		0,0	MANIPULATED VALUE OFFSET / Décalage de valeur de réglage Cette entrée est ajoutée à la valeur de réglage. Elle sert à adapter la plage de la valeur de réglage.
I_TLVAL	REAL	-100,0 à 100,0 (%) ou grandeur physique ²	0,0	INITIALIZATION VALUE OF THE INTEGRAL ACTION / Valeur d'initialisation pour l'action par intégration La sortie de l'intégrateur peut être forcée par l'entrée I_TL_ON. La valeur d'initialisation est à cette entrée.
DISV	REAL	-100,0 à 100,0 (%) ou grandeur physique ²	0,0	DISTURBANCE VARIABLE / Grandeur perturbatrice La grandeur perturbatrice de compensation est reliée à cette entrée.

Paramètres de sortie du FB41(OUTPUT) : [11]

Paramètre	Type de données	Valeurs admises	Par défaut	Description
LMN	REAL		0,0	MANIPULATED VALUE / Valeur de réglage Cette sortie donne en virgule flottante la valeur de réglage agissant réellement.
LMN_PER	WORD		W#16#0000	MANIPULATED VALUE PERIPHERY / Valeur de réglage de périphérie La valeur de réglage en format de périphérie est reliée avec le régulateur à cette sortie.
QLMN_HLM	BOOL		FALSE	HIGH LIMIT OF MANIPULATED VALUE REACHED / Limite supérieure de la valeur de réglage atteinte La valeur de réglage est toujours limitée vers le haut et vers le bas. Cette sortie signale le dépassement de sa limite supérieure.
QLMN_LLM	BOOL		FALSE	LOW LIMIT OF MANIPULATED VALUE REACHED / Limite inférieure de la valeur de réglage atteinte La valeur de réglage est toujours limitée vers le haut et vers le bas. Cette sortie signale le dépassement de sa limite inférieure.
LMN_P	REAL		0,0	PROPORTIONALITY COMPONENT / Composant P Cette sortie contient le composant proportionnel de la grandeur réglante.
LMN_I	REAL		0,0	INTEGRAL COMPONENT / Composant I Cette sortie contient le composant intégral de la grandeur réglante.
LMN_D	REAL		0,0	DERIVATIVE COMPONENT / Composant D Cette sortie contient le composant dérivé de la grandeur réglante.
PV	REAL		0,0	PROCESS VARIABLE / Mesure Cette sortie donne la mesure agissant réellement.
ER	REAL		0,0	ERROR SIGNAL / Signal d'erreur Cette sortie donne le signal d'erreur agissant réellement.

Activer Windows
Accédez aux paramètres pour activer Windows.

ANNEXE 3

Plaque signalétique de la pompe centrifuge

13700 kg/h					
Position et fonction :					
			Caractéristiques		
Rep	Type et marque	N de série	Débit (m³/H)	Puissance (Kw)	Hauteur (m)
	Viking L 4125	900 523 291 900 523 292	16	7,5 344 rpm	90
	Viking KK 4724	900 503 295 900 503 296	6	2,2 128 rpm	60

Plaque signalétique de l'échangeur a plaque

Caractéristiques :	
- Pression : 10 bars	- Nbre de plaques : origine : 28,
	actuelles : 50
- Température : 0 à 135 °c	- Surface de chaleur : 6,24 m ²
- Volume coté ½ : 8,6/9,2 litres	- Année de fabrication 2009
- N° de fabrication : 30101-48460 30101-48457	- Joint : HT nitrile
- Entrée vapeur S ₁ : 132 °c, sortie S ₂ : 132 °c	- Débit : 908 kg/h
- Entrée huile S ₃ : 30 °c, sortie S ₄ : 100 °c 13700 kg/h	- Débit :
Position et fonction :	

Les différents types de variables contenues dans le STEP7

Groupe	Type de donnée	Signification
Type de donnée binaire	BOOL, BYTE, WORD, DWORD	Les données de ce type 1 bit, 8bits (1 octet), 16bits (2octets),32bits (4octets)
Type de données sur caractère	CHAR	Les données de ce type occupent 1 caractère et usuel du jeu de caractère ASCII
Type de données numérique	INTEGER, DINT, REAL	Les données de ce type permettent de traiter la valeur numérique
Type de données temporel	TIME ,DATE TIME OF- DAY, TIME	Les données de ce type représentent les diverses valeur de durée et de date dans STEP7.

Résumé

Ce travail réalisé au sein de l'entreprise de Cevital présente l'étude de régulation de la boucle de chauffage qui comporte la réalisation d'un programme qui assure la mise en marche de cette boucle. Pour cela on a utilisé un programme sous l'environnement du logiciel STEP7 pour que l'automate SIMENS S7-300 assure le bon fonctionnement de cette boucle, le STEP7 contient un simulateur d'automate SIEMENS tel que le S7-300. Nous l'avons utilisé dans notre projet pour simuler le programme et de faire la communication vu l'absence de l'automate réel. Pour la supervision nous avons utilisé le logiciel WinCC.

Abstract

This work carried out within the company of Cevital presents the study of regulation of the heating loop which, and the realization of a program, which ensures the starting of this loop. For this, we used a program under the STEP7 software environment so that the SIMENS S7-300 controller ensures the proper functioning of this loop. STEP7 contains a SIEMENS controller simulator such as the S7-300. We used it in our project to simulate programs and to communicate given the absence of the real automaton. For the supervision, we used the WinCC software.

ملخص

يقدم هذا العمل المنفذ داخل شركة سيفيتال دراسة تنظيم حلقة التسخين، وتحقيق برنامج يضمن بدء هذه الحلقة. لهذا الغرض، استخدمنا برنامجًا ضمن بيئة برنامج STEP7 بحيث تضمن وحدة تحكم SIMENS S7-300 حسن سير هذه الحلقة. يحتوي STEP7 على محاكي وحدة تحكم SIEMENS مثل S7-300. استخدمناه في مشروعنا لمحاكاة البرامج وللتواصل نظرًا لغياب الآلي الحقيقي. للإشراف استخدمنا برنامج WinCC.